



Wir digitalisieren Ihre Geschäftsprozesse

X4 ESB

 **SoftProject**

Die in dieser Dokumentation enthaltenen Informationen und die zugehörigen Programme können ohne besondere Ankündigung geändert werden. Für etwaige Fehler übernimmt SoftProject keine Haftung.

Diese Dokumentation und die zugehörigen Programme dürfen ohne schriftliche Zustimmung der SoftProject GmbH weder ganz noch teilweise kopiert, reproduziert, verändert oder in irgendeine elektronische oder maschinenlesbare Form umgewandelt werden.

Alle genannten Warenzeichen sind Warenzeichen der jeweiligen Eigentümer.

Kontakt

SoftProject GmbH

Am Erlengraben 3

D-76275 Ettlingen

Website: www.softproject.de

Vertrieb

Telefon: +49 7243 56175-0

vertrieb@softproject.de

SoftProject-Support

Telefon: +49 7243 56175-333

support@softproject.de

© SoftProject GmbH. Alle Rechte vorbehalten.

Stand: 03.09.2020

Inhaltsverzeichnis

1	Process Editor	13
1.1	Die Oberfläche des Process Editors	13
1.2	Die Symbolleiste des Process Editors	19
1.3	Kontextmenü im X4 Designer	21
1.4	Prozessbausteine und ihre Verwendung	23
1.4.1	Control-Bausteine	23
1.4.2	Data-Bausteine	41
1.4.3	Modul-Bausteine	67
1.4.4	Drawing	84
1.5	Tastenkombinationen im Process Editor	89
2	Mit ESB-Projekten arbeiten	91
2.1	Die Struktur von ESB-Projekten	91
2.2	ESB-Projekt anlegen	91
2.3	Projekte verwalten	92
2.3.1	Repository-Elemente importieren und exportieren	92
2.3.2	Repository-Elemente intelligent umbenennen	93
2.3.3	Repository-Elemente intelligent verschieben	95
2.3.4	Repository-Referenzen reparieren	95
2.3.5	Nicht verwendete Ressourcen löschen	96
2.3.6	Repository-Elemente durchsuchen	97
2.3.7	Globale Webservice-Konfiguration verwenden	98
2.3.8	Veraltete Projekte migrieren	99
3	Prozesse anlegen und modellieren	100
3.1	Prozess anlegen	100
3.2	Eigenschaften von Prozessen (.wrf)	101
3.3	Prozessbausteine ausrichten	103
3.4	Prozesse validieren	104
3.4.1	Prozess-Validierung durchführen	104
3.4.2	Standardmäßig hinterlegte Validierungsregeln	105

3.4.3	Benutzerdefinierte Validierungsregeln	106
3.5	Prozessbausteine parametrieren	108
3.5.1	Parametrierungs-Möglichkeiten	108
3.5.2	Benutzerdefinierte Parameter verwenden.....	109
3.5.3	Parameter dynamisch setzen	110
3.5.4	Status mit ConditionState dynamisch setzen	113
3.6	Platzhalter verwenden.....	114
3.6.1	System-Platzhalter.....	115
3.6.2	Benutzerdefinierte Platzhalter verwenden	120
3.6.3	Benutzerdefinierte Platzhalter und -gruppen verwalten.....	124
3.7	XSL-Hilfsfunktionen.....	128
3.7.1	TimeHelper (Veraltet)	128
3.7.2	WorkingTimeCalculator (Veraltet).....	137
3.7.3	checkXML	144
3.8	XSLT-Funktionen	145
3.9	XPath-Funktionen	148
3.9.1	Knotengruppen-Funktionen (node-set)	148
3.9.2	Boolesche Funktionen (boolean).....	149
3.9.3	Number-Funktionen.....	149
3.9.4	Zeichenketten-Funktionen (String).....	150
3.10	Entwickler-Hilfswerkzeuge	152
3.11	URL-Protokolle.....	154
3.11.1	context:-URL-Protokoll.....	154
3.11.2	xstore://-URL-Protokoll	155
3.11.3	x4db://-URL-Protokoll.....	156
3.11.4	cryptstore://-URL-Protokoll.....	157
3.11.5	globalcontext:-URL-Protokoll	159
3.12	Logging von technischen Prozessen.....	160
3.13	Save Points für wiederaufsetzbare Prozesse verwenden	162
3.14	Webservices via HTTP(S) bereitstellen	163
3.14.1	Webservice-Konfiguration grafisch bearbeiten	165
3.14.2	ReSTful-Webservices bereitstellen	169

3.14.3	SOAP-Webservices bereitstellen	173
3.14.4	File-Webservice bereitstellen	187
3.14.5	WADL-Services bereitstellen	190
3.15	Prozesse debuggen und ausführen	191
3.15.1	Prozess debuggen	191
3.15.2	Prozess ausführen.....	193
3.16	COBOL Copybook Converter Wizard.....	193
3.16.1	Voraussetzungen	194
3.16.2	Funktionen im COBOL Copybook Wizard	194
4	Prozess-Dokumentation in der X4 Suite	195
5	Mit Adapter-Projekten arbeiten	198
6	Übersicht der X4 Adapter	199
6.1	Über X4 Adapter	199
6.2	Big Data	199
6.2.1	Amazon S3 DynamoDB Connector	199
6.2.2	Apache Cassandra Connector.....	209
6.2.3	ElasticSearch	217
6.2.4	Google Cloud Storage BigTable Connector.....	231
6.2.5	Microsoft Azure Data Lake Gen 1 Storage Connector	237
6.3	Cloud.....	243
6.3.1	Amazon S3 Simple Storage Connector.....	243
6.3.2	Amazon SimpleDB Connector	252
6.3.3	Dropbox Connector	258
6.3.4	Google Cloud Storage Objects Connector	261
6.3.5	Google Drive Connector	270
6.3.6	Microsoft Azure Blob Storage Connector	275
6.3.7	Microsoft Azure Table Storage Connector	281
6.4	Communication	287
6.4.1	AS/400 PCML	287
6.4.2	File System	288
6.4.3	File System Bulk.....	293
6.4.4	FTP	300

6.4.5	iCalendar	317
6.4.6	Ping.....	322
6.4.7	SCP	323
6.4.8	SMS.....	326
6.4.9	TAPI Listener	327
6.4.10	TCP/IP Connector/Listener	331
6.4.11	UDP Listener	333
6.5	Compression	334
6.5.1	gzip Converter.....	334
6.5.2	ZIP	335
6.6	Databases	343
6.6.1	JDBC Connector.....	343
6.6.2	JDBC Iterator	365
6.6.3	JDBC Metadata Explorer.....	387
6.6.4	MongoDB Connector.....	398
6.7	EDI	413
6.7.1	EDIFACT Adapter	413
6.7.2	Unterstützte EDIFACT-Nachrichten-Typen	420
6.8	E-Mail.....	424
6.8.1	IMAP.....	424
6.8.2	MIME Multipart Converter	428
6.8.3	MSG Converter	450
6.8.4	POP/SMTP.....	450
6.9	ERP	456
6.9.1	Abas ERP Connector	456
6.9.2	SAP Connector	461
6.10	File Processing.....	514
6.10.1	ASCII to XML.....	514
6.10.2	Base64 Batch Decoder	520
6.10.3	Base64 Converter.....	520
6.10.4	Charset Changer.....	521
6.10.5	CSV	522

6.10.6	Encoding Converter.....	535
6.10.7	Hex Converter	536
6.10.8	Image Processing.....	536
6.10.9	Java Properties File Reader.....	540
6.10.10	QR Code Generator	541
6.10.11	Regular Expression	543
6.10.12	Stream Splitter.....	544
6.10.13	Structured Text.....	545
6.10.14	Text Replacement	548
6.10.15	Transliteration Adapter	549
6.11	Geo Location	550
6.11.1	Bing Maps.....	550
6.11.2	Geo Calculation	555
6.11.3	Google Maps.....	563
6.12	Insurance	568
6.12.1	Audatex DV90 Converter	568
6.12.2	GDV	569
6.12.3	SZF Reader.....	572
6.13	Internet of Things.....	573
6.13.1	IEC 61850 Client.....	573
6.13.2	Modbus TCP Connector	577
6.13.3	MQTT Publisher/Subscriber	580
6.13.4	OPC Client Connector	593
6.13.5	OPC UA Publisher/Subscriber	595
6.13.6	SIMATIC S7 Connector	605
6.14	Message Oriented Middleware	607
6.14.1	Apache Kafka	607
6.14.2	AMQP Publisher/Subscriber	616
6.14.3	JMS.....	622
6.14.4	Websphere MQ	628
6.15	Office	653
6.15.1	Microsoft Excel	653

6.15.2	Microsoft Word Properties	662
6.15.3	Microsoft Word to Text Converter.....	664
6.15.4	Working Time Calculator	665
6.16	PDF Processing	669
6.16.1	PDF Data Processing	669
6.16.2	PDF Image Processing.....	676
6.16.3	PDF Split and Merge	685
6.17	Security	690
6.17.1	Antivirus.....	690
6.17.2	BCrypt Calculator	692
6.17.3	Checksum Creator.....	694
6.17.4	Digest Calculator	695
6.17.5	JSON WebToken Calculator.....	696
6.17.6	LDAP Connector.....	697
6.17.7	Password Generator.....	703
6.17.8	PEP Adapter	705
6.17.9	Random Number Generator	708
6.17.10	RSA Encryption	709
6.17.11	SAML Calculator	710
6.17.12	SSH Connector.....	714
6.17.13	UUID Generator	716
6.17.14	XML Encryption	717
6.17.15	XML Signature	722
6.17.16	AES Encryption	723
6.18	System Administration	730
6.18.1	JMeter Test Execution	730
6.18.2	JMX MBean	735
6.18.3	JMX Number	740
6.18.4	JMX Switch	742
6.18.5	Nagios Listener.....	743
6.18.6	SNMP Management	747
6.18.7	Windows Event Reader.....	750

6.18.8	Windows Services Explorer	752
6.19	Web Services	753
6.19.1	HTTP(S) Output Parameter	753
6.19.2	JSON.....	755
6.19.3	OData Connector	760
6.19.4	URL Encoding Converter	771
6.19.5	URL Reader	773
6.19.6	Webservice	774
6.19.7	X4 Streaming Support	800
6.19.8	XOP (MTOM).....	802
6.19.9	YAML Converter	812
6.20	XML Tools.....	814
6.20.1	Inline Content Expander	814
6.20.2	Record Counter	817
6.20.3	XHTML Form Extractor	818
6.20.4	XML Checker	823
6.20.5	XML Element Wrapper.....	823
6.20.6	XML Formatter	824
6.20.7	XML Iterator	824
6.20.8	XML Validation	826
6.20.9	XPath Iterator.....	827
6.21	X4 Tools	829
6.21.1	Case Management Documents	829
6.21.2	Case Management	831
6.21.3	Case Management Statistics	834
6.21.4	X4 Variable Collector	837
6.21.5	Commandline	839
6.21.6	Custom Placeholder Initializer	842
6.21.7	Custom Placeholder Storage Manager	844
6.21.8	Document Comparison	847
6.21.9	Document Properties Changer	849
6.21.10	Empty Output Adapter.....	849

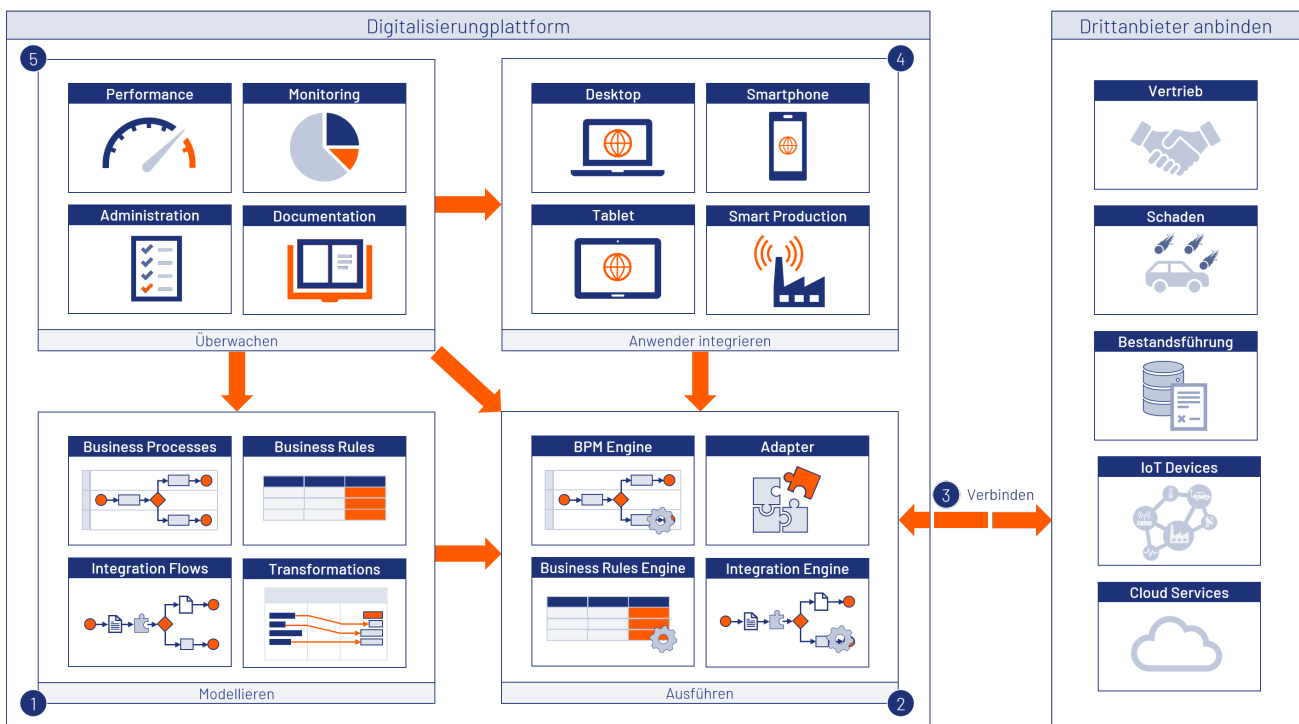
6.21.11 Global Context Manipulation	850
6.21.12 Log4J Logging	852
6.21.13 Parallel Process Starter	854
6.21.14 Process Starter	856
6.21.15 Process Tester	859
6.21.16 Project Access	861
6.21.17 Task Management	866
6.21.18 User Identity Adapter	869
6.21.19 Web App Translation	872
6.21.20 X4 BAM Logging	873
6.21.21 X4 Management.....	875
6.21.22 X4 Server Config Reader.....	875

Über die X4 Suite

Digitalisierung braucht eine ganzheitliche Betrachtung, die sich in der einzusetzenden Lösung widerspiegeln muss. Als zentrale Plattform unterstützt Sie die X4 Suite dabei, diese Herausforderungen zu lösen. Im Fokus stehen dabei die Modellierung, Implementierung und Überwachung Ihrer Geschäftsprozesse. Daher enthält die X4 Suite alle benötigten Werkzeuge und ist mit einer Vielzahl an Schnittstellen und Formaten kompatibel. Mit der X4 Suite vermeiden Sie isolierte Informationssilos, produktivitätshemmende Medienbrüche und beschleunigen die Digitalisierung.

Geschäftsprozesse ohne Programmieraufwand zu realisieren, ermöglicht einem großen Anwenderkreis den Einstieg in das Management von Geschäftsprozessen. Das lohnt sich, denn Mitarbeiter der Fachabteilung wissen in der Regel am besten, worauf es bei den jeweiligen Geschäftsabläufen im Kern ankommt. Setzen Sie daher auf die X4 Suite als Plattform, deren Werkzeuge die Komplexität soweit reduzieren, dass sich auch ohne Programmierkenntnisse Geschäftsprozesse analysieren, optimieren, modellieren als auch kontrollieren und dokumentieren lassen. Alle Werkzeuge unterstützen eine integrierte, grafische Prozessmodellierung und -implementierung und erzeugen Prozesse, die von der X4 Suite performant ausgeführt werden.

- **X4 Designer:** Prozesse und Regeln grafisch modellieren
- **X4 Server:** Simulation und Ausführung der Prozesse und Regeln
- **X4 Adapter:** Drittsysteme in Prozesse integrieren
- **X4 Activities:** Web Apps für Mitarbeiter und Kunden bereitstellen
- **X4 Control Center:** Alle Prozesse und Apps überwachen und verwalten



An wen richtet sich dieses Dokument?

Dieses Dokument bietet Prozessentwicklern und Anwendern der X4 Suite ein umfangreiches Dokument, das ihnen bei der Modellierung von flexiblen Prozesslösungen mit der X4 Suite hilft. Die Dokumentation macht Sie mit den grundlegenden Prinzipien der Modellierung von technischen Prozessen mit der X4 Suite vertraut und erläutert die Bedienung des X4 Designers und der verschiedenen Werkzeuge, die bei der Modellierung von ausführbaren Prozessen eingesetzt werden. Zudem bietet das Handbuch alle relevanten Referenzinformationen zu sämtlichen Adaptern, die in der X4 Suite enthalten sind.

1 Process Editor

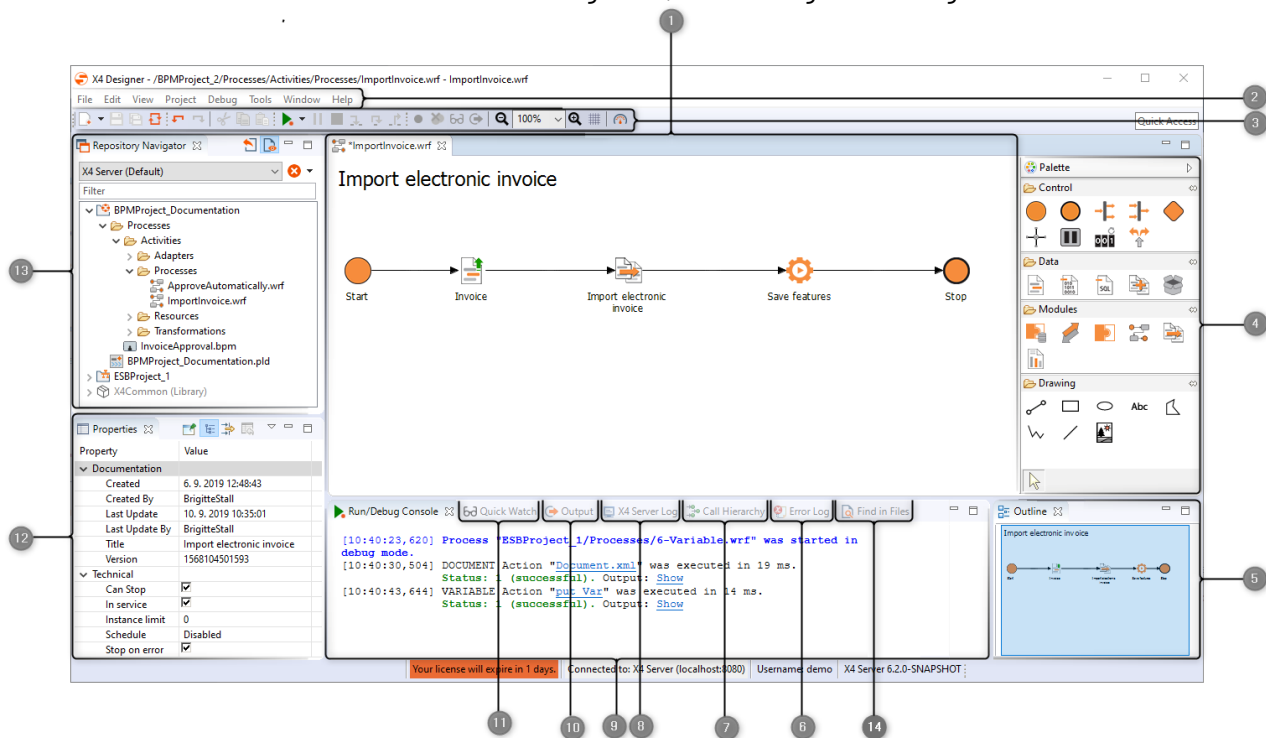
Mit dem Process Editor lassen sich IT-Prozesse grafisch modellieren. Zu jedem Prozess und zu den einzelnen Prozessbausteinen können Sie zudem festlegen, wie diese ausgeführt werden sollen.

Mit dem Process Editor können Sie u. a.:

- Verbindungen zwischen verschiedenen Back-End-Systemen herstellen,
- Daten in Echtzeit transformieren,
- Geschäftsregeln im modellierten Prozess implementieren und
- modellierte Prozesse debuggen und auf dem X4 Server ausführen.

1.1 Die Oberfläche des Process Editors

Der Process Editor ist in mehrere Bereiche aufgeteilt, die im Folgenden vorgestellt werden.




1	Editoren und Zeichnungsfläche	Prozesse und Dateien werden hier zum Anzeigen und Bearbeiten in Registerkarten geöffnet.
2	Menüleiste	In der Menüleiste lassen sich verschiedene Funktionen und Editoren des X4 Designers aufrufen.
3	Symbolleiste	Über die Symbolleiste können verschiedene Funktionen und Anzeigeoptionen aufgerufen werden. Einige Symbole sind nur aktiv, wenn die entsprechende Funktion für das ausgewählte oder geöffnete Element verfügbar ist, siehe auch Symbolleiste des BPMN Editors.

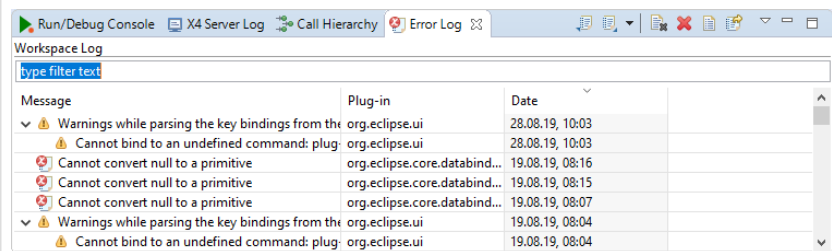
4	Palette	Die Palette enthält Prozess-Bausteine, die per Drag&Drop in die Zeichnungsfläche gezogen werden können. Abhängig vom verwendeten Editor stehen unterschiedliche Bausteine zur Verfügung, siehe auch BPMN-Prozessbausteine und ihre Verwendung.
---	----------------	--

Sichten








5	Outline	Zeigt das vollständige Prozess-Diagramm als Abbildung. Durch Anklicken der einzelnen Bereiche der Abbildung kann leicht durch den Prozess navigiert werden, selbst wenn dieser größer als die Größe des Designer-Fensters ist.
---	----------------	--


 Die Outline-Sicht lässt sich über das Menü **View > Outline** aufrufen.


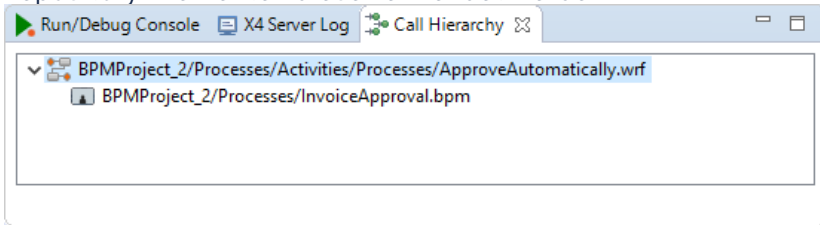
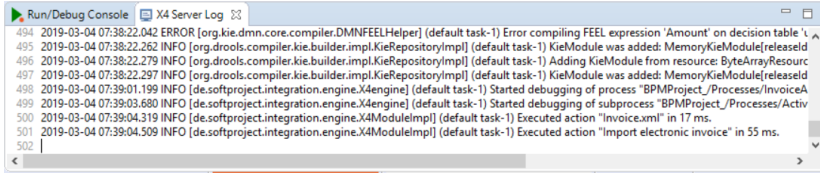

6	Error Log	Im Error Log lassen sich im X4 Designer aufgetretene Fehler einsehen.
---	------------------	---



Über die Symbolleiste lassen sich zusätzlich folgende Aktionen vornehmen:

-  **Export Log:** Error Log exportieren und als .log-Datei speichern
-  **Import Log:** Error Log (.log) importieren
-  **Clear Log Viewer:** Error Log bereinigen
-  **Delete Log:** Error Log unwiderruflich löschen
-  **Open Log:** Log-Datei (.log) im X4 Designer öffnen
-  **Restore Log:** Error Log wiederherstellen
-  **View Menu:** Zusatzoptionen öffnen


 Das Error Log lässt sich über das Menü **View > Error Log** aufrufen.

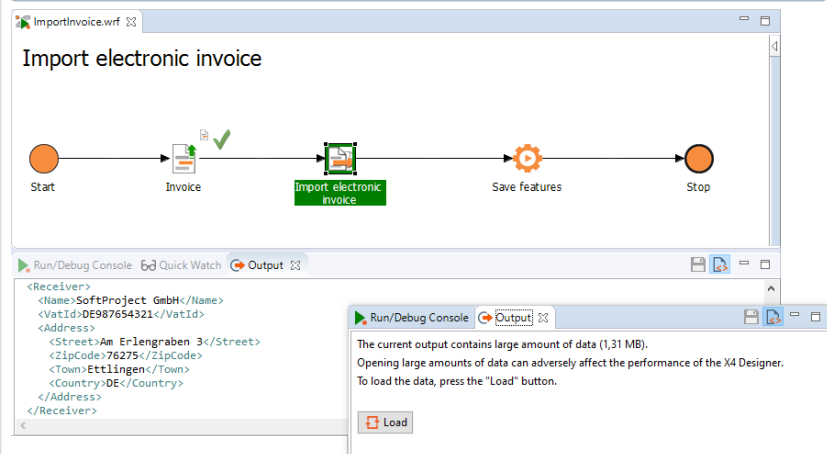
7	Run/Debug Console	<p>Wird ein Prozess auf Fehler überprüft bzw. simuliert, werden in der Run/Debug Console-Sicht zu jedem ausgeführten Prozess-Schritt Debug-Informationen angezeigt. Fehlermeldungen werden rot hervorgehoben, erfolgreiche Statusmeldungen sind grün.</p> <p>Wenn ein Prozess-Schritt ein Zwischenergebnis ausgibt, wird dieses während des Debuggens in einer temporären Datei gespeichert, die Sie über den entsprechenden Verweis ansehen können (z. B. <i>file:///1376054979039/</i>).</p> <div data-bbox="611 633 1434 745"> <p> Die Run/Debug-Console-Sicht lässt sich über das Menü View > Run/Debug Console aufrufen.</p> </div>
8	Call Hierarchy	<p>In der Call-Hierarchy-Sicht lässt sich zu jedem Prozess und Prozessbaustein im Repository anzeigen, von welchen Repository-Elementen diese verwendet werden.</p>  <p>Durch Rechtsklick auf das gewünschte Element im Repository Navigator wird ein Kontextmenü aufzurufen. Mit View Call Hierarchy wird anschließend eine Auflistung aller Repository-Elemente angezeigt, die mindestens eine Referenz auf das markierte Element enthalten.</p>
9	X4 Server Log	<p>In der Server-Log-Sicht lassen sich der Server-Status und insbesondere Fehler im X4-Server-Log aus dem X4 Designer heraus verfolgen.</p>  <div data-bbox="611 1675 1434 1787"> <p> Die Server-Log-Sicht lässt sich über das Menü View > X4 Server Log aufrufen.</p> </div>

10 Output



In der Output-Sicht lassen sich die Ergebnisse jedes einzelnen Debug-Schrittes einsehen.



- Die Output-Sicht lässt sich während des Debuggens über das Menü **Debug > Output** oder über das Output-Icon  in der Symbolleiste aufrufen.
- Enthält das Ergebnis Inhalte, die größer als 1 MB sind, wird eine Warnung angezeigt und das Ergebnis lässt sich über eine Schaltfläche herunterladen.

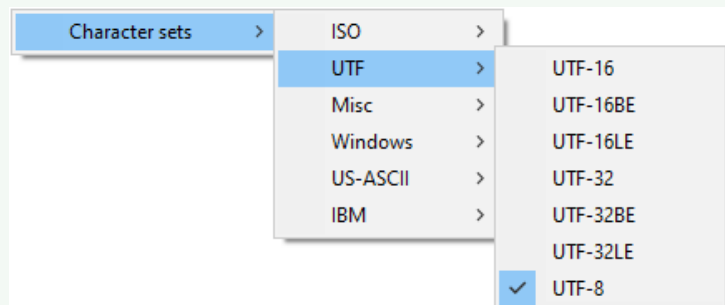


Über die Symbolleiste lassen sich zusätzlich folgende Aktionen vornehmen:

-  *Save content of output view*: Daten aus der Output-Sicht speichern
-  *Format the content of the output view*: Daten aus der Output-Sicht formatieren



Über das Kontextmenü **Character Sets** lässt sich bei Bedarf die Zeichenkodierung ändern.



- Große Datenmengen können sich negativ auf die Leistungsfähigkeit des X4 Designers auswirken.

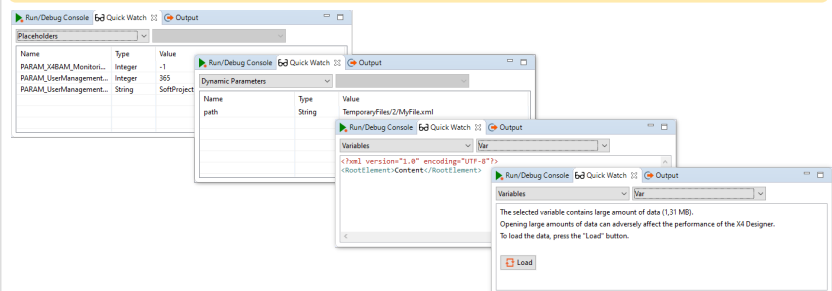
- In der Output-Sicht werden standardmäßig nur Dateien bis zu einer Dateigröße von 50 MB angezeigt. Die maximale Dateigröße kann in der `standalone.xml` durch den Administrator festgelegt werden (vgl. [WildFly-Dokumentation](#)). Dateien, die größer sind als die festgelegte maximale Dateigröße, können abgespeichert und geprüft werden.


11 Quick Watch

In der Quick-Watch-Sicht lassen sich Platzhalter, dynamischen Parameter und Kontextvariablen mit deren Werten während des Debuggens betrachten.

Ausgewählte Variablen werden nach jedem Debug-Schritt aktualisiert.

⚠ Enthält das Ergebnis des Debug-Schrittes Inhalte, die größer als 1 MB sind, wird eine Warnung angezeigt und das Ergebnis lässt sich über eine Schaltfläche herunterladen. Große Datenmengen können sich negativ auf die Leistungsfähigkeit des X4 Designers auswirken.

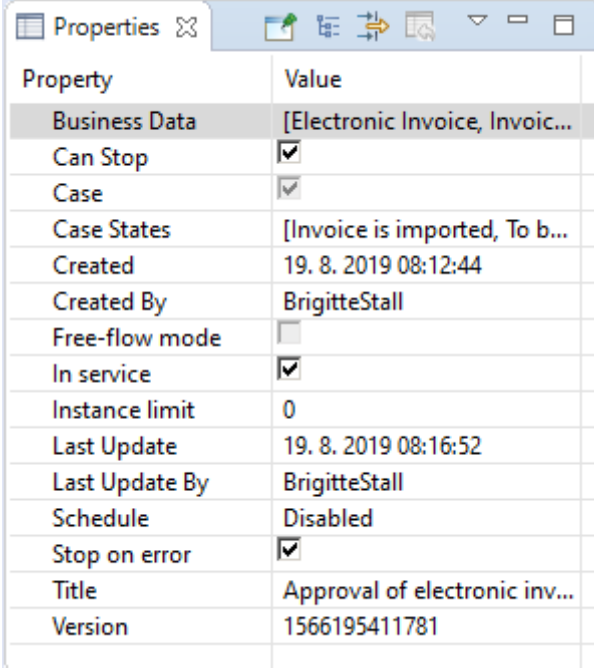


- ① Die Quick-Watch-Sicht lässt sich während des Debuggens über das Menü **Debug > Quick Watch** oder über das Quick-Watch-Icon  in der Symbolleiste aufrufen.

12 Properties






Hier lassen sich die Eigenschaften von markierten Prozessbausteinen und von geöffneten Prozessen anzeigen und bearbeiten:


- In der linken Spalte befinden sich die Eigenschaften, in der rechten Spalte die zugehörigen Werte.
- Für jeden Prozesse kann ein Titel sowie eine Beschreibung hinterlegt werden. Diese werden in der erzeugten Dokumentation angezeigt.
- Um die Eigenschaften eines geöffneten Prozesses zu bearbeiten, auf einen leeren Bereich auf der Zeichnungsfläche klicken.



Property	Value
Business Data	[Electronic Invoice, Invoic...
Can Stop	<input checked="" type="checkbox"/>
Case	<input checked="" type="checkbox"/>
Case States	[Invoice is imported, To b...
Created	19. 8. 2019 08:12:44
Created By	BrigitteStall
Free-flow mode	<input type="checkbox"/>
In service	<input checked="" type="checkbox"/>
Instance limit	0
Last Update	19. 8. 2019 08:16:52
Last Update By	BrigitteStall
Schedule	Disabled
Stop on error	<input checked="" type="checkbox"/>
Title	Approval of electronic inv...
Version	1566195411781






Standardmäßig werden die gebräuchlichsten Eigenschaften angezeigt, über die Symbolleiste der Properties-Sicht lassen sich jedoch zusätzliche Einstellungen vornehmen:




-  *Pin to Selection*: Die angezeigten Eigenschaften in der Properties-Sicht anheften
-  *Show Categories*: Eigenschaften nach Kategorien geordnet anzeigen
-  *Show Advanced Properties*: Erweiterte Eigenschaften anzeigen
-  *Restore Default Value*: Auf Standardwerte zurücksetzen
-  *View Menu*: Zusatzoptionen öffnen




13	Repository Navigator	<p>Die Repository Navigator-Sicht enthält eine Baumstruktur des verbundenen Repositorys mit verschiedenen Elementen wie Projekten, Ordnern und Dateien. Das Kontextmenü lässt sich per Rechtsklick auf ein Element im Repository Navigator aufrufen. Die verfügbaren Optionen ändern sich je nachdem an welcher Stelle in der Baumstruktur das Kontextmenü aufgerufen wird.</p> <p>Über das Filter-Textfeld über dem Projektbaum kann nach Dateien innerhalb des verbundenen Repositorys gesucht werden. Platzhalter wie * oder ? sind nicht möglich.</p> <p>Über verschiedene Schaltflächen lassen sich folgende Aktionen vornehmen:</p> <ul style="list-style-type: none"> •  <i>Jump from active Editor to Repository Navigator:</i> Geöffnete Datei im Editor im Repository Navigator markieren. •  <i>Show or hide all files:</i> Inhalt des Dateisystems einblenden und ausblenden. Aus dem Projekt exkludierte Dateien werden dabei ausgegraut dargestellt.
14	Find in Files	Die Find in Files-Sicht werden die Ergebnisse der Suche im Repository angezeigt. Dateien können über Edit > Find in Files durchsucht werden.
	Entwickler-Hilfswerkzeuge	<p>Entwickler-Hilfswerkzeuge als Sicht öffnen, siehe auch Entwickler-Hilfswerkzeuge</p> <ul style="list-style-type: none"> • Base64: Zeichenketten Base64-kodieren oder dekodieren • Digest: Hash-Wert für Zeichenketten erzeugen • Regex: Regulären Ausdruck (<i>Regex</i>) auf eine Zeichenkette anwenden, um den Ausdruck zu testen oder um die Zeichenkette gemäß einem Muster direkt zu ersetzen • URLCodec: Zeichenketten URL-kodieren oder dekodieren

1.2 Die Symbolleiste des Process Editors

Über die Symbolleiste können verschiedene Funktionen und Anzeigeeoptionen aufgerufen werden. Einige Symbole sind nur aktiv, wenn die entsprechende Funktion für das ausgewählte oder geöffnete Element verfügbar ist.

	<i>New</i>	<p>Neues Element hinzufügen, z. B. einen BPM-Prozess</p> <p> Die Einfügeoption variiert je nachdem, wo man sich in der Baumstruktur im Repository Navigator befindet.</p>
	<i>Save</i>	Änderungen im aktuell geöffneten Fenster speichern
	<i>Save all</i>	Alle Änderungen speichern
	<i>Refresh</i>	Server-seitige Aktualisierung durchführen

	<i>Undo</i>	Letzten Bearbeitungsschritt rückgängig machen
	<i>Redo</i>	Rückgängig gemachten Schritt wiederherstellen
	<i>Cut</i>	Auswahl ausschneiden
	<i>Copy</i>	Auswahl kopieren
	<i>Paste</i>	Auswahl einfügen
	<i>Start Debugging</i>	Optionen zur Prozessausführung aufrufen <ul style="list-style-type: none"> • Start Debugging: Prozess debuggen (Fehler anzeigen und beseitigen) • Start Without Debugging: Prozess direkt ausführen • Debug/Run Options: Run/Debug-Einstellungen im X4 Designer öffnen
	<i>Suspend</i>	Debugging-Vorgang anhalten
	<i>Resume</i>	Debugging fortsetzen
	<i>Stop Debugging</i>	Debugging beenden
	<i>Step Into</i>	Prozess-Schritt ausführen (Schritte des Subprozesses anzeigen)
	<i>Step Over</i>	Prozess-Schritt ausführen (ohne Schritte des Subprozesses anzuzeigen)
	<i>Step Return</i>	Aus Subprozess herausspringen
	<i>Toggle Breakpoint</i>	Breakpoint (Debug-Haltepunkt) zu markiertem Baustein setzen oder entfernen
	<i>Clear all Breakpoints</i>	Alle Breakpoints in diesem Prozess löschen
	<i>Quick Watch</i>	Quick Watch-Sicht öffnen, um Platzhalter, dynamischen Parameter und Kontextvariablen mit deren Werten während des Debuggens zu betrachten.
	<i>Output</i>	Output-Sicht öffnen, um das Ergebnis jedes Prozess-Schrittes während des Debuggens zu betrachten.
	<i>Zoom Out</i>	Ansicht verkleinern
	<i>Zoom</i>	Zoom-Stufe wählen

	<i>Zoom In</i>	Ansicht vergrößern
	<i>Snap to grid</i>	Raster einblenden und Bausteine am Raster ausrichten
	<i>Show Process Quality Index</i>	Process Quality Index einblenden, um die Qualität des modellierten Prozesses in einem Diagramm anzuzeigen

1.3 Kontextmenü im X4 Designer

Kontextmenü für Prozessbausteine


Wenn Sie im Editor-Bereich des X4 Designers auf einen Prozessbaustein rechtsklicken, stehen Ihnen je nach Bausteinart und Prozesstyp verschiedene Funktionen zur Verfügung:

<i>Add Component</i>	Neue Prozesskomponente hinzufügen
<i>Layout</i>	Ausgewählte Prozesskomponenten ausrichten <ul style="list-style-type: none"> • Align Horizontal: Komponenten anhand des Mittelwertes horizontal ausrichten • Align Vertical: Komponenten anhand des Mittelwertes vertikal ausrichten • Distribute Horizontal: Komponenten gleichmäßig horizontal verteilen • Distribute Vertical: Komponenten gleichmäßig vertikal verteilen
<i>Select All</i>	Alle Prozesskomponenten auswählen
<i>Copy Process Diagram to Clipboard</i>	Prozessdiagramm zwischenspeichern
<i>Edit Label...</i>	Namen der Prozesskomponente bearbeiten
<i>Undo ...</i>	Rückgängig machen
<i>Redo ...</i>	Letzten Schritt wiederherstellen
<i>Copy</i>	Markierten Prozessbaustein kopieren
<i>Cut</i>	Markierten Prozessbaustein ausschneiden
<i>Set ... document</i>	Markierten Baustein durch ein anderes Element im Repository austauschen
<i>Edit ...</i>	Markierten Baustein/Datei bearbeiten (nur Modul-Bausteine und Dokumente)
<i>Toggle Breakpoint</i>	Debug-Haltepunkt setzen
<i>Delete</i>	Prozessbaustein aus dem Prozessdiagramm entfernen
<i>Validate</i>	Subprozess auf Gültigkeit prüfen (Nur bei Subprozessen)
<i>Preview</i>	Prozessbaustein in separatem Editor öffnen (Nur bei Modul-Bausteinen und Dokumenten)

<i>Show input schema</i>	Input-Schema des Subprozesses anzeigen, falls vorhanden
<i>Show output schema</i>	Output-Schema des Subprozesses anzeigen, falls vorhanden
<i>Select in Repository Navigator</i>	Prozessbaustein im Repository Navigator zeigen
<i>View Call Hierarchy</i>	Verwendungsnachweis für den markierten Prozessbaustein anzeigen
<i>View Documentation</i>	(Nur bei Subprozessen): Prozess-Dokumentation temporär im Editor-Bereich anzeigen
<i>Refresh</i>	Prozessbaustein neu aus dem Repository laden (Nur bei Modul-Bausteinen und Dokumenten)
<i>Bring to Front</i>	Symbol in den Vordergrund des Diagramms setzen
<i>Bring Forward</i>	Symbol im Diagramm eine Ebene nach vorne setzen
<i>Send Backward</i>	Symbol im Diagramm eine Ebene nach hinten bringen
<i>Send to Back</i>	Symbol in den Hintergrund des Diagramms setzen
<i>Properties</i>	Eigenschaften des Prozessbausteins in der Properties-Sicht anzeigen
<i>Schema Declaration ...</i>	Schema-Deklaration für den Prozess hinterlegen

Kontextmenü im Repository Navigator

Das Kontextmenü können Sie per Rechtsklick auf ein Element im **Repository Navigator** aufrufen.

<i>New</i>	Neue Elemente (Prozesse, Ordner, Adapter etc.) im Repository anlegen
<i>Open / Open with</i>	Markiertes Element (Prozess, Ordner, Adapter etc.) mit dem Standard-Editor oder einem bestimmten Editor öffnen
<i>Save As Template</i>	Markiertes Element als Vorlage speichern
<i>Copy full path to Clipboard / Copy XStore URL to Clipboard</i>	Pfad zur markierten Datei in die Zwischenablage kopieren (als Ordnerpfad oder als xstore://-URL)
<i>Cut, Copy und Paste</i>	Markierte(s) Element(e) ausschneiden, kopieren oder einfügen
<i>Delete</i>	Markiertes Element aus dem Repository löschen (kann nicht rückgängig gemacht werden)
<i>Refactor/Rename</i>	Element (und falls gewünscht dessen Referenzen) umbenennen
<i>Exclude from Project</i>	Ressource aus dem Projekt ausschließen
	<div>  <ul style="list-style-type: none"> Die Datei wird im Dateisystem beibehalten, ist jedoch nicht mehr Bestand des Projektes und wird ausgegraut dargestellt. Die üblichen Operationen wie öffnen, bearbeiten, kopieren, löschen usw. von ausgeschlossenen Dateien ist unterbunden. </div>
<i>Include to Project</i>	Ausgeschlossene Ressource wieder in das Projekt aufnehmen

<i>Repair References</i>	Pfade von Repository-Ressourcen in Prozessdiagrammen reparieren
<i>Validate</i>	Prozess auf Gültigkeit prüfen
<i>Show Process in X4 Control Center</i>	Prozess im X4 Control Center anzeigen
<i>Show Process Documentation</i>	Prozessdokumentation erzeugen und im Browser anzeigen
<i>View Call Hierarchy</i>	Anzeigen, in welchen Prozessen dieses Element verwendet wird
<i>Start Without Debugging</i>	Prozess direkt ausführen
<i>Import</i>	Ressource ins Repository importieren
<i>Export</i>	Projekt oder Ressource exportieren
<i>Clean-Up X4 Project</i>	Nicht verwendete Dateien auswählen und aus dem Repository löschen

1.4 Prozessbausteine und ihre Verwendung

Am rechten Rand der Zeichnungsfläche befindet sich eine ausklappbare Werkzeugleiste, die *Palette*. Sie enthält Prozessbaustein-Symbole, die Sie per Drag&Drop in die Zeichnungsfläche ziehen können.

Folgende Prozessbausteine stehen für die Modellierung von IT-Prozessen (.wrf) zur Verfügung:

1.4.1 Control-Bausteine

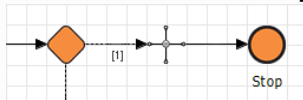
Folgende Control-Bausteine stellen die Basisfunktionen für den Ablauf von Prozessen und deren Kontrollstrukturen bereit:

1.4.1.1 Start und Stop

Die Grundelemente jedes Prozesses sind ein definierter Prozess-Beginn (Start) und mindestens ein Prozess-Ende (Stop).



i Setzen Sie zwischen einem Condition-Baustein und einem direkt darauf folgenden Stop-Baustein einen Verbindungsbaustein, siehe [Connection](#).



Eigenschaften und Parameter

Label	Beschriftung des Symbols im Prozessdiagramm
-------	---

Input und Ausgabe



Sowohl der Start- als auch der Stop-Baustein geben einen ggf. erhaltenen Input des vorherigen Prozess-Schrittes (bei Start den Prozess-Input) ohne Veränderung an den nächsten Prozessbaustein weiter. Der Stop-Baustein gibt außerdem den Status des vorherigen Prozessbausteins unverändert weiter und setzt damit den Ausführstatus des Prozesses.

1.4.1.2 Transition



Transition: Verbindet die einzelnen Prozessbausteine und definiert die Reihenfolge der Prozess-Schritte durch seine Pfeilrichtung.

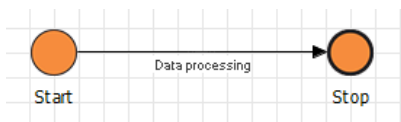
Eigenschaften

Label	Freitextfeld zur Beschriftung des Übergangs (zur übersichtlicheren Gestaltung bzw. Dokumentation)
Log4J Log	<p>Logging über <i>Log4j</i> an dieser Stelle des Prozesses mit den aktuellen Input-Daten aktivieren; Um das Logging über Log4j zu aktivieren, die Eigenschaft markieren und auf  klicken.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Disabled</i>: Keine Log4j-Aktion wurde festgelegt (Standard) • <i>Enabled</i>: Eine Log4j-Aktion wurde festgelegt
BAM Log	<p>Logging über X4 BAM an dieser Stelle des Prozesses mit den aktuellen Input-Daten aktivieren; Um BAM-Logging zu aktivieren, die Eigenschaft markieren und auf  klicken.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Disabled</i>: Keine BAM-Logging-Aktion wurde festgelegt (Standard) • <i>Enabled</i>: Eine BAM-Logging-Aktion wurde festgelegt

Save Point	<p>Save Point an dieser Stelle des Prozesses setzen, um den kompletten Zustand der Prozess-Engine mit den verarbeiteten Daten und Statusinformationen bei der Ausführung einer Prozess-Instanz im Dateisystem zu persistieren. Die Prozess-Instanz kann vom letzten gespeicherten Save Point über die <i>Monitoring-Oberflächen</i> wiederaufgesetzt werden.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>No Savepoint</i>: Kein Save Point wurde festgelegt (Standard) • <i>Save and Continue</i>: Save Point erstellen und die Prozess-Ausführung direkt fortsetzen • <i>Save and Stop</i>: Save Point erstellen und die Ausführung der Prozess-Instanz anhalten • <i>Delete</i>: aktuellen Save Point löschen
Source Component	Baustein, von dem die Übergangslinie beginnt (wird vom <i>X4 Designer</i> gesetzt)
Target Component	Baustein, zu dem die Übergangslinie führt (wird vom <i>X4 Designer</i> gesetzt)
Priority	<p>Bei Gabelungen (<i>Fork</i>) und Verzweigungen (<i>Condition</i>): Position in der Ausführreihenfolge (Priorität)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • <i>1</i>: Höchste Priorität (Standard)
Condition	Bei Bedingungs-Bausteinen: vom Condition Manager hinterlegte Ausführ-Bedingung (wird vom Condition Manager gesetzt)

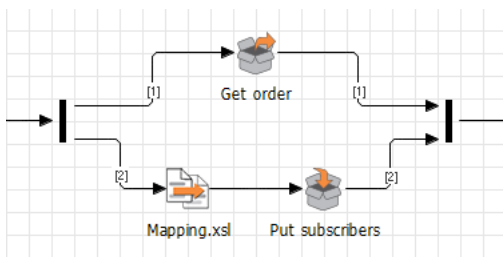
Beispiel

Übergang zwischen zwei Prozessbausteinen mit einer Beschriftung (Label):



1.4.1.3 Fork und Join

Um zwei oder mehrere voneinander unabhängige Ausführungszweige mit demselben Input auszuführen, können Sie eine Gabelung (Fork-Baustein) verwenden. Wieder zusammenführen können Sie die separaten Ausführungszweige mit einem Join-Baustein.



Eigenschaften und Parameter

Rotation	Drehung der Komponente
	Mögliche Werte: <ul style="list-style-type: none"> • 0: Keine Drehung (Standard) • 90: Um 90° gegen den Uhrzeigersinn drehen • 180: Um 180° drehen • 270: Um 270° gegen den Uhrzeigersinn drehen

Input und Ausgabe

Der Fork-Baustein gibt ein Input-Dokument an alle Ausführungszweige unverändert weiter.

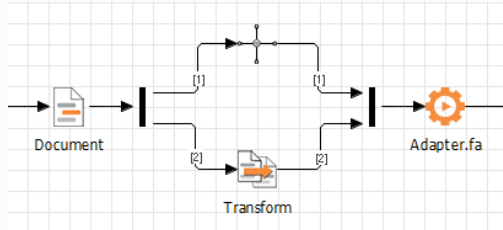
Der Join-Baustein hingegen behält die Ausgabe-Dokumente des letzten ausgeführten Bausteins innerhalb der Gabelung im Speicher. Wenn es sich um XML-Dokumente handelt, dann können sämtliche Ausgabe-XML-Dokumente mit einem Merge-Baustein in einem XML-Dokument zusammengeführt werden. Wenn die Gabelungen jedoch beispielsweise als letzten Baustein vor dem Join ein Binärdokument und ein XSL-Mapping zur dynamischen Parametrierung enthalten, dann kann ein Transfer-Adapter sowohl die dynamischen Parameter als auch den Binärdatenstrom verarbeiten.

Fork und Join verwenden

Ein Ausführungszweig kann aus einem oder mehreren Prozess-Schritten bzw. Bausteinen bestehen.

Die Ausführungszweige innerhalb einer Gabelung werden nicht parallel abgearbeitet, sondern anhand einer Priorität gesteuert, die an den Übergängen durch eine Zahl dargestellt wird. Die Priorität der Ausführungszweige können Sie in den Eigenschaften des Übergangs festlegen.

- ❗ Bei Fork/Join-Konstrukten, in denen in einem Ausführzweig eine Operation ausgeführt wird (z. B. ein XSL-Mapping zur dynamischen Parametrierung eines Adapters) und im anderen Zweig lediglich das Input-Dokument durchgeleitet werden soll, einen Verbindungsbaustein einsetzen, siehe [2019-03-01_16-20-35_Connection \(Verbindungsbaustein\)](#).



⚠ ACHTUNG: Unerwartetes Verhalten der Prozess-Engine!

Wenn Sie innerhalb einer Schleife, die ein Fork- und Join-Element enthält, eine weitere Schleife in Ihrem Prozessdiagramm definieren, dann kann die Prozess-Engine den Status für die bereits abgearbeiteten Prozess-Schritte nicht korrekt zurücksetzen.

Lagern Sie stattdessen die innere Schleife mit sämtlichen Prozessbausteinen als Subprozess aus, siehe [Subprocess](#).

1.4.1.4 Condition

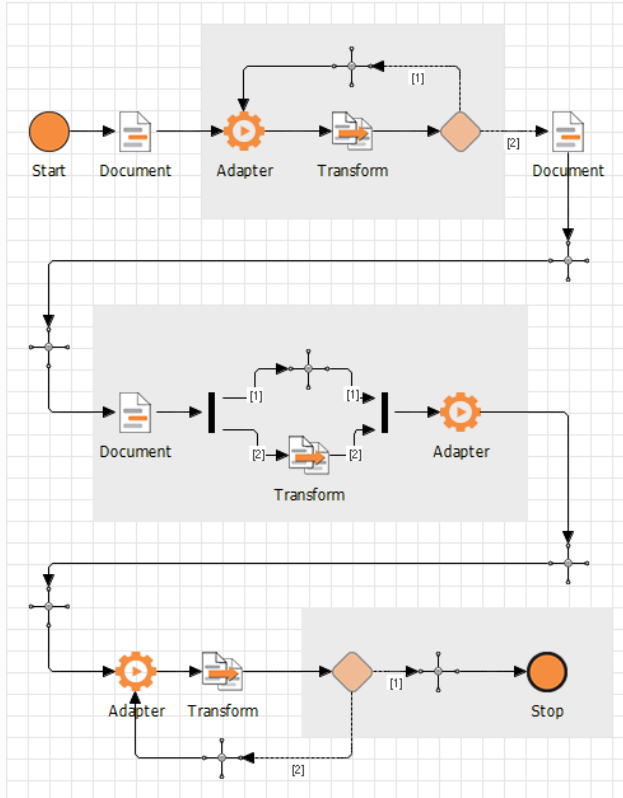


Condition: Trifft Fallunterscheidungen mithilfe von Verzweigungen im Prozess, wobei jeder dahinterliegende Zweig bei einer zutreffenden Bedingung ausgeführt wird.

Eigenschaften

Label	Freitextfeld zur Beschriftung des Condition-Bausteins
Condition	Ausführ-Bedingung (wird automatisch gesetzt); auf die Übergangslinie doppelklicken, um Bedingungen zu hinterlegen
Rotation	<p>Drehung des Condition-Bausteins auf der Zeichnungsfläche in Grad, um je nach Anordnung der Prozessbausteine (auch von rechts nach links oder von unten nach oben) den Ablauf korrekt darzustellen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • 0: keine Drehung (Standard) • 90: um 90° gedreht • 180: um 180° gedreht • 270: um 270° gedreht

- ① Verwenden Sie in Schleifen beim Rücksprung vom Bedingungsbaustein zum ersten Schleifenbaustein sowie zwischen einem Bedingungsbaustein und einem direkt darauf folgenden Stop-Baustein einen Verbindungsbaustein (Connection).



1.4.1.4.1 Bedingungen erstellen und verwenden

Mit dem Condition-Baustein können Sie Verzweigungen im Prozess abbilden, wobei jeder Ausführungszweig eine Bedingung enthalten muss.

Für Fallunterscheidungen können Sie Bedingungen anhand des Statuswertes eines bereits ausgeführten Prozess-Schrittes definieren, anhand von System- oder benutzerdefinierten Platzhalten sowie anhand von Kontextvariablen. Alle Werte werden zur Prozesslaufzeit vom X4 Servers ausgewertet.

Wenn Sie für einen Ausführungszweig keine Bedingung festgelegt haben, wird dieser *immer* ausgeführt. Dies wird durch ein halbtransparentes Symbol des Condition-Bausteins dargestellt. Wenn mehrere Bedingungen zutreffen, werden alle entsprechenden Zweige gemäß deren Priorität nacheinander ausgeführt.


Voraussetzungen

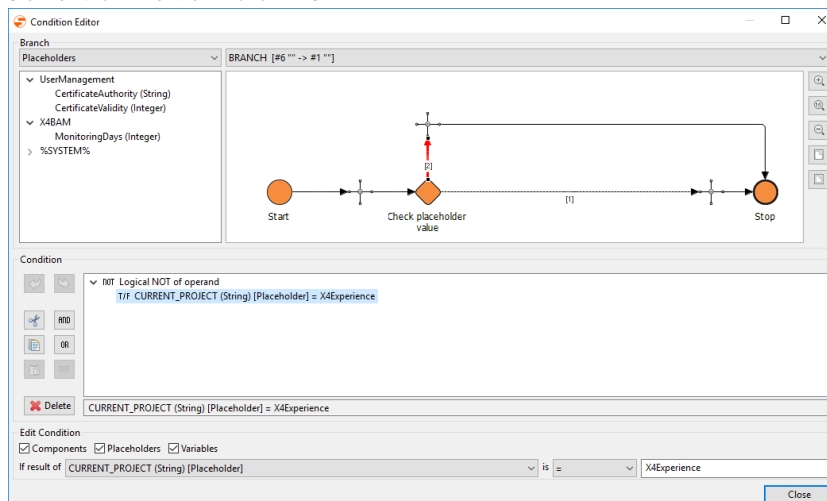
Um den **Condition**-Baustein korrekt nutzen zu können, müssen folgende Voraussetzungen erfüllt sein:

- Ein **Condition**-Baustein muss mindestens zwei Ausführzweige besitzen.
- Ausführzweige dürfen nicht komplett leer sein; Mindestens ein Ausführzweig muss einen Prozessbaustein enthalten.
- Für jeden Ausführzweig muss mindestens eine eindeutige und gültige Bedingung definiert werden.
- Der **Condition**-Baustein sollte sich nicht innerhalb eines Fork/Join-Konstruktes befinden.


1.4.1.4.1 Condition-Baustein in einen Prozess einfügen

Die grundlegende Vorgehensweise, um den **Condition**-Baustein in einen Prozess einzufügen ist wie folgt:

1. **Condition**-Baustein  per Drag&Drop aus der Palette einfügen.
2. Übergangslinien vom davor liegenden Prozessbaustein zum **Condition**-Baustein und für die Ausführungszweige definieren.
3. **Condition**-Baustein oder einen der dahinter liegenden Ausführzweige doppelklicken, um den **Condition Editor** zu öffnen.



4. Im **Condition Editor** mindestens eine Bedingung für jeden Ausführungszweig festlegen. Dazu auf einen Übergang hinter dem **Condition**-Baustein klicken, um einen Ausführungszweig zu wählen. Der gewählte Übergang wird nun rot gestrichelt hervorgehoben. Sie können im **Condition**-Bereich nun Auswertungskriterien wählen (anhand des Ausführstatus, von System- und benutzerdefinierten Platzhaltern sowie anhand von Kontextvariablen, siehe unten).

 Jede Bedingung kann aus mehreren Bedingungelementen bestehen, die Sie über logische Operatoren (AND, OR und NOT) verknüpfen bzw. kombinieren und verschachteln können. Weitere Bedingungelemente können Sie über die entsprechenden Schaltflächen einfügen.

5. **Close** klicken, um die Bedingungen speichern.

⚠️ ACHTUNG: Unerwartetes Verhalten der Prozess-Engine!

Wenn Sie innerhalb einer Schleife, die ein Fork- und Join-Element enthält, eine weitere Schleife in Ihrem Prozessdiagramm definieren, dann kann der X4 Server den Status für die bereits abgearbeiteten Prozess-Schritte nicht korrekt zurücksetzen.

Lagern Sie stattdessen die innere Schleife mit sämtlichen Prozessbausteinen als Subprozess aus, siehe [Subprocess](#).

1.4.1.4.1.2 Bedingungen anhand des Ausführstatus definieren

Mit **Condition**-Bausteinen können Sie anhand des *Statuswertes* eines bereits ausgeführten Prozess-Schrittes eine oder mehrere Bedingungen definieren: So kann beispielsweise überprüft werden, ob ein Iterator-Adapter keine Daten mehr liefert, sodass die Prozessausführung auf einen **Stop**-Baustein umgeleitet und damit beendet wird.

Um eine Bedingung anhand des Ausführstatus zu definieren, gehen Sie wie folgt vor:

1. Im **Condition Editor** auf eine Übergangslinie hinter dem **Condition**-Baustein klicken, um einen Ausführungszweig zu wählen.
2. In **Edit Condition** den Prozessbaustein auswählen, dessen Ausführstatus zur Prozesslaufzeit ausgewertet werden soll.
Hier stehen nur vorangegangene Bausteine des Prozesses zur Auswahl. Alternativ können Sie auch einen Prozessbaustein aus der Prozessvorschau per Klick wählen.
3. Vergleichsoperator (z.B. <, =) auswählen sowie einen ganzzahligen Vergleichswert angeben, auf den in der Bedingung geprüft werden soll, z.B. *1 (ok)*.

Bei Bedarf können Sie auch Bedingungen mit logischen Operatoren (AND, OR, NOT) verknüpfen bzw. verschachteln.

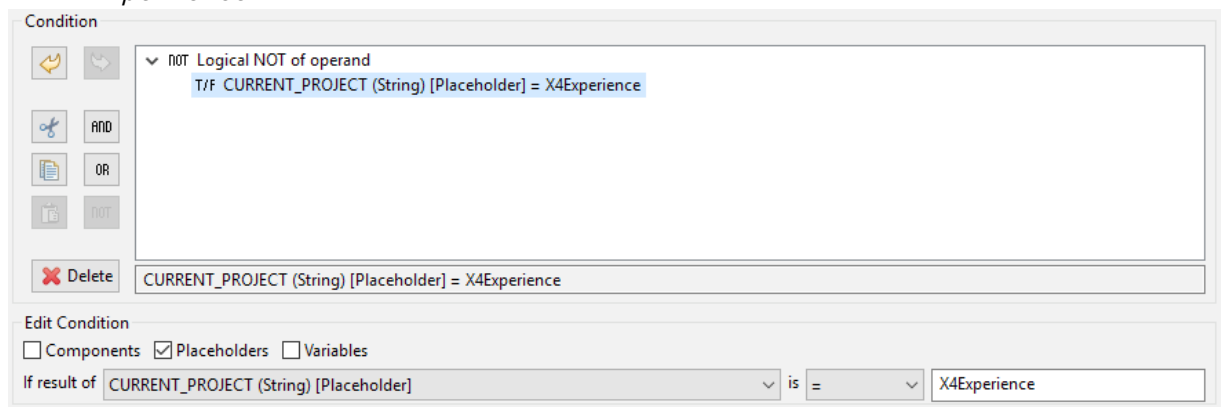
4. **Close** klicken, um die Bedingungen speichern.

1.4.1.4.1.3 Bedingungen anhand von System-Platzhaltern definieren

Mit **Condition**-Bausteinen können Sie anhand von aktuellen Umgebungsbedingungen die Ausführung des technischen Prozesses beeinflussen. Dazu können Sie in Bedingungen über System-Platzhalter wie `%CURRENT_PROJECT%` die aktuellen Werte zur Prozesslaufzeit ermitteln, siehe [System-Platzhalter](#).

Um eine Bedingung anhand eines Platzhalters zu definieren, gehen Sie wie folgt vor:

1. Im **Condition Editor** auf eine Übergangslinie hinter dem **Condition**-Baustein klicken, um einen Ausführungszweig zu wählen.
2. In der Auswahlliste links oben den Eintrag **Placeholders** wählen, um darunter in einer Liste der verfügbaren Platzhaltergruppen und deren Platzhalter anzuzeigen.
3. Platzhalter auswählen, dessen aktueller Wert zur Prozesslaufzeit ausgewertet werden soll, z.B. *CURRENT_PROJECT*.
4. Einen der folgenden Vergleichsoperatoren wählen:
 - = gleich wie ...
 - != ungleich ...
 - contains enthält ...
 - startsWith beginnt mit ...
 - endsWith endet mit ...
5. Einen beliebigen alphanummerischen Vergleichswert angeben, auf den geprüft werden soll, z.B. *X4Experience*.



Bei Bedarf können Sie auch Bedingungen mit logischen Operatoren (AND, OR, NOT) verknüpfen bzw. verschachteln.

6. **Close** klicken, um die hinterlegte Bedingung zu speichern.

1.4.1.4.1.4 Bedingungen anhand von benutzerdefinierten Platzhaltern definieren

Mit **Condition**-Bausteinen können Sie unter bestimmten Voraussetzungen auch benutzerdefinierte Platzhalter für die Definition von Bedingungen verwenden. Wie mit System-Platzhaltern lässt sich somit je nach aktuellen Umgebungsbedingungen die Ausführung des technischen Prozesses beeinflussen.

Um in Bedingungen benutzerdefinierte Platzhalterwerte im **Condition Editor** zu verwenden und zur Prozesslaufzeit auswerten zu können, gehen Sie grundsätzlich wie bei der Definition von Bedingungen mit System-Platzhaltern vor.

❗ Beachten Sie jedoch folgende zusätzlichen Voraussetzungen:

- Der benutzerdefinierte Platzhalter muss bereits im *Custom Placeholder Storage* definiert sein, um im **Condition Editor** ausgewählt werden zu können, siehe [Benutzerdefinierte Platzhalter und -gruppen verwalten](#).
- Der benutzerdefinierte Platzhalter muss zudem auch während der Prozessausführung im *Custom Placeholder Storage* definiert sein, um zur Laufzeit vom X4 Servers ausgewertet werden zu können.
- In den Prozess-Eigenschaften muss in Eigenschaft Placeholder Groups die entsprechende Platzhaltergruppe ausgewählt sein, siehe [Benutzerdefinierte Platzhalter und -gruppen verwalten](#).

1.4.1.4.1.5 Bedingungen anhand von Kontextvariablen definieren

Mit **Condition**-Bausteinen können Sie anhand von aktuellen Kontext-Variablenwerten Bedingungen definieren, welche die Ausführung des technischen Prozesses beeinflussen.

Um eine Bedingung anhand einer Kontext-Variable zu definieren, gehen Sie wie folgt vor:

1. Im **Condition Editor** auf eine Übergangslinie hinter dem **Condition**-Baustein klicken, um einen Ausführungszweig zu wählen.
2. In der Auswahlliste links oben den Eintrag **Variables** wählen, um darunter eine Liste von Kontextvariablen anzuzeigen, die bereits im Prozess definiert wurden.
3. Kontextvariable auswählen, deren aktueller Wert zur Prozesslaufzeit ausgewertet werden soll, z.B. *MyVar*.
4. Einen der folgenden Vergleichsoperatoren wählen:
 - = gleich wie ...
 - != ungleich ...
 - contains enthält ...
 - startsWith beginnt mit ...
 - endsWith endet mit ...
5. Einen beliebigen alphanummerischen Vergleichswert angeben, auf den geprüft werden soll, z.B. *1*.


The screenshot shows the 'Condition Editor' window. The main editing area contains a logical expression: 'NOT Logical NOT of operand' with a sub-expression 'T/F MyVar (String) [Variable] contains 1'. Below the main area, there is a 'Delete' button and the text 'MyVar (String) [Variable] contains 1'. At the bottom, the 'Edit Condition' section has three checkboxes: 'Components', 'Placeholders', and 'Variables' (which is checked). Below these checkboxes, there is a dropdown menu showing 'If result of MyVar (String) [Variable]', followed by 'is contains' and the value '1'.

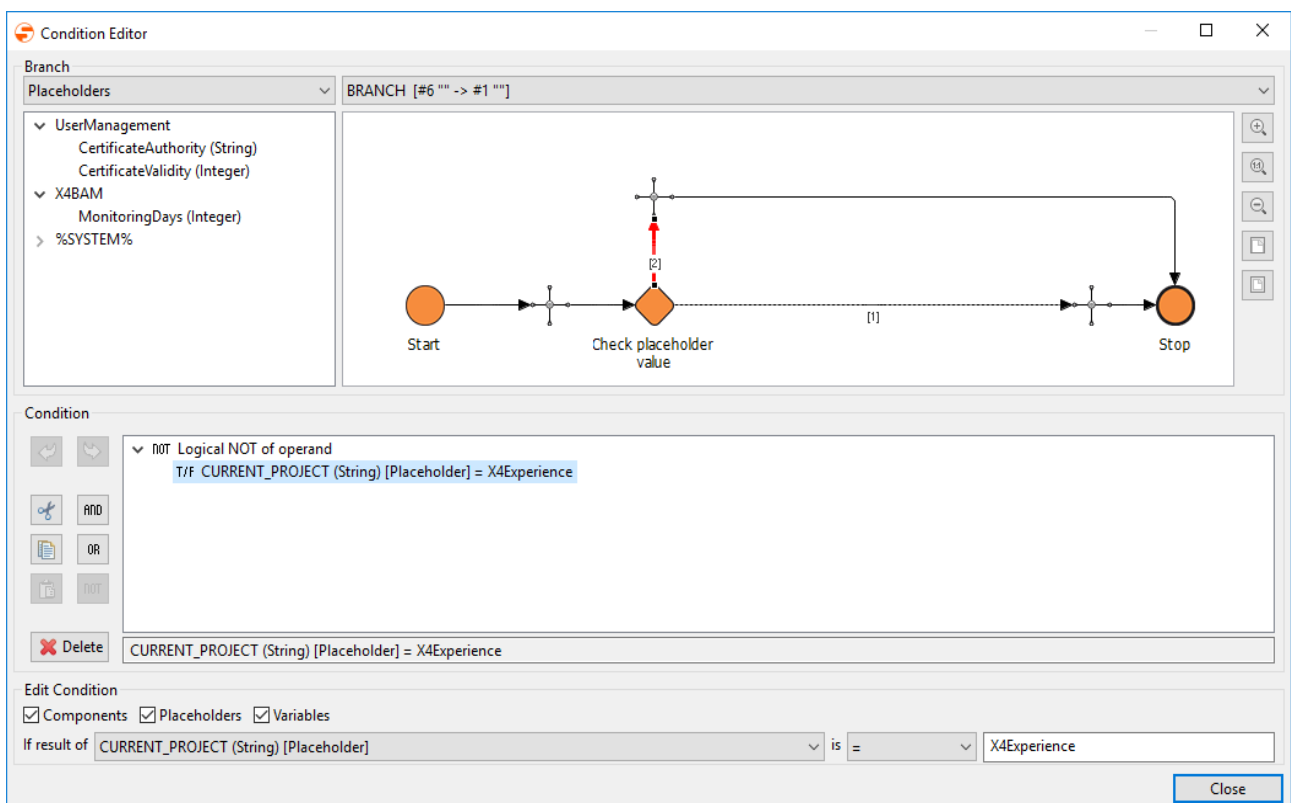
Bei Bedarf können Sie auch Bedingungen mit logischen Operatoren (AND, OR, NOT) verknüpfen bzw. verschachteln.

6. **Close** klicken, um die hinterlegte Bedingung zu speichern.

1.4.1.4.2 Condition Editor

Mit dem **Condition Editor** können Sie für *Condition*-Bausteine innerhalb eines technischen Prozesses festlegen, wann verschiedene Verzweigungen ausgeführt werden sollen. Für jede Verzweigung lassen sich Bedingungen definieren, die Statuswerte von vorangehenden Prozessbausteinen, Kontextvariablenwerte oder Platzhalterwerte (benutzerdefinierte oder Systemplatzhalter) zur Prozesslaufzeit auswerten.

Um den **Condition Editor** aufzurufen, öffnen Sie einen Prozess und doppelklicken Sie darin auf ein *Condition*-Symbol  oder auf einen der Ausführungswege hinter dem Condition-Baustein. Im **Condition Editor** wird eine vereinfachte Ansicht des Prozessdiagramms ohne Zeichnungselemente angezeigt:



Rechte Seite des Branch-Bereichs

Im Bereich **Branch** wählen Sie zunächst auf der rechten Seite über die Auswahlliste den Zweig, für den Sie eine Bedingung festlegen möchten. Alternativ können Sie in der Prozessansicht auch die gewünschte Übergangslinie anklicken, um für diesen Zweig eine Bedingung zu definieren.

Ansichtsoptionen für die Prozessdarstellung:

	Zoom in	Ansicht vergrößern
	Original zoom	Ursprüngliche Größe anzeigen
	Zoom out	Ansicht verkleinern

	Zoom to page	Alles anzeigen
	Zoom to width	Ansicht auf voller Breite

Linke Seite des Branch-Bereichs

Auf der linken Seite können Sie über eine Auswahlliste zusätzliche Kriterien zur Definition von Bedingungen wählen. Im Bereich darunter werden gemäß Ihrer Auswahl alle verfügbaren Einträge angezeigt. Als Kriterien stehen folgende Optionen zur Auswahl:









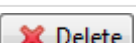
- **Placeholders:** Eine Liste der aktuell verfügbaren Platzhaltergruppen und deren Platzhalter (System- und benutzerdefinierte Platzhalter, sofern letztere bereits im [Placeholder Storage](#) registriert wurden)
- **Variables:** Eine Liste der Kontextvariablen, die im aktuellen Prozess bereits definiert wurden

Condition-Bereich

Im Bereich **Condition** werden die Bedingungen für den ausgewählten Zweig angezeigt und bearbeitet.

Um eine neue Bedingung zu erstellen oder zu bearbeiten, doppelklicken Sie in den **Condition**-Bereich.

Bearbeitungsoptionen für Bedingungen:

	Undo	Rückgängig machen
	Redo	Rückgängig gemachten Schritt wiederherstellen
	Cut	Auswahl ausschneiden
	Copy	Auswahl kopieren
	Paste	Auswahl einfügen
	Logical AND	Logische UND-Verknüpfung einfügen
	Logical OR	Logische ODER-Verknüpfung einfügen
	Logical NOT	Logische NICHT-Verknüpfung einfügen
	Delete	Markierte Bedingung löschen

Bereich Edit Condition

Im Bereich **Edit Condition** können Sie die gewählte Bedingung erstellen oder bearbeiten, siehe [Bedingungen erstellen und verwenden](#).

Components	Option, ob in der Auswahlliste <i>If result of</i> Prozessbausteine angezeigt werden sollen, um deren Ausführstatus zur Prozesslaufzeit auszuwerten.
Placeholders	Option, ob in der Auswahlliste <i>If result of</i> eine Liste aller aktuell verfügbaren Platzhalter (System- und ggf. benutzerdefinierte Platzhalter) angezeigt werden sollen, um deren Werte zur Prozesslaufzeit auszuwerten.
Variables	Option, ob in der Auswahlliste <i>If result of</i> eine Liste aller im aktuellen Prozess bereits definierten Kontextvariablen angezeigt werden, um deren Variableninhalt zur Prozesslaufzeit auszuwerten.
If result of	In dieser Auswahlliste wählen Sie das Kriterium (Prozessbaustein, Platzhalter oder Variable) für Ihre Bedingung.
is	<p>Hier wählen Sie einen Vergleichsoperator:</p> <ul style="list-style-type: none"> • < kleiner als ... • <= kleiner als oder gleich wie ... • = gleich wie ... • >= größer als oder gleich wie ... • > größer als ... • <> ungleich zu ... <p>Für Platzhalter und Variablen stehen hier folgende Vergleichsoperatoren für Zeichenketten zur Verfügung:</p> <ul style="list-style-type: none"> • = gleich wie ... • != ungleich ... • contains enthält ... • startsWith beginnt mit ... • endsWith endet mit ...
(Vergleichswert)	<p>In der Auswahlliste ganz rechts geben Sie den Wert an, der mit dem ausgewählten Kriterium verglichen werden soll.</p> <p>Bei Prozessbausteinen ist dies deren möglicher Ausführstatuswert:</p> <ul style="list-style-type: none"> • error: Wenn ein Fehler auftritt (Status -1) • ok: Wenn kein Fehler auftritt (Status 1) • empty: Wenn das Ergebnis eine leere Menge ist (Status 0) • ±n: Beliebiger ganzzahliger Wert <p>Bei Kontextvariablen und Platzhaltern ist dies ein alphanummerischer Vergleichswert.</p>

1.4.1.5 Connection



Connection: Gibt den Input aus dem vorangegangenen Prozess-Schritt ebenso wie dessen Status unverändert aus. Es markiert den jeweiligen Ausführungszweig für den X4 Server eindeutig. So lässt sich mit diesem Baustein in Schleifen der Rücksprung zum ersten Schleifenbaustein explizit setzen. Es verbindet einen Condition-Baustein mit einem Stop-Baustein oder ein Fork-Baustein mit einem direkt darauf folgenden Join-Baustein.

Eigenschaften

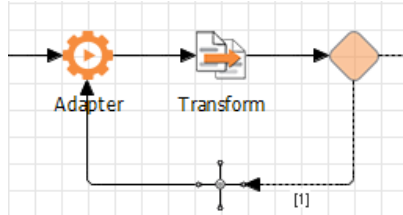
Label	Beschriftung des Symbols im Prozess-Diagramm
-------	--

Output

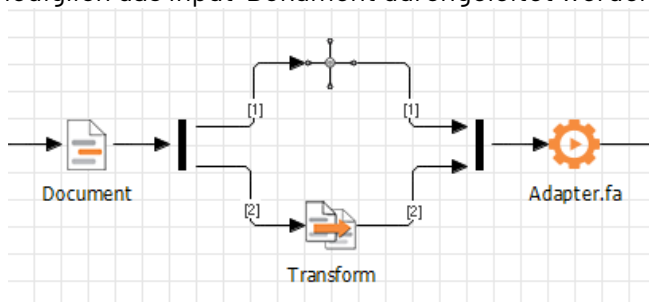
Der Verbindungsbaustein gibt den Input unverändert aus, inklusive Ausführstatus des vorherigen Prozessbausteins.

Beispiele

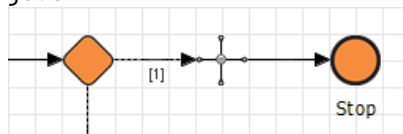
- Bei Schleifenrücksprüngen wird als letzter Schleifenbaustein ein Verbindungsbaustein gesetzt.



- Bei Fork/Join-Konstrukten, in denen in einem Ausführzweig eine Operation ausgeführt wird (z. B. ein XSL-Mapping zur dynamischen Parametrierung eines Adapters) und im anderen Zweig lediglich das Input-Dokument durchgeleitet werden soll, wird ein Verbindungsbaustein gesetzt.



- Zwischen einem Condition-Baustein und einem Stop-Baustein wird ein Verbindungsbaustein gesetzt.



1.4.1.6 Pause



Pause: Definiert Wartezeiten innerhalb eines Prozesses. Nach einer festgelegten Dauer oder zu einem bestimmten Zeitpunkt wird die Ausführung des Prozesses fortgesetzt. Dabei kann ein Pause-Baustein einen Input vom vorangegangenen Prozessbaustein erhalten und diesen an den darauf folgenden Prozessbaustein unverändert weitergeben.

Eigenschaften

Label	Beschriftung des Symbols im Prozessdiagramm
Duration	<p>Dauer der Pause</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige Dauer mit Angabe von Tagen, Stunden, Minuten, Sekunden und Millisekunden • <code>9999d 23h 59m 59s 999ms</code>: Maximale Wartezeit pro Pause-Baustein • <code>0d 0h 0m 0s 0ms</code>: Keine Wartezeit (Standard) <p>i Alternativ zur Angabe einer Wartezeit lässt sich über <code>Until</code> ein Endzeitpunkt im Format <code>yyyy-MM-dd hh:mm:ss</code> für die Pause festlegen, z. B. <code>2010-08-30T01:55:10</code>.</p>

- i** Um die Pausendauer zu definieren, den Pause-Baustein doppelklicken oder in der **Properties**-Sicht unter `Duration` auf klicken. Anschließend im Dialog, die Dauer definieren.

The dialog box titled "Duration" contains a "Configure Interval" section. It has two radio buttons: "Duration" (selected) and "Until".

Under "Duration", there are input fields for:

- Duration: 0
- Days: 0
- Hours: 0
- Minutes: 0
- Seconds: 0
- Milliseconds: 0

Under "Until", there are two input fields:

- Date: 07.04.2020
- Time: 08:48:16

At the bottom right, there are "OK" and "Cancel" buttons.

1.4.1.6.1 Pause-Baustein dynamisch parametrieren

Der Pause-Baustein lässt sich auch dynamisch über die technischen Parameter `sleep_time` und `sleep_until` parametrieren.

- **Dynamische Parametrierung über `sleep_time`**

Im folgenden Beispiel wird der Pause-Baustein auf eine Pausendauer von 5 Sekunden

parametriert:

Dynamische Parametrierung über sleep_time

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="2.0">
  <xsl:output method="xml" media-type="text/xml"></xsl:output>
  <xsl:template match="/">
    <!-- Pause duration 5 seconds / used format "0(d):0(h):0(m):0(s).0(ms)"-->
    <xsl:processing-instruction name="DynamicParameter">
      <xsl:text>sleep_time="0:0:0:5.0"</xsl:text>
    </xsl:processing-instruction>
    <Empty></Empty>
    <!-- <xsl:processing-instruction name="NoResult"></xsl:processing-
instruction> -->
  </xsl:template>
</xsl:stylesheet>
```

- **Dynamische Parametrierung über sleep_until**

Im folgenden Beispiel wird der Pause-Baustein auf das aktuelle Datum und die aktuelle Uhrzeit zuzüglich 1 Minute und 15 Sekunden parametriert:

Dynamische Parametrierung über sleep_until

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" xmlns:xs="http://www.w3.org/2001/XMLSchema" version="2.0">
  <xsl:output method="xml" media-type="text/xml"></xsl:output>
  <xsl:template match="/">
    <xsl:processing-instruction name="DynamicParameter">
      <xsl:text>sleep_until="</xsl:text>
      <xsl:value-of select="format-dateTime(current-dateTime()+
xs:dayTimeDuration('PT0H1M15S'),'[Y0001]-[M01]-[D01] [H10]:[m01]:[s01]')"></xsl:
value-of>
      <xsl:text>"</xsl:text>
    </xsl:processing-instruction>
    <!-- <Empty></Empty> -->
    <xsl:processing-instruction name="NoResult"></xsl:processing-instruction>
  </xsl:template>
</xsl:stylesheet>
```

1.4.1.7 Counter



Counter: Ermöglicht die Implementierung von Zählschleifen in Prozessen. Hierfür wird in Eigenschaft **Start** ein Anfangswert und in Eigenschaft **End** ein Endwert gesetzt. Bei jedem Durchlauf des Zählers wird dessen Wert um die in Eigenschaft **Step Size** angegebenen Wert verringert bzw. erhöht, bis der Endwert erreicht ist. Über ein XSLT-Mapping lassen sich die Eigenschaften dynamisch

setzen und der aktuelle Index kann über `indexName` in eine Variable gespeichert werden.

Eigenschaften

Start Dynamischer Parameter: <code>start</code>	Anfangswert von dem heruntergezählt bzw. hochgezählt wird Mögliche Werte: beliebige ganze Zahl, die größer 0 ist
End Dynamischer Parameter: <code>end</code>	Endwert bis zu dem heruntergezählt bzw. hochgezählt wird Mögliche Werte: beliebige ganze Zahl, die größer 0 ist; Ist der Endwert kleiner als der Startwert, wird heruntergezählt. Andernfalls wird hochgezählt.
Step Size Dynamischer Parameter: <code>step</code>	Zählschritte für das Hochzählen bzw. Herunterzählen, z. B. 2 Mögliche Werte: beliebige ganze Zahl, die größer 0 ist
Label	Beschriftung des Symbols im Prozessdiagramm
Index Name Dynamischer Parameter: <code>indexName</code>	Name der Variablen Mögliche Werte: Name der Variablen, um auf den aktuellen Index des Zählers zuzugreifen

Input

Der Baustein erwartet nur dann einen Input, wenn die Eigenschaften dynamisch gesetzt werden.

Ausgabe

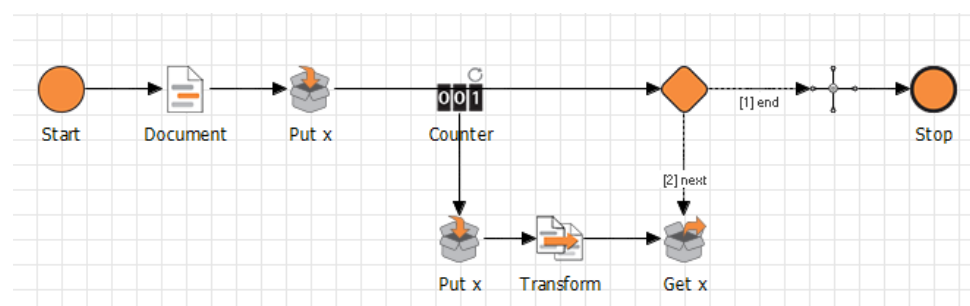
Wird die Eigenschaft dynamisch gesetzt, gibt der Baustein eine Variable mit `<indexName>` und den Wert des aktuellen Index aus. Andernfalls liefert der Baustein keinen Output.

Statuswerte

0 (counting)	Zähler läuft (Endwert wurde noch nicht erreicht)
1 (finished)	Herunterzählen ist beendet (Endwert wurde erreicht)

Beispiel

Einfache Zählschleife mit Zähler- und Bedingungs-Baustein (Counter und Condition):



1.4.1.8 Transaction



Transaction: Führt Prozess-Schritte transaktionssicher durch: Bei einem *Commit* werden sämtliche Operationen auf dem Zielsystem persistent ausgeführt; bei einem *Rollback* werden alle Operationen auf den Ursprungszustand vor der Transaktion zurückgesetzt. Um Prozess-Schritte in eine Transaktion zu kapseln, setzen Sie vor und hinter die Prozess-Schritte die entsprechenden Transaktions-Bausteine. Der Transaktions-Baustein wird hauptsächlich für JMS-Warteschlangen-Operationen und für Datenbank-Operationen verwendet.

i Beachten Sie folgende Hinweise:

- Transaktionen können nicht im Debug-Modus des X4 Designers durchgeführt werden.
- Innerhalb von Transaktionen können auch Subprozesse ausgeführt werden.
- Wenn innerhalb einer Transaktion ein Subprozess über den [Process Starter](#) Adapter gestartet wird, dann gilt die Transaktion nicht für diesen Subprozess (Transaktionen sind nicht Thread-übergreifend).
- Transaktionen in Transaktionen zu verschachteln ist nicht möglich.
- Innerhalb einer Transaktion können mehrere Datenbankverbindungen und Warteschlangen verwendet werden.
- Die verwendete Datenbankverbindung (JDBC DataSource) muss Transaktionen unterstützen.
- Um Transaktionen ausführen zu können, darf bei der Konfiguration der zu verwendenden Datasource innerhalb der `standalone.xml` das Attribut `jta` nicht auf `false` gesetzt werden.

Eigenschaften

Label	Beschriftung des Symbols im Prozessdiagramm
Operation	Transaktions-Operation festlegen Mögliche Werte: <ul style="list-style-type: none"> • <i>Not assigned</i>: noch keine Transaktion festgelegt • <i>Begin</i>: Transaktion beginnen • <i>Commit</i>: Transaktion durchführen/abschließen • <i>Rollback</i>: Transaktion abbrechen und innerhalb der Transaktion durchgeführte Operationen rückgängig machen

Statuswerte

1	Transaktionsbaustein wurde erfolgreich ausgeführt
0	Transaktionsbaustein wurde ohne Ergebnis ausgeführt
-1	Während der Transaktion trat ein Fehler auf

1.4.2 Data-Bausteine

Folgende Daten-Bausteine speichern Daten temporär oder permanent und führen sie innerhalb eines Prozesses zusammen:

1.4.2.1 XML



XML: Bindet wohlgeformte XML-Daten in den Prozess ein. Dabei wird die Zeichenkodierung aus der XML-Deklaration und der MIME-Typ aus der Dateinamensendung ermittelt.

Eigenschaften

Document	Pfad zum verknüpften XML-Dokument im Repository
Label	Beschriftung des Symbols im Prozessdiagramm
Operation	<p>Bestimmt, ob die Datei gelesen oder geschrieben werden soll</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Read</i>: XML-Dokument in den Prozess einlesen und Daten an den nächsten Prozess-Schritt weitergeben (Standard) • <i>Write</i>: XML-Dokument mit den Daten aus dem vorangegangenen Prozess-Schritt speichern

XML-Dokument erstellen und verwenden

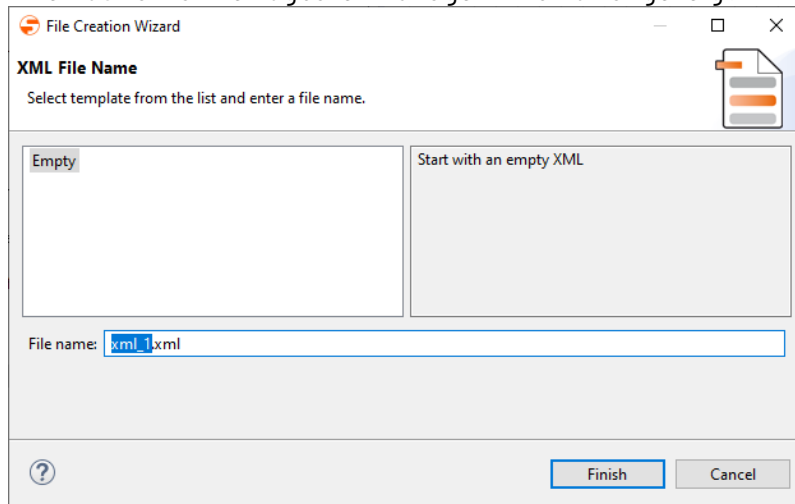
XML-Dokumente lassen sich innerhalb eines ESB-Projektes im Ordner **Resources** anlegen. Hierzu stehen zwei Möglichkeiten zur Verfügung:

- über das Menü **Project > Add XML Document**
- über das Kontextmenü **New > XML Document**

1. Innerhalb des ESB-Projektes auf den Ordner **Resources** rechtsklicken, um das Kontextmenü aufzurufen.

2. Im Kontextmenü **New> XML Document** aufrufen.

Eine Auswahl an verfügbaren Vorlagen wird nun angezeigt.



3. Leere Vorlage oder eine vorhandene Vorlage wählen.
4. In **File Name** den Dateinamen eingeben.
5. **Finish** klicken, um das XML-Dokument anzulegen.
Das XML-Dokument wurde angelegt und wird im [XML-Editor](#) geöffnet.
6. XML-Dokument wie gewünscht bearbeiten.
7. XML-Dokument per Drag&Drop aus dem Repository Navigator ins Prozessdiagramm ziehen.
Beispiel: Zwischen Funktions-Adapter und Stop-Baustein einfügen.
8. In der **Properties**-Sicht in Operation festlegen, ob das XML-Dokument gelesen (*Read*) oder geschrieben (*Write*) werden soll.

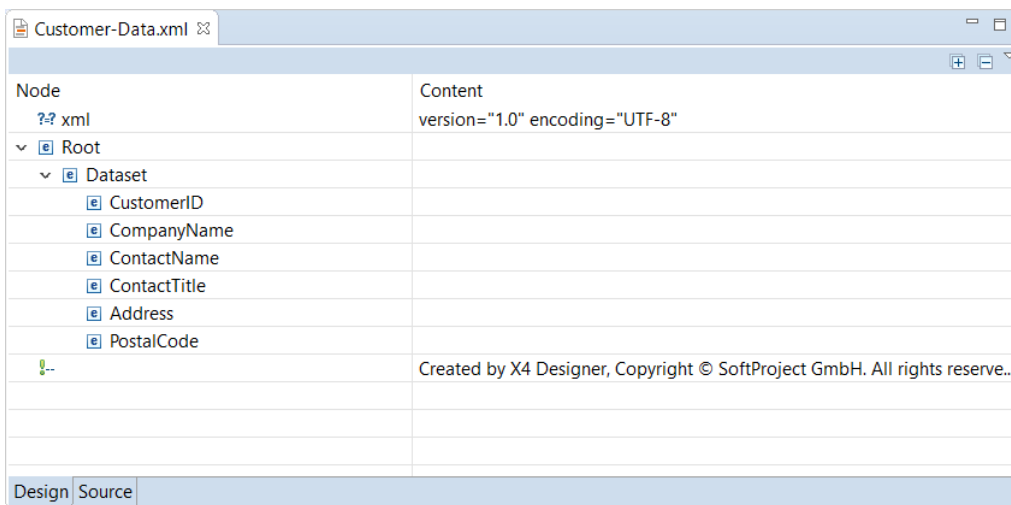
1.4.2.1.1 XML-Editor

Mit dem XML-Editor können Sie wohlgeformte XML-Dokumente erstellen und bearbeiten.

-  Um den XML-Editor zu starten, legen Sie entweder ein neues XML-Dokument an oder öffnen Sie ein bereits vorhandenes Dokument.

Aufbau des XML-Editors

Zum Bearbeiten von XML-Dokumenten stehen zwei Ansichten zur Verfügung, die **Design**-Ansicht und die **Source**-Ansicht, zwischen denen Sie über die Registerkarten am unteren Rand des Editors wechseln können.



Design-Ansicht

In der **Design**-Ansicht wird die Baumstruktur des geöffneten XML-Dokuments tabellarisch dargestellt. In der linken Spalte befinden sich die verschiedenen Knoten, die Sie auf- und zuklappen können, sofern Kindknoten existieren. In der rechten Spalte werden die Inhalte der Knoten angezeigt.

Das Kontextmenü bietet folgende Bearbeitungsfunktionen:

- **Remove:** Markierten Knoten entfernen
- **Add DTD Information:** Strukturdefinition in Form einer DTD (*Document Type Definition*) hinzufügen
- **Edit Namespaces:** Namensräume bearbeiten
- **Add Attribute:** Neues Attribut hinzufügen
- **Add Child:** Neuen Kindknoten (Element, Processing Instruction etc.) hinzufügen
- **Add Before:** Neuen Knoten vor dem markierten Knoten einfügen
- **Add After:** Neuen Knoten hinter dem markierten Knoten einfügen

Source-Ansicht


In der **Source** -Ansicht können Sie den Quelltext von XML-Dokumenten bearbeiten. Dabei unterstützt Sie der Editor durch folgende Funktionen:

- **Syntax-Highlighting:** Farbliche Hervorhebung der verschiedenen Knotentypen
- **Wohlgeformtheit:** Eine Überprüfung, ob das XML-Dokument wohlgeformt ist, erfolgt beim Speichern.
- **Eingabehilfe:** Der Editor macht Vorschläge, wenn Sie neue Elemente einfügen möchten.
- **Pretty-Print:** XML-Dokument mit Einrückungen und Zeilenumbrüchen lesbarer formatieren über das Menü **Source > Format**.

1.4.2.2 Document



Document: Bindet beliebige nicht-XML-Daten, beispielsweise ASCII-Text oder Binär-Formate, in den Prozess ein. Dabei wird versucht, den MIME-Typ aus der Dateinamensendung zu ermitteln. Als Zeichenkodierung wird die System-Zeichenkodierung verwendet.

 Wenn Zeichenkodierung und MIME-Typ nicht korrekt ermittelt werden, dann verwenden Sie den [Document Properties Changer](#), um diese entsprechend zu setzen.

Eigenschaften

Document	Pfad zur verknüpften Text- oder Binär-Datei im Repository
Label	Beschriftung des Symbols im Prozessdiagramm
Operation	Bestimmt, ob die Datei gelesen oder geschrieben werden soll Mögliche Werte: <ul style="list-style-type: none"> • <i>Read</i>: Datei in den Prozess einlesen und Daten an den nächsten Prozess-Schritt weitergeben (Standard) • <i>Write</i>: Datei mit den Daten aus dem vorangegangenen Prozess-Schritt speichern

1.4.2.3 JSON Document



JSON document: Bindet JSON-Dokumente in den Prozess ein.

Eigenschaften

Document	Pfad zur verknüpften JSON-Datei im Repository
Label	Beschriftung des Symbols im Prozessdiagramm
Operation	Bestimmt, ob die Datei gelesen oder geschrieben werden soll Mögliche Werte: <ul style="list-style-type: none"> • <i>Read</i>: Datei in den Prozess einlesen und Daten an den nächsten Prozess-Schritt weitergeben (Standard) • <i>Write</i>: Datei mit den Daten aus dem vorangegangenen Prozess-Schritt speichern

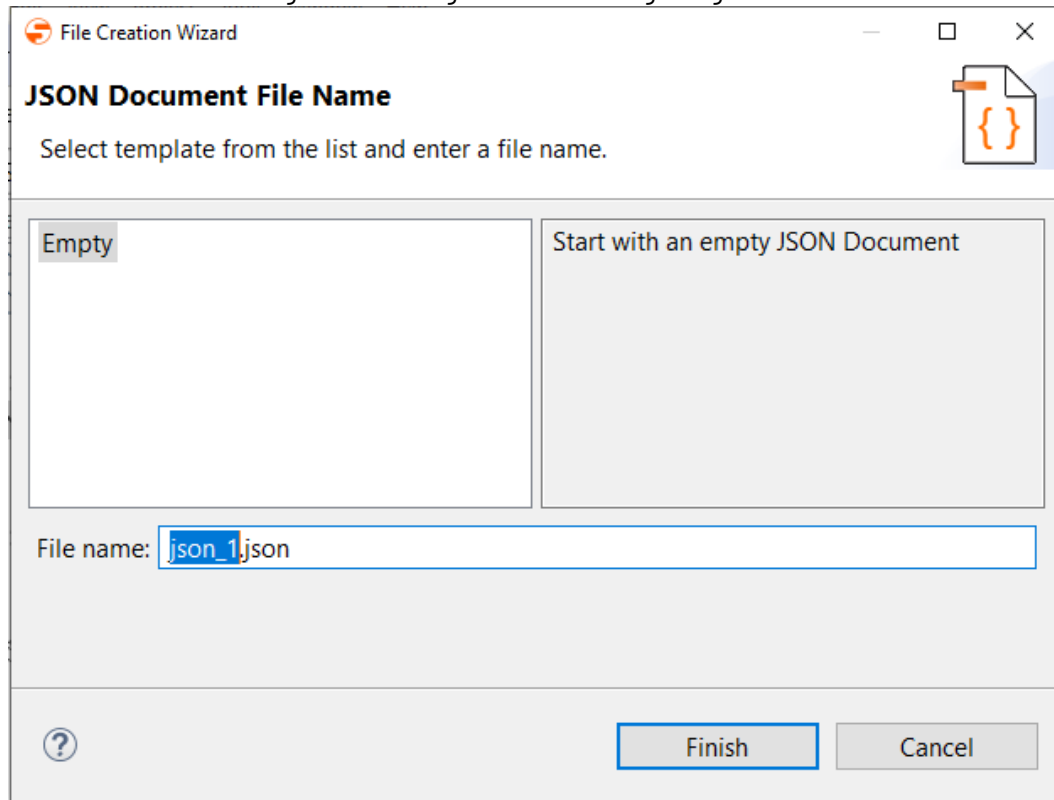
JSON-Dokument erstellen

JSON-Dokumente lassen sich innerhalb eines ESB-Projektes im Ordner Resources anlegen. Hierzu stehen verschiedene Möglichkeiten zur Verfügung:

- über das Menü **Project > Add JSON Document**
- über das Kontextmenü **New > JSON Document**
- über das Menü **New > Add JSON Document**

Um ein JSON-Dokument anzulegen, gehen Sie wie folgt vor:


1. Innerhalb des ESB-Projektes auf den Ordner **Resources** rechtsklicken, um das Kontextmenü aufzurufen.
2. Im Kontextmenü **New > JSON Document** aufrufen.
Eine Auswahl an verfügbaren Vorlagen wird nun angezeigt.



3. Leere Vorlage oder eine vorhandene Vorlage wählen.
4. In **File Name** den Dateinamen eingeben.
5. **Finish** klicken, um das JSON-Dokument anzulegen.
Das JSON-Dokument wurde angelegt und wird im [JSON-Editor](#) geöffnet.

1.4.2.3.1 JSON-Editor

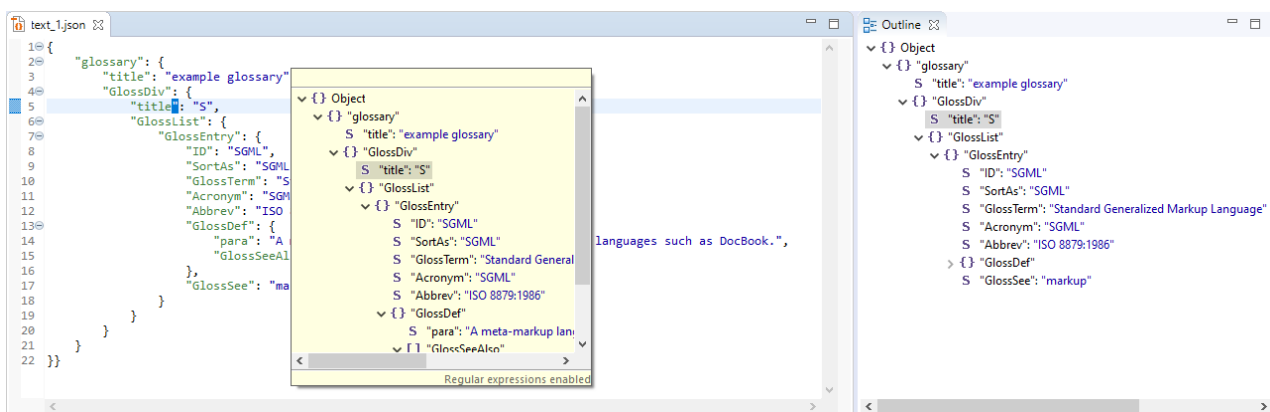
Mit dem JSON-Editor lassen sich JSON-Dokumente direkt im X4 Designer bearbeiten.

 Um den JSON-Editor zu starten, legen Sie entweder ein neues [JSON-Dokument](#) an oder öffnen Sie ein bereits vorhandenes Dokument.

In der **Outline**-Sicht wird die Struktur des geöffneten JSON-Dokuments angezeigt. Zudem kann über das Kontextmenü **Quick Outline** oder über die Tastenkombination **STRG+O** die Quick-Outline-Sicht geöffnet werden, um nach Schlüsseln und Werten innerhalb des JSON-Dokuments zu suchen.

Tastenkombinationen

STRG+O	Quick-Outline-Sicht öffnen
STRG+Umschalttaste+F	JSON-Code formatieren



- ✓ Die Formatierung des JSON-Codes sowie die Farben zur Syntax-Hervorhebung lassen sich unter **Tools > Options > JSON Editor** ändern, siehe auch JSON-Editor konfigurieren.

1.4.2.4 SQL Statement



SQL-Statement : Enthält ein oder mehrere SQL-Statements und dient als Input für einen Datenbankverbindungs-Baustein. Bei einem SQL-Statement-Baustein handelt es sich um ein XML-Dokument mit einer spezifischen Struktur.

Eigenschaften

Document	Pfad zur verknüpften SQL-Statement-Datei im Repository
Label	Beschriftung des Symbols im Prozess-Diagramm
Operation	<p>Bestimmt die Operation für die Datei</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> <p><i>Read</i></p> <p>: SQL-Statement in den Prozess einlesen und an den nächsten Prozess-Schritt (in der Regel ein Datenbankverbindungs-Baustein) weitergeben</p>

- ✓ **Siehe auch:**
 - Datenbank-Editor
 - SQL-Statements erstellen und mit einer Datenbank verknüpfen

1.4.2.4.1 SQL-Statements erstellen und mit einer Datenbank verknüpfen

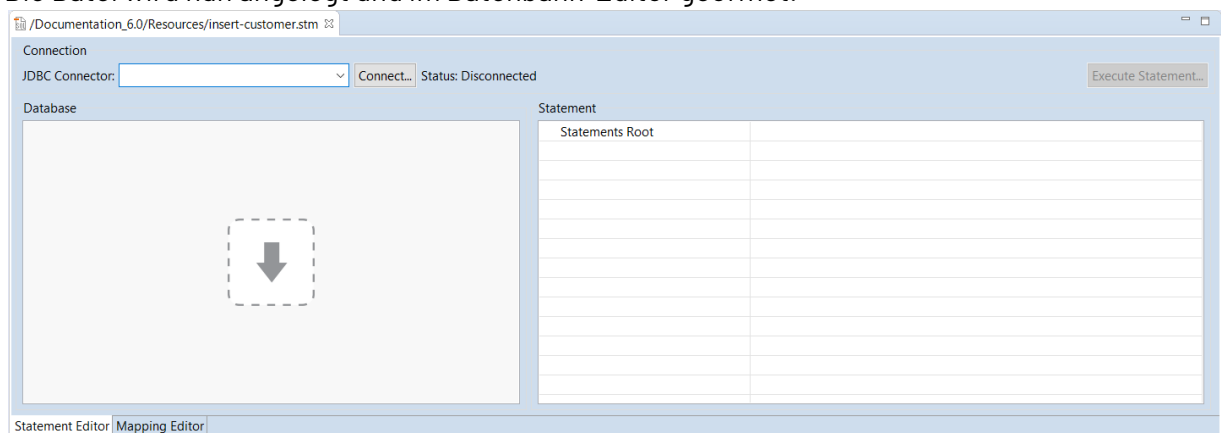
Um in einem Prozess eine Verbindung zu einer relationalen Datenbank aufzubauen, sind zwei Prozessbausteine erforderlich.

In einem SQL-Statement-Baustein definieren Sie Datenbank-Statements. Diese SQL-Abfragen werden in einer spezifischen XML-Struktur im Prozessbaustein gespeichert und an den darauffolgenden Prozessbaustein, einen Datenbankverbindungs-Baustein weitergegeben. Mit diesem wird eine Verbindung zur Datenbank aufgebaut. Dies erfolgt über den [JDBC Connector](#).

SQL-Statement-Datei anlegen

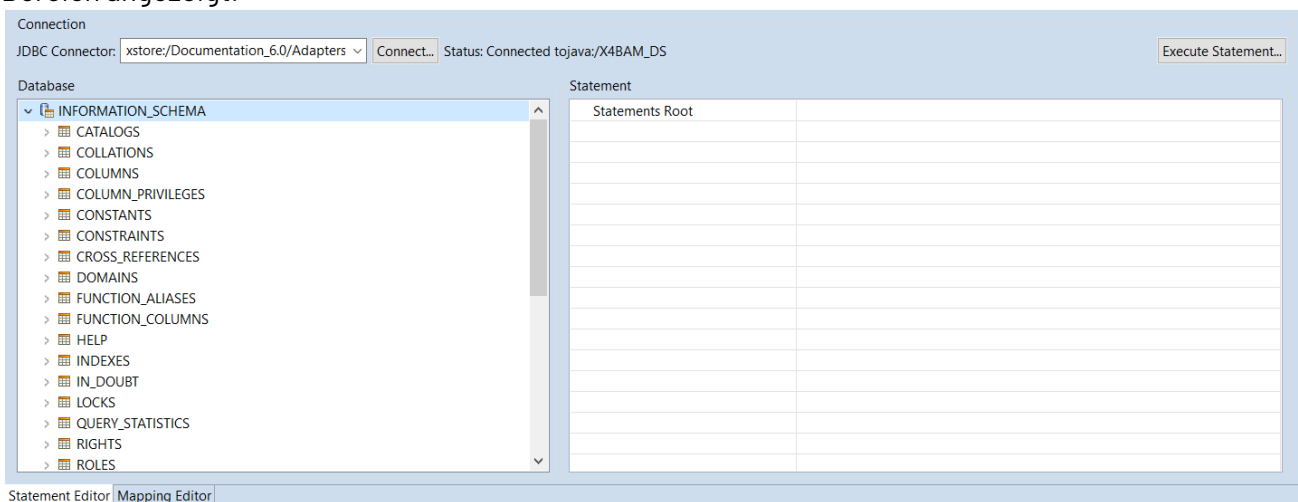
1. Innerhalb des ESB-Projektes auf den Ordner Resources rechtsklicken, um das Kontextmenü aufzurufen.
2. Im Kontextmenü **New> SQL Statement** aufrufen.
3. In File name Dateinamen eingeben.
4. **Finish** klicken, um die SQL-Statement-Datei anzulegen.

Die Datei wird nun angelegt und im Datenbank-Editor geöffnet.



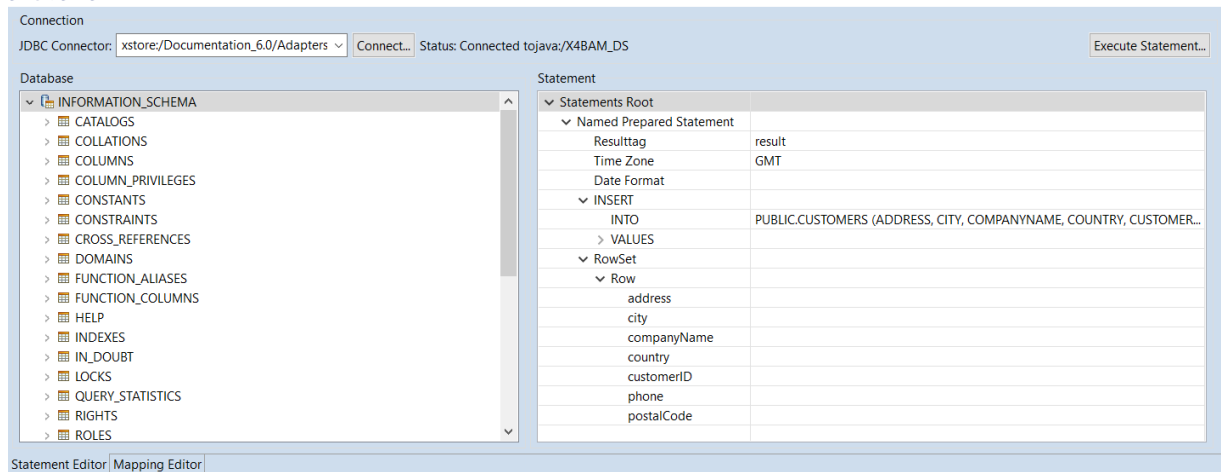
Verbindung zur Datenbank erstellen

Einen bereits gültig parametrisierten [JDBC Connector](#) in den Database-Bereich ziehen. Die Verbindung zur Datenbank wird nun aufgebaut und die Struktur der Datenbank wird im Database-Bereich angezeigt.

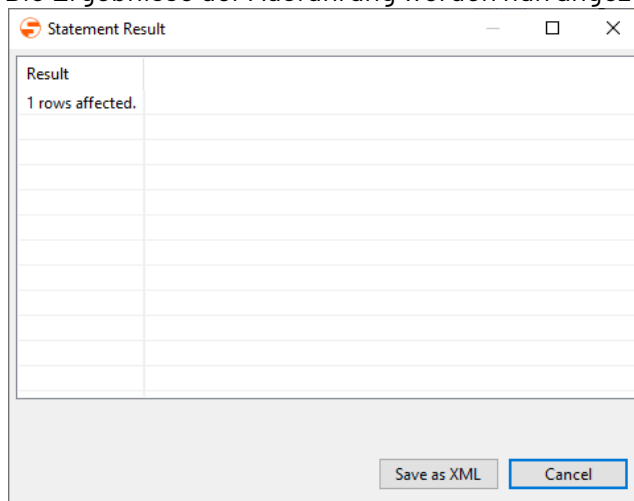


Statements erstellen und ausführen

1. Gewünschte Statements erstellen, siehe auch [Statement für eine verbundene Datenbank erstellen](#).



2. **Execute Statement** klicken, um die SQL-Statements auszuführen.
Die Ergebnisse der Ausführung werden nun angezeigt.

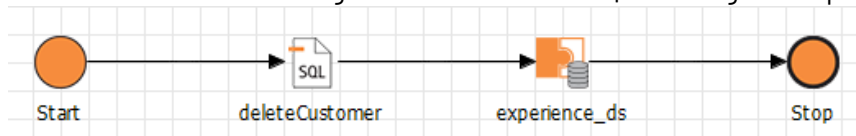


3. Bei Bedarf, mit **Save as XML**, das Ergebnis der Ausführung als XML-Datei speichern.

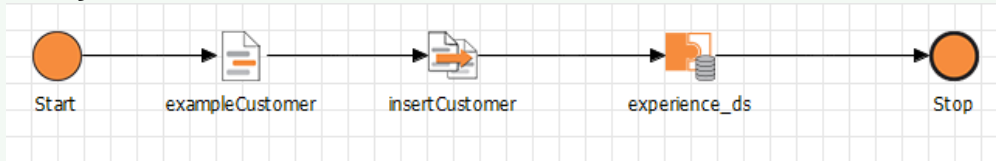
SQL-Statement im Prozess verwenden

1. Die SQL-Statement-Datei aus dem Repository per Drag&Drop in den Prozess ziehen.
2. In der **Properties**-Sicht für die Eigenschaft Operation den Wert *Read* wählen.
3. Den Datenbankverbindungs-Baustein hinter den SQL-Statement-Baustein in den technischen Prozess einfügen.

Die Datenbank-Verbindung verwendet nun die SQL-Abfrage als Input und führt diese aus.



- ✓ Alternativ können Sie auch das Input-XML-Dokument dynamisch über ein XSL-Mapping erzeugen:



1.4.2.4.1.1 Statement für eine verbundene Datenbank erstellen

- Statement für eine verbundene Datenbank erstellen
- Named Prepared Statement anlegen und bearbeiten
- SQL-Statement: INSERT
- SQL-Statement: SELECT
- SQL-Statement: UPDATE
- SQL-Statement: DELETE

Named Prepared Statement anlegen und bearbeiten

Beim Named Prepared Statement handelt es sich um ein vom Datenbank-Editor unterstütztes XML-Format für den *JDBC Connector*, welches die Nutzung von benannten Platzhaltern (Named Parameters) innerhalb des SQL-Blocks unterstützt.

Neues Named Prepared Statement anlegen

1. Auf Statements Root rechtsklicken, um das Kontextmenü aufzurufen.
2. **New Named Prepared Statement...** wählen.

Das Fenster *Add Named Prepared Statement* wird geöffnet.

Das Dialogfeld 'Add Named Prepared State...' enthält folgende Felder:

- Resulttag:** Ein Textfeld mit einem roten 'x' Symbol rechts daneben.
- Time Zone:** Ein Textfeld.
- Date Format:** Ein Textfeld.
- Unter dem Feldern befinden sich zwei Buttons: **OK** und **Cancel**.

3. Informationen zum *Named Prepared Statement* angeben:
 - **Resulttag** (Pflichtfeld): Name des XML-Elements, welches im Output-XML-Dokument die Ergebnisse des Statements enthält
 - **Time Zone:** Zeitzone eingeben
 - **Date Format:** gewünschtes Datenformat eintragen

4. **OK** klicken.

Das Named Prepared Statement wurde nun als Kindelement von Statements Root angelegt.

Statement	
▼ Statements Root	
▼ Named Prepared Statement	
Resulttag	result
Time Zone	GMT
Date Format	dd.MM
▼ RowSet	
Row	

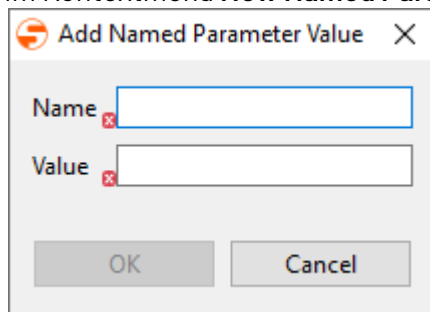
Named Prepared Statement bearbeiten

1. Auf ein vorhandenes Named Prepared Statement rechtsklicken.
2. Im Kontextmenü **Edit Named Prepared Statement..** wählen.

3. Gewünschte Informationen bearbeiten:
 - Resulttag (Pflichtfeld): Name des XML-Elements, welches im Output-XML-Dokument die Ergebnisse des Statements enthält
 - Time Zone: Zeitzone
 - Date Format: gewünschtes Datenformat
4. **OK** klicken.
Die Änderungen wurden nun übernommen.

Neuen Named Parameter Wert hinzufügen

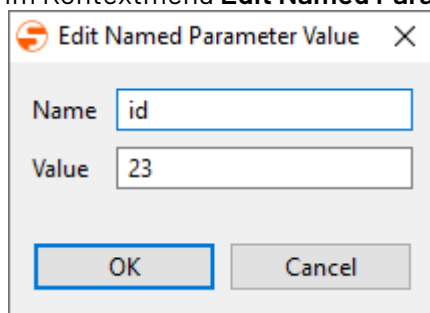
1. Innerhalb eines RowSets auf eine Row rechtsklicken.
2. Im Kontextmenü **New Named Parameter Value...** wählen.



3. Gewünschte Informationen eingeben
 - Unter **Name** den Namen des Named Parameters eingeben
 - Unter **Value** den entsprechenden Parameter-Wert
4. **OK** klicken.
Der Named Parameter Wert wurden nun angelegt.

Named Parameter bearbeiten

1. Innerhalb eines Rows auf einen Parameter rechtsklicken.
2. Im Kontextmenü **Edit Named Parameter...** wählen.



3. Gewünschte Informationen ändern
 - Unter **Name** den Namen des Named Parameters eingeben
 - Unter **Value** den entsprechenden Parameter-Wert
4. **OK** klicken.
Die Änderungen wurden nun gespeichert.

SQL-Statement: INSERT

Neues INSERT-Statement einfügen

1. Auf ein Named Prepared Statements rechtsklicken, um das Kontextmenü aufzurufen.
2. **New INSERT Statement** wählen.

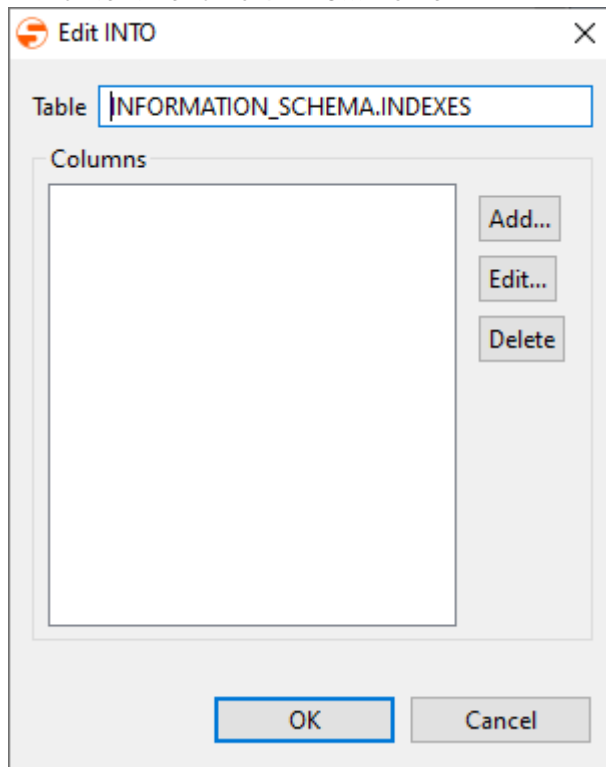
Ein neues INSERT-Statement mit folgenden Elementen wird angelegt:

- INTO
- VALUES.

INSERT INTO bearbeiten

1. Innerhalb eines INSERT-Statements auf INTO rechtsklicken.

2. Im Kontextmenü **Edit INTO...** wählen.



3. Gewünschte Änderungen vornehmen:
- Unter **Table** ggf. einen neuen Tabellennamen eingeben
 - Über **Add** eine neue Spalte anlegen
 - Mit **Edit** einen zuvor ausgewählten Wert bearbeiten
 - Mit **Delete** einen zuvor ausgewählten Wert löschen.
4. **OK** klicken.
Die Änderungen wurden nun übernommen.

- ✓ Sie können die Tabelle und die entsprechenden Spalten, in die geschrieben werden soll, auch per Drag & Drop direkt im *Statement*-Bereich setzen.

Neue Spalte einfügen

ℹ Siehe [Neue Spalte einfügen](#).

Neuen Eintrag in VALUES anlegen

1. Innerhalb eines INSERT-Statements auf VALUES rechtsklicken.

- Im Kontextmenü **New VALUE entry...** wählen.

The screenshot shows the 'Add Value' dialog box. It has a title bar with a red 'X' icon and a close button. The dialog contains three radio buttons: 'Text' (selected), 'Number', and 'Named Parameter'. Each radio button is followed by a text input field. Below these is a 'Type' dropdown menu showing 'VARCHAR' and a 'Mode' dropdown menu. At the bottom are 'OK' and 'Cancel' buttons.

- Gewünschte Informationen eingeben:

- Um einen Text-Wert einzugeben, **Text** wählen und den entsprechenden Wert eintragen
- Um eine Zahl einzugeben, **Number** wählen und den entsprechenden Wert eintragen
- Um einen Named Parameter einzugeben, **Named Parameter** wählen und den entsprechenden Wert eintragen.
 - Unter **Type** ggf. einen Feldtypen aus der Drop-down-Liste wählen; Standardmäßig ist VARCHAR hinterlegt
 - Unter **Mode** ggf. mit leer, in, out oder inout ein Modus wählen

- OK** klicken.

Der Eintrag wurde nun als Kindelement von VALUES angelegt.

VALUE bearbeiten

- Innerhalb eines INSERT-Statements auf einen VALUES-Wert rechtsklicken.

- Im Kontextmenü **Edit VALUE...** wählen.

- Gewünschte Änderungen vornehmen:

- Um einen Text-Wert zu ändern, in **Text** den gewünschten Wert eintragen
- Um einen Zahlen-Wert zu ändern, in **Number** den gewünschten Wert eintragen
- Um einen Named Parameter zu ändern, in **Named Parameter** den gewünschten Wert eintragen.
 - Unter **Type** ggf. einen neuen Feldtypen aus der Drop-down-Liste wählen
 - Unter **Mode** ggf. mit leer, in, out oder inout den Modus ändern

- OK** klicken.

Der Eintrag wurde nun geändert.

SQL-Statement: SELECT

Neues SELECT-Statement einfügen

- Auf ein Named Prepared Statements rechtsklicken, um das Kontextmenü aufzurufen.
- New SELECT Statement** wählen.

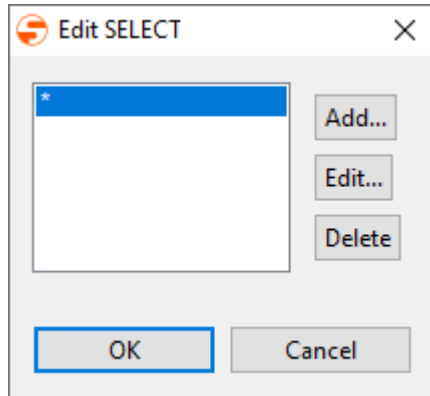
Ein neues SELECT-Statement mit folgenden Elementen wird angelegt:

- SELECT
- FROM
- WHERE .

SELECT bearbeiten

- Innerhalb eines SELECT-Statements auf den SELECT-Befehl rechtsklicken.

2. Im Kontextmenü **Edit SELECT...** wählen.

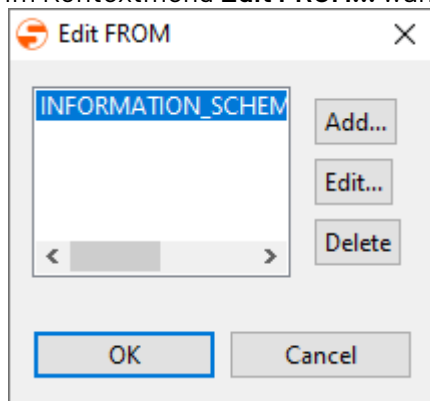


3. Gewünschte Änderungen vornehmen:
 - Über **Add** eine neue Spalte anlegen
 - Mit **Edit** einen zuvor ausgewählten Wert bearbeiten
 - Mit **Delete** einen zuvor ausgewählten Wert löschen.
4. **OK** klicken.
Die Änderungen wurden nun übernommen.

✓ Sie können die zu selektierende Spalte auch per Drag & Drop direkt im *Statement*-Bereich setzen.

FROM bearbeiten

1. Innerhalb eines SELECT-Statements auf FROM rechtsklicken.
2. Im Kontextmenü **Edit FROM...** wählen.



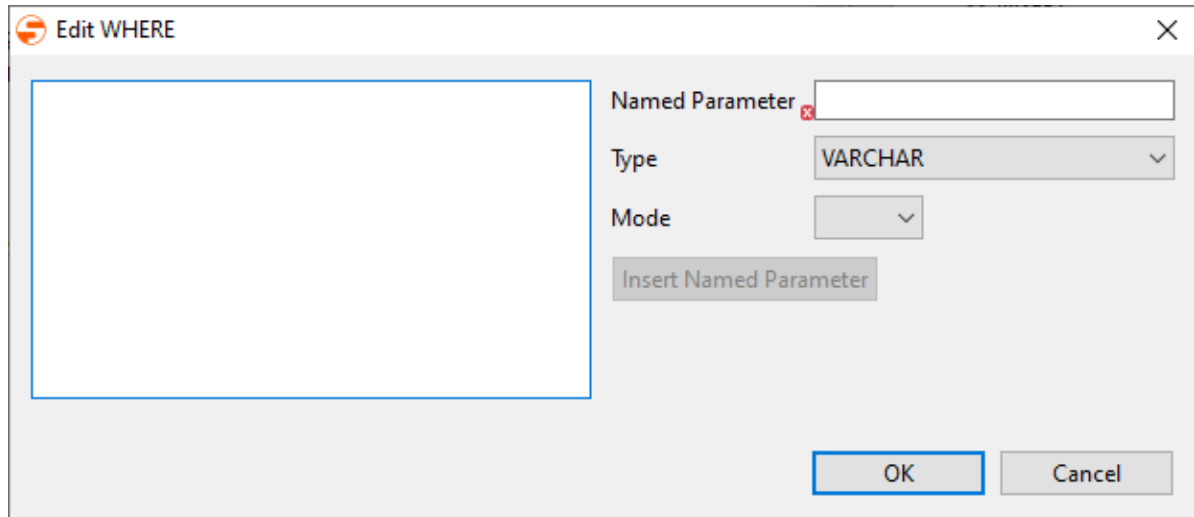
3. Gewünschte Änderungen vornehmen:
 - Über **Add** eine neue Tabelle anlegen
 - Mit **Edit** einen zuvor ausgewählten Wert bearbeiten
 - Mit **Delete** einen zuvor ausgewählten Wert löschen.
4. **OK** klicken.
Die Änderungen wurden nun übernommen.

✓ Sie können die Tabelle aus der selektiert werden soll auch per Drag & Drop direkt im *Statement*-Bereich setzen.

WHERE bearbeiten

WHERE wird verwendet, um Auswahlbedingungen zu formulieren. Je sorgfältiger die Auswahlbedingungen formuliert werden, desto genauer ist das Ergebnis der Abfrage. Über Named Parameter besteht dabei zusätzlich die Möglichkeit, mit Hilfe von Platzhaltern und Variablen die Auswahlbedingungen zu parametrieren.

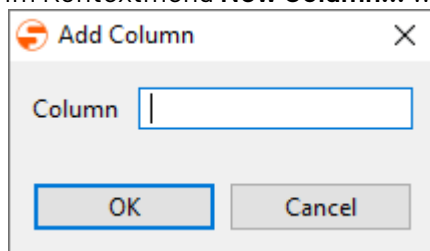
1. Innerhalb eines DELETE-, SELECT- oder UPDATE-Statements auf einen WHERE-Eintrag rechtsklicken.
2. Im Kontextmenü **Edit WHERE...** wählen.



3. Gewünschte Named Parameter hinterlegen:
 - In **Named Parameter** einen Parameter eingeben.
 - Unter **Type** einen neuen Feldtypen aus der Drop-down-Liste wählen
 - Unter **Mode** mit leer, in, out oder inout den Modus angeben
4. **Insert NamedParameter** klicken, um den Named Parameter hinzuzufügen.
Der Parameter wird nun übernommen und im Textfeld angezeigt.
5. Wenn noch weitere Named Parameter hinterlegt werden sollen, die Schritte 3 und 4 entsprechend oft wiederholen.
6. **OK** klicken.
Die Eingabe wurden nun gespeichert.

Neue Spalte einfügen

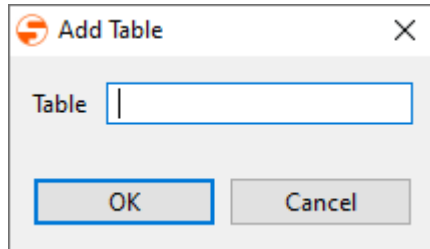
1. Innerhalb eines INSERT-oderSELECT-Statements auf einen INTO- bzw. auf einen SELECT-Eintrag rechtsklicken.
2. Im Kontextmenü **New Column...** wählen.



3. Unter **Column** einen Spalten-Namen eingeben.
4. **OK** klicken.
Die Spalte wurden nun angelegt.

Neue Tabelle einfügen

1. Innerhalb eines UPDATE-oder SELECT-Statements auf einen UPDATE- bzw. auf einen FROM-Befehl rechtsklicken.
2. Im Kontextmenü **New Table...** wählen.



3. Unter **Table** einen Tabellen-Namen eingeben.
4. **OK** klicken.
Die Tabelle wurden nun angelegt.

SQL-Statement: UPDATE

Neues UPDATE-Statement einfügen

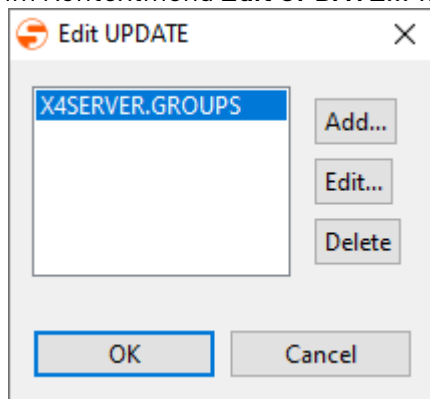
1. Auf ein Named Prepared Statements rechtsklicken, um das Kontextmenü aufzurufen.
2. **New UPDATE Statement** wählen.

Ein neues UPDATE-Statement mit folgenden Elementen wird angelegt:

- UPDATE
- SET
- WHERE .

UPDATE bearbeiten

1. Innerhalb eines UPDATE-Statements auf UPDATE rechtsklicken.
2. Im Kontextmenü **Edit UPDATE...** wählen.



3. Gewünschte Änderungen vornehmen:
 - Über **Add** eine neue Tabelle anlegen
 - Mit **Edit** einen zuvor ausgewählten Wert bearbeiten
 - Mit **Delete** einen zuvor ausgewählten Wert löschen.
4. **OK** klicken.
Die Änderungen wurden nun übernommen.

- ✓ Sie können die zu aktualisierende Tabelle auch per Drag & Drop direkt im *Statement*-Bereich setzen.

Neuen Eintrag in SET anlegen

1. Innerhalb eines UPDATE-Statements auf SET rechtsklicken.

- Im Kontextmenü **New SET entry...** wählen.

- Gewünschte Informationen eingeben:

- Unter **Column** den Spaltennamen eingeben
- Um einen Text-Wert einzugeben, **Text** wählen und den entsprechenden Wert eintragen
- Um eine Zahl einzugeben, **Number** wählen und den entsprechenden Wert eintragen
- Um einen Named Parameter einzugeben, **Named Parameter** wählen und den entsprechenden Wert eintragen.
 - Unter **Type** ggf. einen Feldtypen aus der Drop-down-Liste wählen; Standardmäßig ist VARCHAR hinterlegt
 - Unter **Mode** ggf. mit leer, in, out oder inout ein Modus wählen

- OK** klicken.

Der Eintrag in der SET-Klausel wurde nun angelegt

SET bearbeiten

- Innerhalb eines UPDATE-Statements auf einen SET-Eintrag rechtsklicken.
- Im Kontextmenü **Edit SET entry...** wählen.
- Gewünschte Informationen bearbeiten:
 - Unter **Column** den neuen Spaltennamen eingeben
 - Um einen Text-Wert zu ändern, in **Text** den gewünschten Wert eintragen
 - Um einen Zahlen-Wert zu ändern, in **Number** den gewünschten Wert eintragen
 - Um einen Named Parameter zu ändern, in **Named Parameter** den gewünschten Wert eintragen.
 - Unter **Type** ggf. einen neuen Feldtypen aus der Drop-down-Liste wählen
 - Unter **Mode** ggf. mit leer, in, out oder inout den Modus ändern
- OK** klicken.
Die Änderungen wurden nun gespeichert.



Sie können die zu aktualisierende Spalte auch per Drag & Drop direkt im *Statement*-Bereich setzen.

WHERE bearbeiten



Siehe [WHERE bearbeiten](#).

SQL-Statement: DELETE

Neues DELETE-Statement einfügen

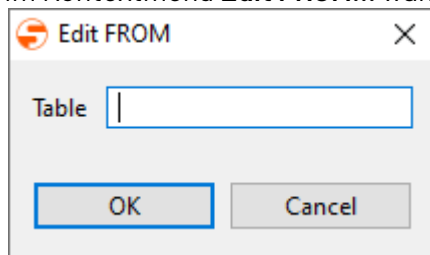
1. Auf ein Named Prepared Statements rechtsklicken, um das Kontextmenü aufzurufen.
2. **New DELETE Statement** wählen.

Ein neues DELETE-Statement mit folgenden Elementen wird angelegt:

- FROM
- WHERE .

DELETE FROM bearbeiten

1. Innerhalb eines DELETE-Statements auf einen FROM-Eintrag rechtsklicken.
2. Im Kontextmenü **Edit FROM...** wählen.



3. In **Table** den gewünschten Tabellen-Namen eingeben
4. **OK** klicken.

Die Eingaben wurden nun gespeichert.

- ✓ Sie können die zu löschende Tabelle auch per Drag & Drop direkt im *Statement*-Bereich setzen.

WHERE bearbeiten

- ❗ Siehe [WHERE bearbeiten](#).

Neue Tabelle einfügen

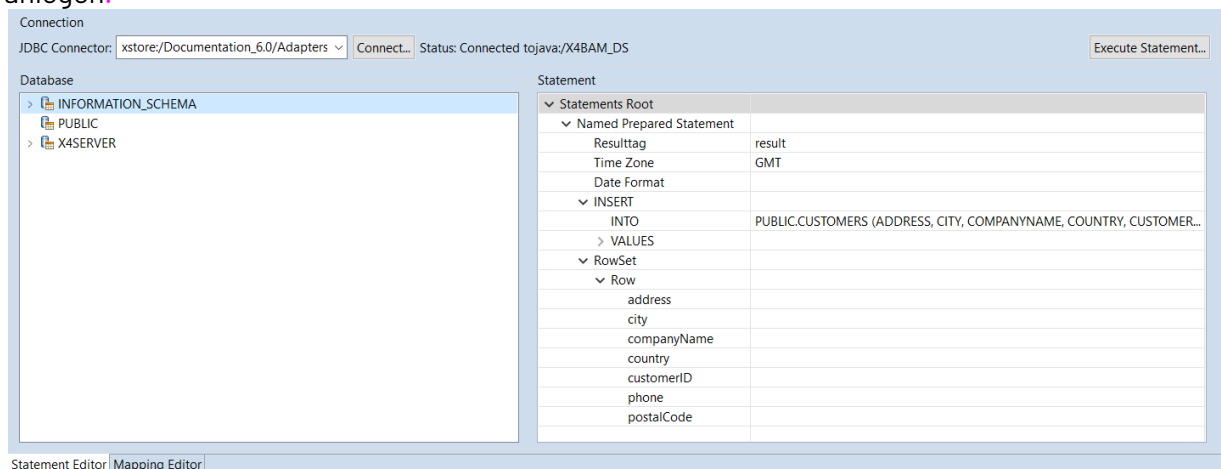
- ❗ Siehe [Neue Tabelle einfügen](#).

1.4.2.4.1.2 Daten-Mapping erstellen und XSL-Datei erzeugen

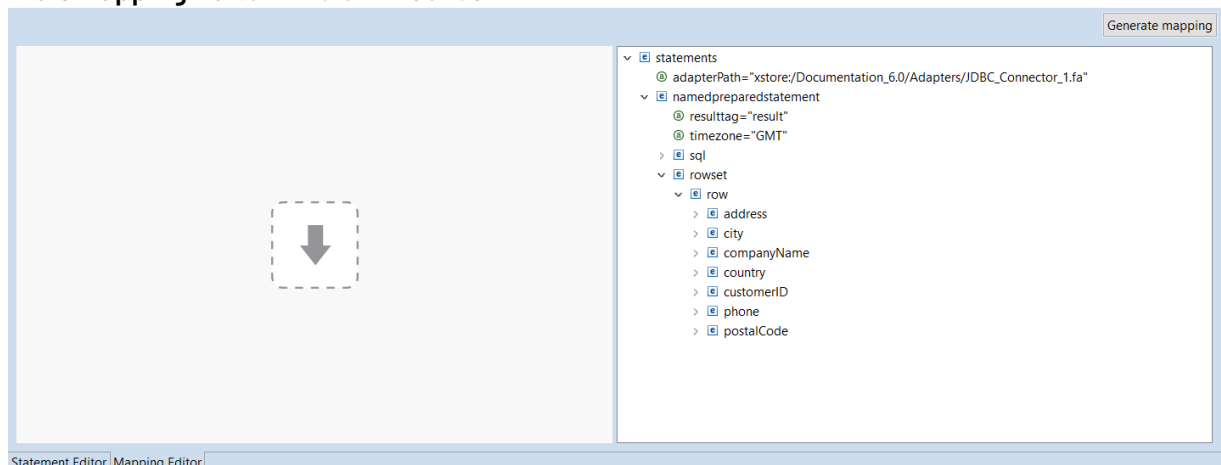
Für SQL-Statements, die zuvor im Datenbank-Editor angelegt wurden, lassen sich XSL-Mappings von einem Input-XML-Dokument auf die Named Parameters der Statements per Drag & Drop konfigurieren und anschließend erstellen. Diese können mit einem passenden Input-XML-Dokument eine dynamisch erzeugte Eingabe für den [JDBC Connector](#) erzeugen.

1. SQL-Statements erstellen, siehe auch [Statement für eine verbundene Datenbank erstellen](#).

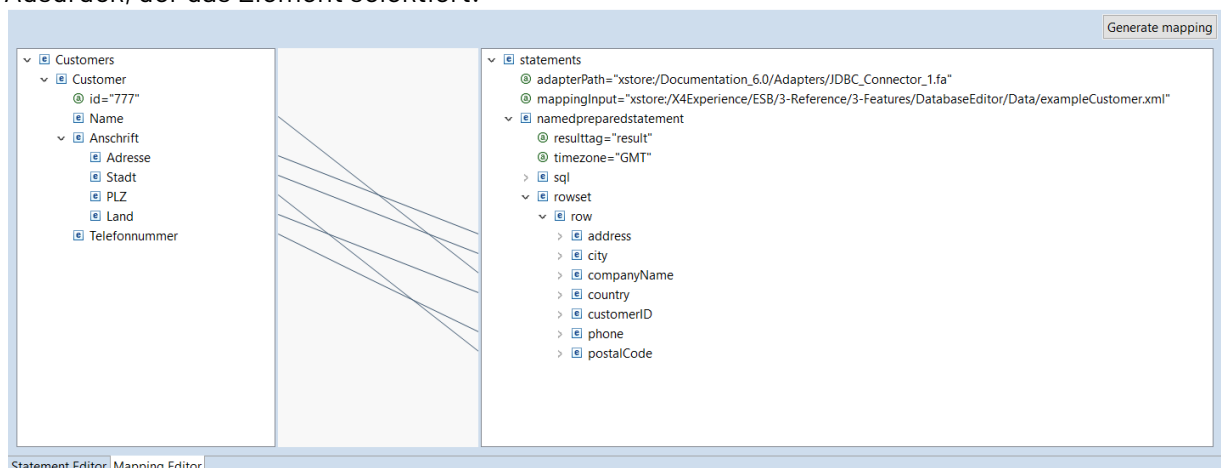
2. Unter **RowSet > Row** die **Named Parameter Values** für die entsprechenden *Named Parameters* anlegen.



3. In die **Mapping Editor**-Ansicht wechseln.




4. Das gewünschte Input-XML-Dokument per Drag & Drop aus dem Repository in den Quelldokument-Bereich ziehen.
Die Struktur der XML-Datei wird nun angezeigt.
5. Die gewünschten Elemente aus dem Quelldokument-Bereich per Drag & Drop auf die entsprechenden Named Parameter ziehen.
Auf den *Named Parameters* wird jeweils ein `value-of` Unterknoten erstellt mit einem XPath-Ausdruck, der das Element selektiert.



6. Speichern und mit **Generate Mapping** das XSL-Mapping erzeugen.

1.4.2.4.2 Datenbank-Editor

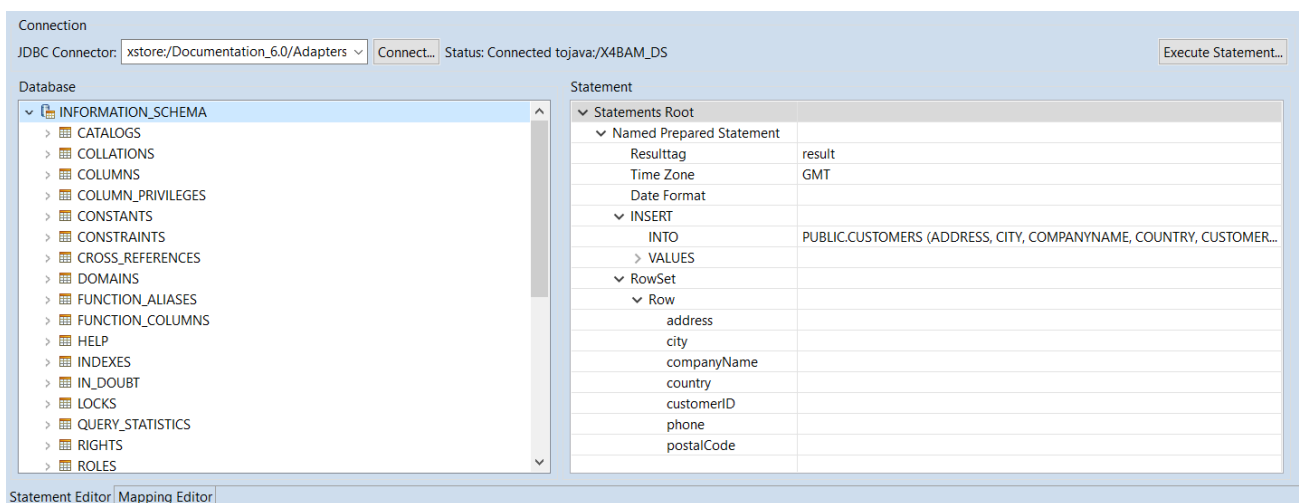
Mit dem Datenbank-Editor lassen sich SQL-Abfragen für Datenbankverbindungs-Bausteine definieren, um Daten aus einer relationalen Datenbank abzufragen. Durch die logische Trennung der Prozessbausteine SQL-Anweisung und Datenbank-Verbindung lassen sich im X4 Designer diese Prozessbausteine flexibel einsetzen und mehrfach verwenden.

 Um den Datenbank-Editor aufzurufen, legen Sie eine neue SQL-Statement-Datei (.stm) an oder öffnen Sie eine bereits vorhandene SQL-Statement-Datei.

Aufbau des Datenbank-Editors








Der Datenbank-Editor besteht aus folgenden Bereichen:

- **Statement Editor:** Hier kann eine Verbindung zu einer Datenbank aufgebaut und die Struktur der verbundenen Datenbank angezeigt werden. Mit dieser Struktur können anschließend SQL-Statements für die Datenbank erstellt werden.
- **Mapping Editor:** Hier lässt sich für das im Statement Editor zuvor erstellte Statement ein Mapping von Elementen eines Input-XML-Dokuments auf die Named Parameter des Statements definieren. Aus diesem Mapping lässt sich anschließend ein XSL-Mapping erstellen, welches mit einem passenden Input-XML-Dokument eine dynamisch erzeugte Eingabe für den [JDBC Connector](#) erzeugt.



1.4.2.4.2.1 Statement Editor

Der Statement Editor ist in folgende Bereiche unterteilt:

Connec- tion- Bereich	<p>Hier können Sie über einen <i>JDBC Connector</i> eine Verbindung zu einer Datenbank aufbauen</p> <ul style="list-style-type: none"> • JDBC Connector: xstore-URL zu einem <i>JDBC Connector</i> <div data-bbox="319 398 1433 573" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> • Um eine Verbindung aufbauen zu können, muss der Adapter innerhalb des Parameters <code>JndiName</code> einen gültigen JNDI-Namen enthalten.</p> <p>• In der Drop-Down-Liste werden die letzten 8 verbundenen URLs zur Auswahl angeboten.</p> </div> <ul style="list-style-type: none"> • Connect: Mit der Datenbank verbinden  Unter Status wird der Verbindungsstatus angezeigt. • Execute Statement: Statement ausführen
Databa- se- Bereich	<p>Hier wird nach dem Verbindungsaufbau die Struktur der Datenbank angezeigt. Folgende Icons werden für die Darstellung der Struktur verwendet:</p> <ul style="list-style-type: none"> •  Datenbankschema •  Datenbanktabelle •  Primärschlüssel •  Primärschlüssel / Fremdschlüssel •  Fremdschlüssel

**Statement-
ent-
Bereich**






Hier lassen sich über das Kontextmenü verschiedenen Statements für die verbundene Datenbank erstellen:

- **New Named Prepared Statement:** Im Statements Root ein neues Named Prepared Statement anlegen
- **Edit Named Prepared Statement:** Named Prepared Statement bearbeiten
- **New RowSet:** Neues RowSet anlegen.
 ⓘ Innerhalb eines Named Prepared Statements kann lediglich ein RowSet existieren. Dieses wird automatisch beim Anlegen des Named Prepared Statements mit angelegt. Ein neues RowSet kann nur angelegt werden, wenn das bestehende RowSet gelöscht wurde.
- **New Row:** Neue Zeile innerhalb eines RowSets hinzufügen
- **New Named Parameter Value:** Einen neuen Named Parameter-Wert innerhalb einer Zeile hinzufügen
- **Edit Named Parameter:** Named Parameter bearbeiten
- **New SELECT Statement:** Ein neues SELECT-Statement anlegen
- **Edit SELECT:** SELECT-Befehl bearbeiten
- **Edit FROM:** FROM innerhalb eines SELECT-Statements bearbeiten
- **New INSERT Statement:** Ein neues INSERT-Statement anlegen
- **Edit INTO:** INTO-Befehl bearbeiten
- **New VALUE Entry:** Neuen Eintrag unter VALUE anlegen
- **Edit VALUE:** Wert eines VALUES bearbeiten
- **New UPDATE Statement:** Ein neues UPDATE-Statement anlegen
- **Edit UPDATE:** UPDATE-Befehl bearbeiten
- **New SET Entry:** Neuen Eintrag innerhalb von SET anlegen
- **Edit SET Entry:** Eintrag innerhalb von SET bearbeiten
- **New DELETE Statement:** Ein neues DELETE-Statement anlegen
- **Edit FROM:** FROM-Befehl bearbeiten
- **Edit WHERE:** WHERE-Bedingung bearbeiten
- **New Column:** Neue Spalte innerhalb eines INSERT- oder SELECT-Befehls einfügen
- **New Table:** Neue Tabelle innerhalb eines UPDATE- oder SELECT-Befehls einfügen
- **Delete:** Element löschen

ⓘ Mehr Informationen zum Anlegen und Bearbeiten von Statements finden Sie im Abschnitt [Statement für eine verbundene Datenbank erstellen](#).

Drag&Drop-Aktionen

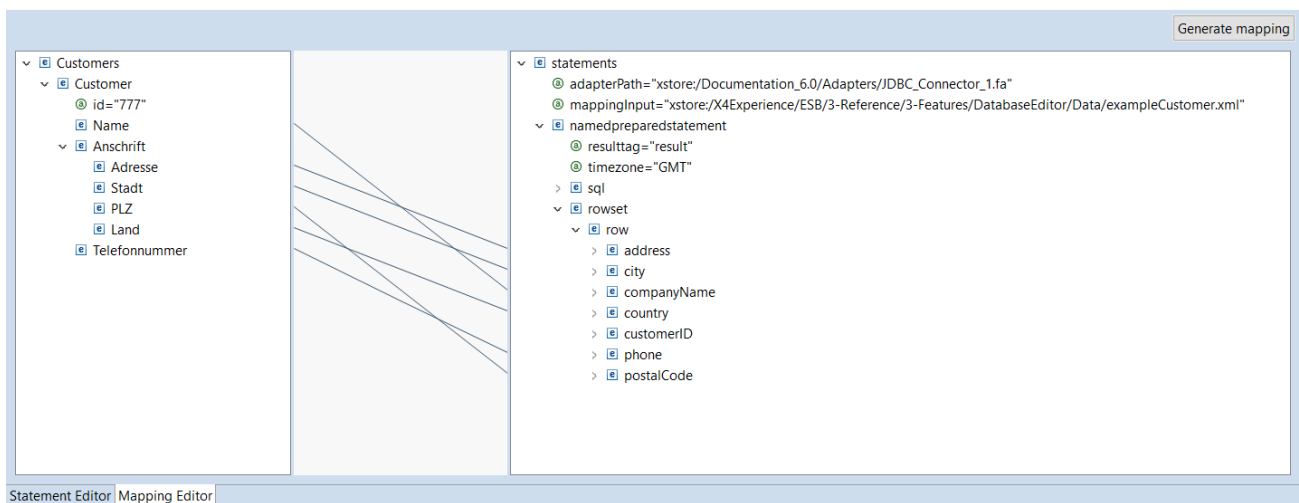
Folgende Drag&Drop-Aktionen sind im Statement-Bereich möglich:

SQL-Statement	Datenbank-Element	Statement-Element	Aktion
SELECT	Column	SELECT	Die zu selektierende Spalte hinzufügen.  Steht die Tabelle der Spalte noch nicht im FROM, wird diese hinzugefügt.
	Table	FROM	Tabelle aus der selektiert werden soll hinzufügen.
UPDATE	Column	SET Entry	Spalte als Eintrag setzen.  Steht die Tabelle der entsprechenden Spalte noch nicht im UPDATE, wird diese hinzugefügt.
	Table	UPDATE	Die zu aktualisierende Tabelle setzen
INSERT	Table	INTO	Tabelle setzen in die geschrieben werden soll.  Ist bereits eine Tabelle vorhanden, wird diese ersetzt. Dabei werden auch bereits gesetzte Spalten (Columns) gelöscht.
	Column	INTO	Spalte setzen in die geschrieben werden soll.  Dies ist nur möglich, wenn die Spalte aus der gesetzten Tabelle stammt.
DELETE	Table	FROM	Die zu löschende Tabelle setzen.  Ist bereits eine Tabelle vorhanden, wird diese ersetzt.

1.4.2.4.2.2 Mapping Editor

Innerhalb des Mapping Editors lassen sich für die im Statement Editor zuvor erstellten Statements, Mappings von einem Input-XML-Dokument auf die Named Parameters der Statements per Drag & Drop konfigurieren und anschließend erstellen. Diese erzeugen mit einem passenden Input-XML-Dokument eine dynamisch erzeugte Eingabe für den [JDBC Connector](#).

Der Mapping Editor ist in folgende Bereiche unterteilt:



Quelldokument-Bereich	<p>Auf der linken Seite befindet sich in der Quelldokument-Bereich, in dem Sie die Input-XML-Datei per Drag & Drop ablegen und anzeigen können.</p> <p>Über das Kontextmenü können Sie den gewählten Knoten folgendermaßen bearbeiten:</p> <ul style="list-style-type: none"> • Collapse: Knoten mit aufgeklappten Unterelementen zuklappen • Expand: Unterelemente des Knotens aufklappen • Remove Input: Input-XML-Dokument entfernen
Verknüpfungsbereich	In der mittleren Spalte befindet sich der Verknüpfungsbereich. Hier werden die <code>xs1:value-of</code> Verknüpfungen zwischen Element im Quelldokument und Named Parameter im Statement als Linien visualisiert.
Statements-Bereich	Auf der rechten Seite befindet sich der Statements-Bereich. Hier wird die gesamte Struktur der Statement-Datei angezeigt. Dabei können Mappings vom Input-XML auf die Kind-Elemente eines Row-Elements erstellt werden.

1.4.2.5 Merge



Merge: Führt XML-Dokumente aus unterschiedlichen Quellen zu einem einzigen XML-Dokument oder führt Text- oder Binärdatenströme zusammen.

Eigenschaften

Label	Beschriftung des Symbols im Prozess-Diagramm
Root Element	<p>Name des Wurzelements, das den gesamten Inhalt der zusammengeführten XML-Dokumente enthält.</p> <p>Mögliche Werte: Zeichenkette, die für XML-Elementnamen gültig ist. Standard: <i>Merge</i></p>

Daten mit Merge zusammenführen

Der Merge-Baustein wird meist hinter der Zusammenführung einer Gabelung (Join) verwendet. Wenn XML-Dokumente zusammengeführt werden, dann enthält das durch den Merge-Baustein erzeugte XML-Dokument ein Wurzelement, das wiederum die Root-Elemente und sämtliche Knoten der

zusammengeführten XML-Dokumente enthält. Diese XML-Struktur kann vom nächsten Prozess-Schritt verarbeitet werden. Der Name des Wurzelements kann über die Eigenschaft `Root Element` des Merge-Bausteins festgelegt werden.

Wenn zwei XML-Dokumente zusammengeführt werden, die bereits in einem vorherigen Schritt durch den Merge-Baustein zusammengeführt wurden, enthält das Ergebnis-XML-Dokument lediglich ein Wurzelement und sämtliche Knoten der zusammengeführten XML-Dokumente.

Zusammenführung von zwei Dokumenten

```
<?xml version="1.0" encoding="UTF-8" ?>
<Merge>
  <!-- Daten 1 -->
  <!-- Daten 2 -->
</Merge>
```

Wenn die Input-Dokumente Textdaten (MIME-Typ `text ...` oder `application/xml`) mit derselben Zeichenkodierung sind, dann wird deren Zeichenkodierung ausgegeben; bei unterschiedlicher Zeichenkodierung wird eine UTF-8-Sequenz ausgegeben. Wenn der MIME-Typ bei allen Input-Dokumenten gleich ist, dann hat das Ergebnis auch diesen MIME-Typ, ansonsten `text/plain`.

1.4.2.6 Variable



Variable: Speichert temporär Daten, während ein Prozess ausgeführt wird. Innerhalb desselben Prozesses oder eines Subprozesses können Sie die Variable auslesen. Mit Hilfe des `context:-`Protokolls können Sie auch in XSL-Mappings auf solche Variablen zugreifen.

Eigenschaften

Operation	Bestimmt, ob die Variable initialisiert, verwendet oder verworfen werden soll Mögliche Werte: <ul style="list-style-type: none"> • <i>Put</i>: Daten aus dem vorangegangenen Prozess-Schritt in der Variable speichern (Standard) • <i>Get</i>: Variablen-Inhalt lesen und an den nächsten Prozess-Schritt weitergeben • <i>Remove</i>: Variable verwerfen (aus dem Arbeitsspeicher löschen) • <i>Test</i>: Prüfen, ob eine Variable vorhanden ist; Ist keine Variable gesetzt, wird der Status <code>0</code> ausgegeben
-----------	---

Name	<p>Eindeutige Bezeichnung der Variable</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Zeichenkette, die aus alphanumerischen <i>US-ASCII</i>-Zeichen (<i>Aa-Zz</i> und <i>0-9</i>) sowie aus Unterstrichen (<i>_</i>) bestehen darf und nicht mit einer Ziffer beginnt (z. B. <i>Report_Data01</i>) • <i>X4STATE</i>: Enthält den Status des vorherigen Prozess-Schritts als XML-Dokument; siehe <i>X4 Entwickler-Referenz - Kontextvariablen</i> • <i>X4PREVACTION</i>: Enthält Action ID, Beschriftungstext und Status des vorherigen Prozess-Schritts als XML-Dokument
------	--

1.4.2.6.1 Variablen verwenden

In Kontextvariablen lassen sich Daten im Kontext eines aktuell ausgeführten technischen Prozesses zwischenspeichern. Jede Kontextvariable muss zunächst initialisiert werden, also einen eindeutigen Namen erhalten und mit einem Wert belegt werden (*Put*-Operation). In späteren Prozess-Schritten kann dann per *Get*-Operation auf die Kontext-Variable zugegriffen werden. Schließlich können Variableninhalte über die *Remove*-Operation aus dem Speicher gelöscht werden.

<i>Variablen-Baustein erstellen</i>	Per Drag&Drop aus der Palette in den technischen Prozess ziehen und in der Properties -Sicht in der Eigenschaft <i>Name</i> einen eindeutigen Variablennamen eingeben (z. B. <i>Var1</i>).
<i>Variable initialisieren</i>	in der Properties -Sicht in der Eigenschaft <i>Operation</i> den Wert <i>Put</i> wählen, um die Daten aus dem vorangegangenen Prozess-Schritt in die Variable zu laden.
<i>Variable auslesen</i>	Den Variablen-Baustein mit demselben Namen erneut einfügen und die Eigenschaft <i>Operation</i> auf <i>Get</i> setzen.
<i>Variable testen</i>	Den Variablen-Baustein mit demselben Namen erneut einfügen und die Eigenschaft <i>Operation</i> auf <i>Test</i> setzen.
<i>Variable aus dem Speicher löschen</i>	Den Variablen-Baustein mit demselben Namen erneut einfügen und die Eigenschaft <i>Operation</i> auf <i>Remove</i> setzen.
<i>Innerhalb von XSL-Mappings auf Prozessvariablen zugreifen</i>	Das <i>context://</i> -URL-Protokoll verwenden, z. B. über den XPath-Ausdruck <code>document(' context:// Var1 ')</code>

1.4.2.6.2 Kontextvariablen

In Prozessen stehen standardmäßig eine Reihe von Kontextvariablen zur Verfügung, die Informationen zur aktuell ausgeführten Prozessinstanz liefern. Diese Variablen können Sie sowohl im

Prozess über einen `Variable`-Baustein über die Operation `Get` als auch in XSL-Mappings über das `context:-URL`-Protokoll abrufen.

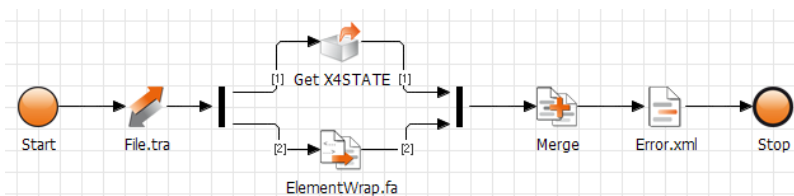
X4STATE	<p>Gibt den Status des vorherigen Prozess-Schritts (unmittelbar vor dem aktuellen Prozess-Schritt) in einem XML-Dokument aus</p> <div> <p>Beispiel</p> <pre><state>-1</state></pre> </div>
X4PREV ACTION	<p>Gibt Informationen (Action ID, Beschriftungstext und Status) zum vorherigen Prozess-Schritt als XML-Dokument aus</p> <div> <p>Beispiel</p> <pre><action> <id>4</id> <label>File.tra</label> <state>-1</state> </action></pre> </div>

Beispiel

Mit diesen Kontextvariablen wird eine erweiterte Fehlerbehandlung ermöglicht. So können Sie z. B. den Fehlermeldungstext eines Adapters gemeinsam mit dem Adapterstatus ausgeben.

```
<Merge>
  <state>-1</state>
  <error>General File Error (see log for details).</error>
</Merge>
```

Adapter-Fehlermeldung und dessen Statuscode zusammengeführt in ein XML-Dokument:



1.4.3 Modul-Bausteine

Die folgenden Modul-Bausteine verbinden IT-Prozesse mit der Außenwelt und anderen Prozessen, beispielsweise in Form von Datenbank-Verbindungen, Funktions-Adaptoren und Subprozessen. Modul-Bausteine liegen als wiederverwendbare Dateien im Repository vor. Deren Konfiguration erfolgt entweder in der jeweiligen Adapter-Konfiguration oder direkt in einem XSL-Stylesheet (bei Mappings und Reports).

1.4.3.1 Database



Database: Baut eine Verbindung zu einer Datenbank auf und kommuniziert mit dieser. Im vorangehenden Prozess-Schritt werden in einer SQL-Anweisung die Datenbank-Operation übermittelt, siehe auch [SQL Statement](#). Der Datenbankverbindungs-Baustein führt diese Anweisungen aus.

- i** Um den Datenbankverbindungs-Typ (JDBC-Verbindung) zu wählen und Parameter festzulegen, doppelklicken Sie auf der Zeichnungsfläche auf das entsprechende Datenbankverbindungs-Symbol.

Eigenschaften

Version	Adapter-Version
Document	Pfad zum verknüpften Datenbank-Adapter im Repository
Label	Beschriftung des Symbols im Prozess-Diagramm
Custom Parameters	Benutzerdefinierte Parameter setzen, siehe Benutzerdefinierte Parameter verwenden
Operation	Operationen des jeweiligen Datenbank-Adapters

1.4.3.2 Transfer





Transfer: Greift über verschiedene Protokolle auf Daten zu. Transfer-Adapter ermöglichen den Datenaustausch von und zum Dateisystem, via FTP und SFTP, E-Mail, SMS und HTTP ebenso wie die Kommunikation mit Nachrichten-Warteschlangen wie JMS, MQ Series und RequestReply.

- i** Um den Transfer-Adaptertyp zu wählen und Parameter festzulegen, doppelklicken Sie auf der Zeichnungsfläche auf das entsprechende Transfer-Adapter-Symbol. Die spezifische Bedeutung der im folgenden genannten Eigenschaften der Transfer-Adapter, ist in der jeweiligen Adapter-Referenz beschrieben.

Eigenschaften


Version	Adapter-Version
Document	Pfad zum verknüpften Transfer-Adapter im Repository Alternativ dazu können Sie eine URL angeben, die auf die Datei in einem Web-Archiv statisch oder dynamisch verweist. Damit muss nur noch das Web-Archiv umgebungsabhängig ausgetauscht werden. Das Web-Archiv wird direkt im Ordner deploy des Applikations-Servers abgelegt.

File name	<p>Name der zu verarbeitenden Datei</p> <p> Wenn Sie diese Eigenschaft dynamisch parametrieren möchten, müssen Sie im zugehörigen XSL-Mapping deren technischen Namensdocumentname verwenden, z. B.</p> <pre><xsl:processing-instruction name="DynamicParameter"> documentname=" <xsl:value-of select="."/"/>" </xsl:processing-instruction></pre> <p>Mögliche Werte: Beliebige Zeichenkette, z. B. <i>Data.xml</i></p>
Custom Parameters	Benutzerdefinierte Parameter setzen, siehe Benutzerdefinierte Parameter verwenden
Operation	Bestimmt, welche Transfer-Operation mit den oben festgelegten Dateien durchgeführt wird. Je nach Adapter unterscheiden sich die Werte und Funktionsweisen.
Delete	Quelldatei(en) nach erfolgreich abgeschlossener Verarbeitung löschen (abhängig vom jeweiligen Transfer-Adaptertyp)
I/O Mode	<p>Art der Quelle der zu verarbeitenden Daten</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> Not assigned: Nicht zugewiesen; Hierbei wird die Komponente mit einem  im Prozessdiagramm gekennzeichnet. Actions: Einzelne Datei verarbeiten, deren Name in Eigenschaft <i>Filename</i> angegeben wird Directory: Alle Dateien eines angegebenen Ordners verarbeiten (oder Dateien gemäß Filterregel in <i>Parameter filter</i>)

1.4.3.3 Adapter



Adapter : Stellt grundlegende oder zusätzliche Funktionen bereit, etwa Zeichensatz-Konvertierungen oder ermöglicht beispielsweise den Zugriff auf ein SAP-System. Da der X4 Designer hauptsächlich mit XML-Daten arbeitet, werden mit Adaptern meist XML-Dokumente erzeugt oder verändert. Als Ausgabe erzeugen Adapter meist XML-Daten, die in einem nächsten Prozess-Schritt weiterverarbeitet werden.

-  Um den Adaptertyp zu wählen und Parameter festzulegen, doppelklicken Sie auf der Zeichnungsfläche auf das entsprechende Funktions-Adapter-Symbol. Die spezifische Bedeutung der im folgenden genannten Eigenschaften der Funktions-Adapter ist in der jeweiligen Adapter-Referenz beschrieben.

Eigenschaften

Vesion	Adapter-Version
--------	-----------------

Document	Pfad zum verknüpften Adapter im Repository
Custom Parameters	Benutzerdefinierte Parameter setzen, siehe Benutzerdefinierte Parameter verwenden
Operation	Bestimmt, welche Operation mit dem Funktions-Adapter durchgeführt wird. Je nach Adapter unterscheiden sich die Werte und Funktionen.

Adapter anlegen und verwenden

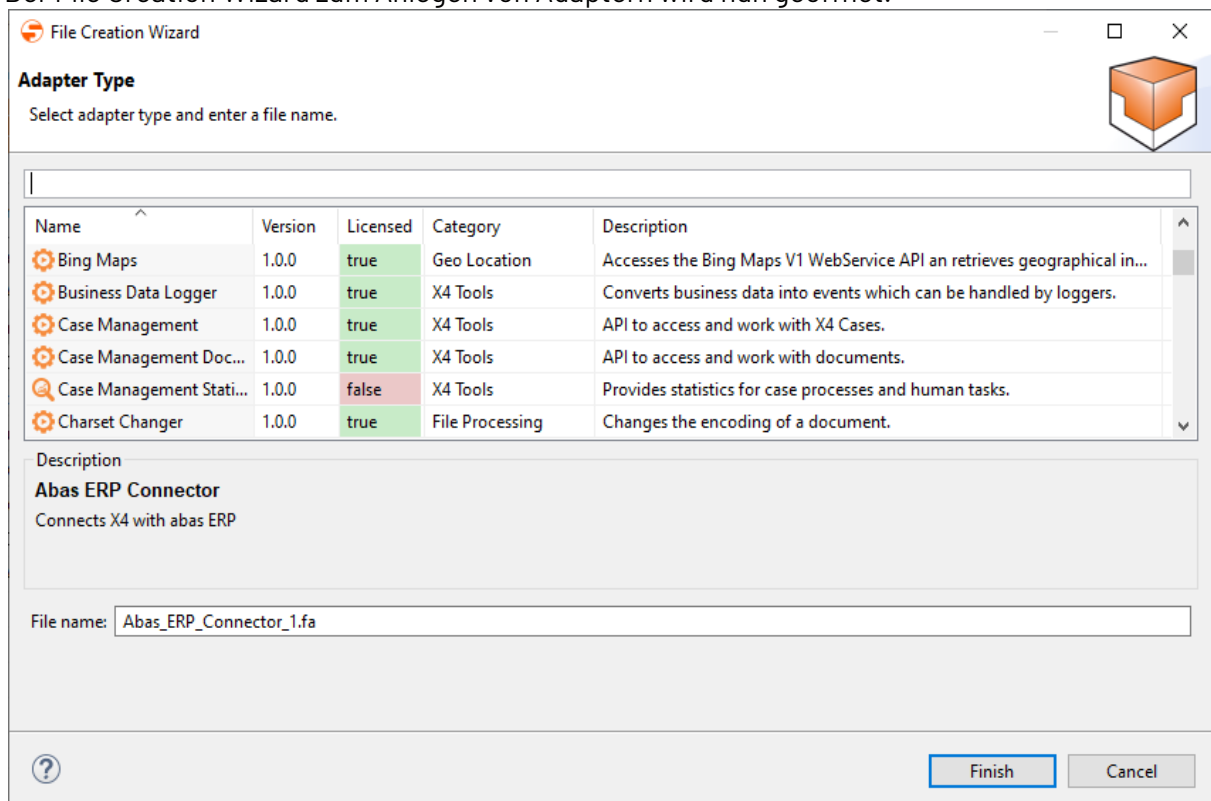
Adapter lassen sich innerhalb eines ESB-Projektes im Ordner Adapters anlegen. Hierzu stehen zwei Möglichkeiten zur Verfügung:

- über das Menü **Project > Add Adapter**
- über das Kontextmenü **New > Adapter**

Um einen Adapter anzulegen und zu parametrieren, gehen Sie wie folgt vor:

1. Innerhalb des ESB-Projektes auf den Ordner Adapters rechtsklicken, um das Kontextmenü aufzurufen.
2. Im Kontextmenü **New > Adapter** aufrufen.

Der File Creation Wizard zum Anlegen von Adaptern wird nun geöffnet.



File Creation Wizard

Adapter Type
Select adapter type and enter a file name.

Name	Version	Licensed	Category	Description
Bing Maps	1.0.0	true	Geo Location	Accesses the Bing Maps V1 WebService API and retrieves geographical in...
Business Data Logger	1.0.0	true	X4 Tools	Converts business data into events which can be handled by loggers.
Case Management	1.0.0	true	X4 Tools	API to access and work with X4 Cases.
Case Management Doc...	1.0.0	true	X4 Tools	API to access and work with documents.
Case Management Stati...	1.0.0	false	X4 Tools	Provides statistics for case processes and human tasks.
Charset Changer	1.0.0	true	File Processing	Changes the encoding of a document.

Description
Abas ERP Connector
Connects X4 with abas ERP

File name:

? Finish Cancel

3. Gewünschten Adapter aus der Liste wählen.

- ✓ Im Suchfeld können Sie die Adapternamen und Beschreibungen nach einem Stichwort durchsuchen und somit die angezeigten Adapter einschränken. Zudem können Sie die Adapter durch Klick auf den Spaltentitel **Category** nach Adapter-Kategorien sortieren.

4. In **File name** den Adapter-Namen eingeben.
5. **Finish** klicken, um den Adapter im Repository anzulegen.

Der Adapter wird im Adapter-Editor zur Konfiguration der Parameter geöffnet.

Adapter

Type: **Case Management** Category: **X4 Tools**

Version: 1.0.0

Description: API to access and work with X4 Cases.

Parameter Filter: Show All

Parameter	Value	Comment
P Adapter	de.softproject.x4.server.bpm.casemanagement.adapter.CaseManagementAdapter	Implementation class
P CaseId	%CASE_ID%	
P CaseProcess	%START_PROCESS%	
P CaseUserId	%USER_ID%	

Parameter

Name: **Adapter** Reset

Description: API to access and work with X4 Cases. Reset All

6. Im Adapter-Editor die Parameterwerte hinterlegen, mindestens jedoch die Pflichtparameter. Pflichtfelder erkennen Sie am Symbol

P

- ⓘ
- Setzen Sie alle Adapter-Parameter, die vertrauliche Informationen enthalten, immer dynamisch per XSL-Mapping, z. B. über die Processing Instruction `<?xsl:param name="..." value="...">`, siehe [Benutzerdefinierte Parameter verwenden](#). So werden die Zugangsdaten erst zur Laufzeit des Prozesses im Adapter gesetzt und die entsprechenden Parameter müssen nicht im Adapter angegeben werden und sind somit nicht einsehbar.
 - Für eine Beschreibung der Parameter für einzelnen Adapter, siehe [Übersicht der X4 Adapter](#).

7. Adapter speichern.
 8. Adapter per Drag&Drop aus dem Repository Navigator in den technischen Prozess einfügen.
 9. In der **Properties**-Sicht für den markierten Adapter in Eigenschaft **Operation** eine Adapter-Operation auswählen.
- Der Funktions-Adapter ist nun einsatzbereit.

10. Den technischen Prozess speichern.

1.4.3.4 Subprocess



Subprocess: Bietet die Möglichkeit, Teile von Prozessen bzw. Teilprozesse in separate Prozess-Dateien auszulagern. Sie können jeden Prozess in einen anderen Prozess als Subprozess einbinden. Dabei kann ein Subprozess einen Input aus dem vorangegangenen Prozessbaustein erhalten und gibt den letzten Status innerhalb des Subprozesses sowie ggf. ein Ausgabe-Dokument an den darauf folgenden Prozessbaustein im übergeordneten Prozess weiter.



- Um den Subprozess zu bearbeiten, auf das entsprechende Symbol auf der Zeichnungsfläche doppelklicken.
- Falls ein Input- bzw. Output-Schema beim Anlegen des Prozesses hinterlegt wurde, kann dieses über den Kontextmenü-Eintrag **Show input schema** bzw. **Show output schema** aufgerufen werden.

Eigenschaften

Document	Relativer Pfad zum Subprozess Dieser Pfad ist über den Parameter workflow dynamisch parameterisierbar. Die Verwendung von Platzhaltern (z. B. %START_PROCESS_PATH% oder %CALLING_PROCESS_PATH%) ist ebenfalls möglich (nicht im Debug-Modus des X4 Designers)
Label	Beschriftung des Symbols im Prozess-Diagramm

Teilprozesse als Subprozess auslagern

Bei umfangreicheren Prozessen ist es sinnvoll, zusammenhängende Bausteine als Subprozesse auszulagern. So wird der Prozess übersichtlicher und die Subprozesse lassen sich mehrfach verwenden. Prozesse lassen sich praktisch beliebig tief ineinander verschachteln. Subprozesse werden als separate .wrf-Dateien gespeichert.



Prozesse lassen sich wie andere Prozessbausteine per Drag&Drop aus dem Repository Navigator in die Zeichnungsfläche des Process Editors ziehen.

1. Mindestens 2 Bausteine inklusive aller Übergangslinien markieren, die ausgelagert werden sollen.
2. Funktion **Pack As Subprocess** im Kontextmenü aufrufen:
 - In **File Name** den Namen des Subprozesses eingeben.
 - **Finish** klicken.
3. Der Subprozess mit den markierten Bausteinen wird als eigenständige Prozess-Datei im Repository angelegt. Im Hauptprozess werden die gewählten Bausteine automatisch durch den neu angelegten Subprozess-Baustein ersetzt.

1.4.3.5 Mapping



Mapping: Transformiert ein XML-Dokument in einer Quellstruktur über ein XSLT-Stylesheet in eine Zielstruktur (meist XML) und ermöglicht so die einzelnen Datenelemente einander zuzuordnen (sog. *Mapping*). XSLT dient der X4 Suite zudem als Skriptsprache, mit der Prozessbausteine dynamisch parametrisiert werden können.

Eigenschaften

Document	Pfad zur verknüpften Datei im Repository
Label	Beschriftung des Symbols im Prozess-Diagramm
Custom Parameters	Benutzerdefinierte Parameter setzen; siehe Benutzerdefinierte Parameter verwenden .
Empty Input (X4EMPTYINPUT)	Keine Daten aus dem vorangehenden Prozess-Schritt im XSL-Mapping verarbeiten Mögliche Werte: <ul style="list-style-type: none"> • <i>true</i>: Keine Input-Daten verarbeiten • <i>false</i>: Input-Daten verarbeiten (Standard)
Skip Dynamic Params (X4SKIPDYNPARAM)	Auswertung von dynamischen Parametern abschalten. Dies kann die Ausführung von XSL-Mappings deutlich beschleunigen. Mögliche Werte: <ul style="list-style-type: none"> • <i>true</i>: Auswertung von dynamischen Parameter abschalten • <i>false</i>: Dynamische Parameter im XSL-Mapping verarbeiten (Standard)
Remove X4 Processing Instructions	X4 Processing Instructions im Mapping-Ergebnis entfernen. Mögliche Werte: <ul style="list-style-type: none"> • <i>true</i>: X4 Processing Instructions aus dem Ergebnis löschen (Standard) • <i>false</i>: X4 Processing Instructions im Ergebnis anzeigen

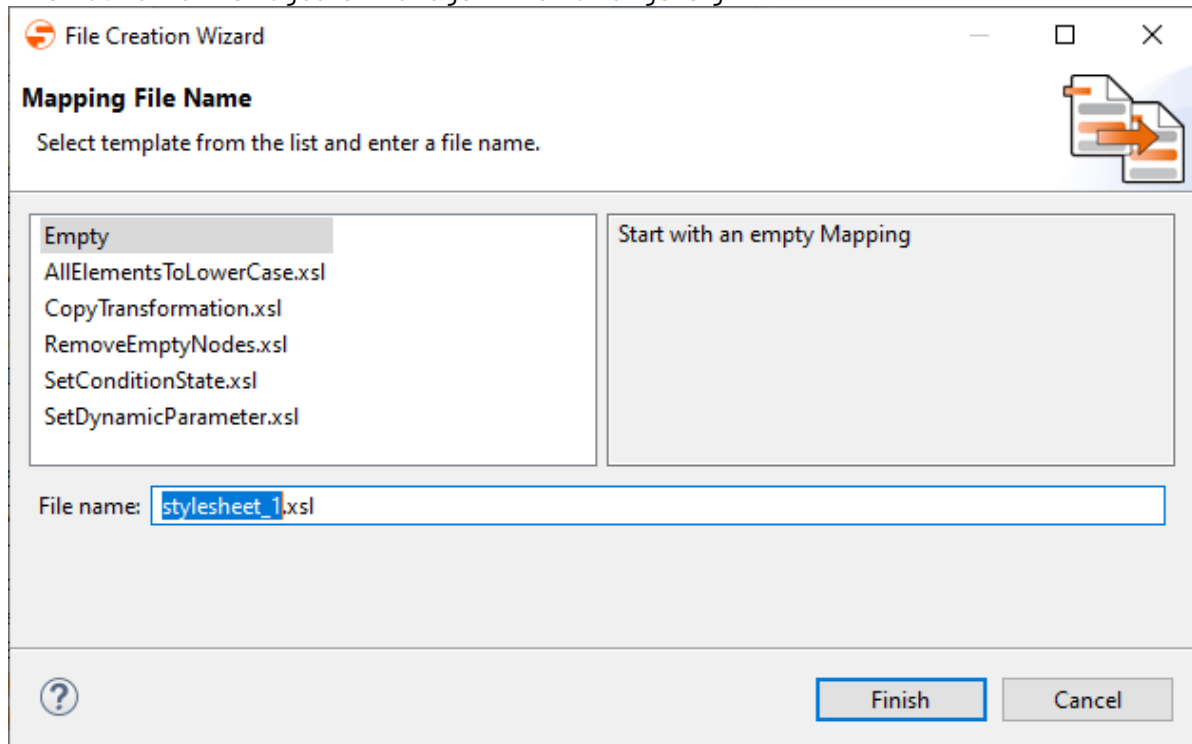
1.4.3.5.1 XSL-Mapping anlegen und verwenden

Mappings lassen sich innerhalb eines ESB-Projektes im Ordner Transformations anlegen. Hierzu stehen zwei Möglichkeiten zur Verfügung:

- über das Menü **Project > Add XSL Stylesheet**
- über das Kontextmenü **New > XSL Stylesheet**

1. Innerhalb des ESB-Projektes auf den Ordner Transformations rechtsklicken, um das Kontextmenü aufzurufen.

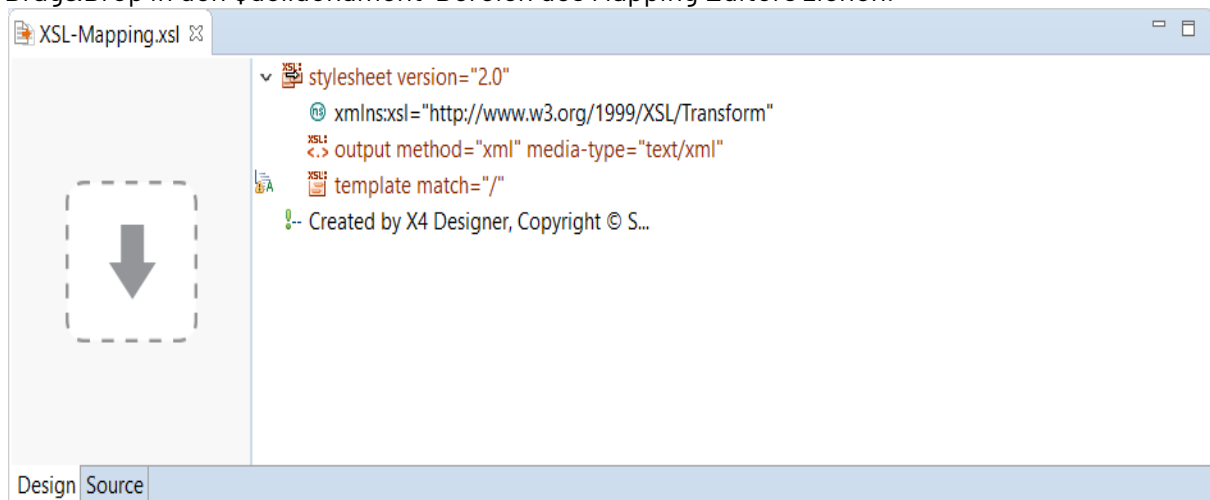
2. Im Kontextmenü **New> XSL Stylesheet** aufrufen.
Eine Auswahl an verfügbaren Vorlagen wird nun angezeigt.



3. Leere Vorlage oder eine vorhandene Vorlage wählen.
4. In **File Name** den Dateinamen eingeben.
5. **Finish** klicken, um das XSL-Mapping anzulegen.
Das XSL-Mapping wurde angelegt und wird im [Mapping Editor](#) geöffnet.

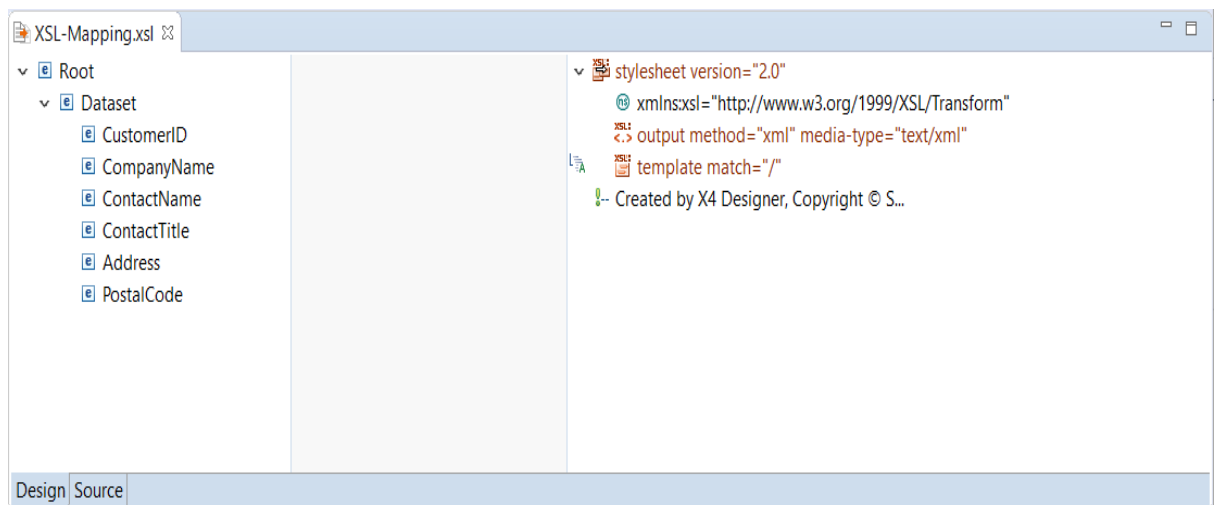
1.4.3.5.1.1 Quell- und Zieldatenstruktur definieren

1. Ein XML-Quelldokument (XML-Dokument oder XML-Schema) aus dem Repository per Drag&Drop in den Quelldokument-Bereich des Mapping Editors ziehen.



Alternativ kann das Quelldokument auch über das Kontextmenü **Assign Input** zugewiesen werden.

Das XML-Quelldokument wird nun in einer Baumansicht dargestellt:



2. Grundstruktur des Zieldokuments im XSL-Mapping anlegen:

- Im Stylesheet-Bereich auf das Root-Template (🔗 `template match="/"`) rechtsklicken, um das Kontextmenü zu öffnen.
- Im Kontextmenü **XSL Element** wählen, um den **XSL Element Creation Wizard** zu öffnen.
- Sämtliche benötigten Elemente mit der entsprechenden Hierarchie einfügen.
Beispiel: Die Struktur des Zieldokuments ist ein HTML-Grundgerüst (`<html>`, `<body>`), das eine Tabelle (`<table>`) mit einer Zeile (`<tr>`) mit drei Kopfzellen (`<th>`) und einer weiteren Zeile (`<tr>`) mit drei Zellen (`<td>`) enthält.

📘 Wenn Sie bereits die XML-Struktur kennen, die durch das XSL-Mapping erzeugt werden soll, dann können Sie per Drag&Drop aus dem Repository Navigator ein XML-Dokument oder ein XML-Schema zuweisen. Dieses können Sie dann als Zielstruktur für das Mapping verwenden.
Beim Zuweisen einer XML-Struktur per Drag&Drop werden Sie gefragt, wie diese eingefügt werden soll:

- **Full XML structure including data** fügt die komplette XML-Dokumentstruktur inklusive Element- und Attributwerte ein.
- **Virtual nodes (structure without data)** fügt lediglich die Baum-Struktur als virtuelle Knoten in den Stylesheet-Bereich ein. Erst per Doppelklick oder über ein Datenmapping auf einen virtuellen Knoten werden diese tatsächlich in der Zielstruktur erzeugt.

- Wenn Sie Text hinzufügen möchten: Im Kontextmenü **Add Other Nodes > Text** wählen.
Beispiel: Kopfzeilen-Beschriftung für die Tabellenkopfzellen einfügen.

1.4.3.5.1.2 Daten-Mapping: *for-each* verwenden

Mit `<xsl:for-each>` eine sich pro Datensatz wiederholende Zielstruktur aufbauen:

1. Im Quelldokument die wiederholt vorkommende Knotenstruktur markieren, **Shift**-Taste gedrückt halten und per Drag&Drop auf das zu wiederholende Zielelement ziehen. Ein `for-each`-Element wird erzeugt.

Beispiel: <Dataset> auf diese Weise mit dem zweiten <tr>-Element verknüpfen, um HTML-Tabellenzeilen dynamisch zu erzeugen.

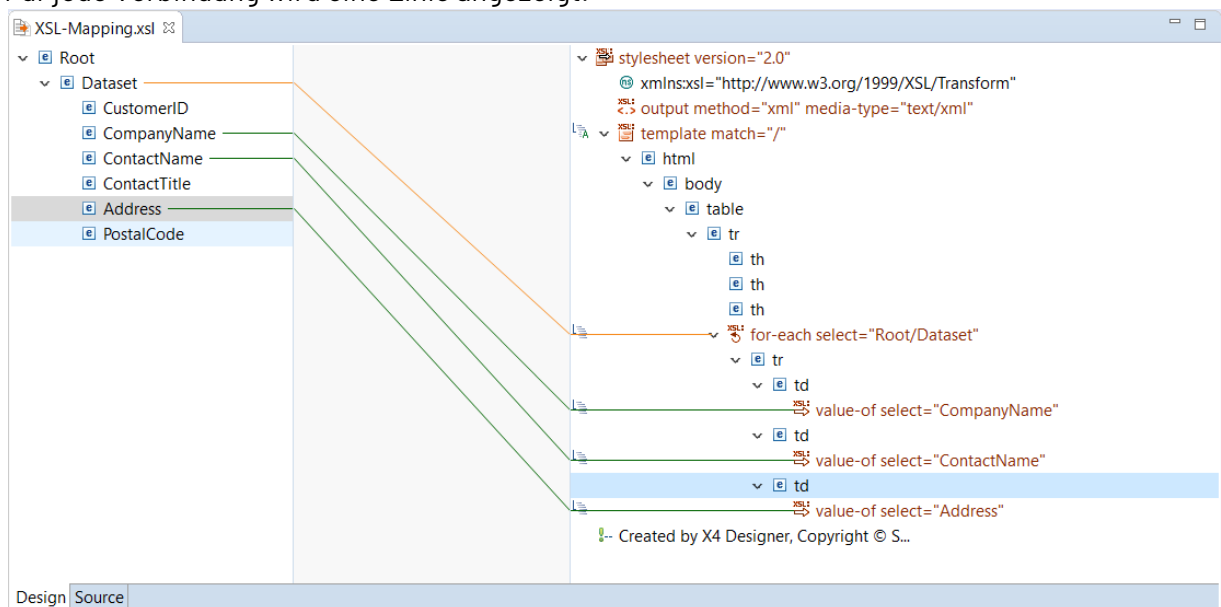
2. Sicherstellen, dass das for-each-Element unmittelbarer Elternknoten des zweiten <tr>-Elements ist.

1.4.3.5.1.3 Daten-Mapping: *value-of* verwenden

Mit <xsl:value-of> Knoteninhalte vom Quelldokument ins Zieldokument übernehmen:

1. Quellknoten anklicken und per Drag&Drop auf den Zielknoten ziehen.
Beispiel: <CustomerName>-Element dem ersten <td>-Element, <CompanyName> dem zweiten <td>-Element und <Address> dem dritten <td>-Element zuweisen.
2. Weitere Verbindungslinien ziehen.

Für jede Verbindung wird eine Linie angezeigt.



1.4.3.5.1.4 Daten-Mapping: *copy-of* verwenden

Mit <xsl:copy-of> einen Knoten inklusive aller Kindknoten (Elemente, Attribute, Namespace-Definitionen etc.) kopieren:

Im Quelldokument den zu kopierenden Knoten markieren, Strg-Taste gedrückt halten und per Drag&Drop auf das Ziel-Element ziehen.

Ein copy-of-Element wird erzeugt.

1.4.3.5.1.5 Source-Ansicht: XSLT-Funktionen verwenden

Beispiel

Die Elementinhalte von ContactName und ContactTitle sollen über die Funktion concat zusammengeführt werden.

1. Registerkarte **Source** wählen, um das XSL-Mapping im Quelltext anzuzeigen.

2. XSLT-Syntax bearbeiten.

Beispiel: Zeile `<xsl:value-of select="ContactName"></xsl:value-of>` durch `<xsl:value-of select="concat(ContactName, ', ', ContactTitle)"/>` ersetzen.

1.4.3.5.1.6 Ausgabeformat ändern

1. In Element `<xsl:output>` in Attribut `method="xml"` das Ausgabeformat anpassen.


Beispiel: Zum Erzeugen eines HTML-Dokuments `method="html"` setzen.

2. In Element `<xsl:output>` in Attribut `media-type="text/xml"` den MIME-Typ des Ausgabedokuments anpassen.

Beispiel: Zum Erzeugen eines HTML-Dokuments `mime-type="text/html"` setzen.

1.4.3.5.1.7 XSL-Mapping testen

Wenn ein Input-Dokument verfügbar ist, können Sie die integrierte Vorschaufunktion des Mapping Editors verwenden.

1. Das XSL-Mapping mit **Strg+S** speichern.
2. In der Symbolleiste  und anschließend **Run Transformation** klicken, um das XSL-Mapping lokal auszuführen.

Das Transformationsergebnis wird nun in der Output-Sicht angezeigt.

1.4.3.5.1.8 XSL-Mapping in einen Prozess einbauen


1. XSL-Mapping aus dem Repository Navigator per Drag&Drop in den geöffneten Prozess ziehen.
Beispiel: Zwischen XML-Dokument und Stop-Baustein einfügen.
2. Bei Bedarf in Eigenschaft **Custom Parameters** globale Parameter und deren Werte definieren. Diese werden beim Ausführen des Prozesses an XSL-Mapping übergeben, siehe [Prozessbausteine parametrieren](#).

1.4.3.5.2 Mapping Editor

Mit dem Mapping Editor lassen sich XSL-Mappings in einer grafischen Ansicht erstellen und bearbeiten. Eine XML-Datenstruktur kann hier mithilfe von XSLT in eine andere XML-Datenstruktur oder in ein XSL-FO-Dokument transformiert werden. Der Mapping Editor unterstützt Sie beim Erstellen eines XSL-Mappings durch eine grafische Oberfläche und durch eine automatische Gültigkeitsprüfung des Stylesheets sowie der XPath-Ausdrücke.

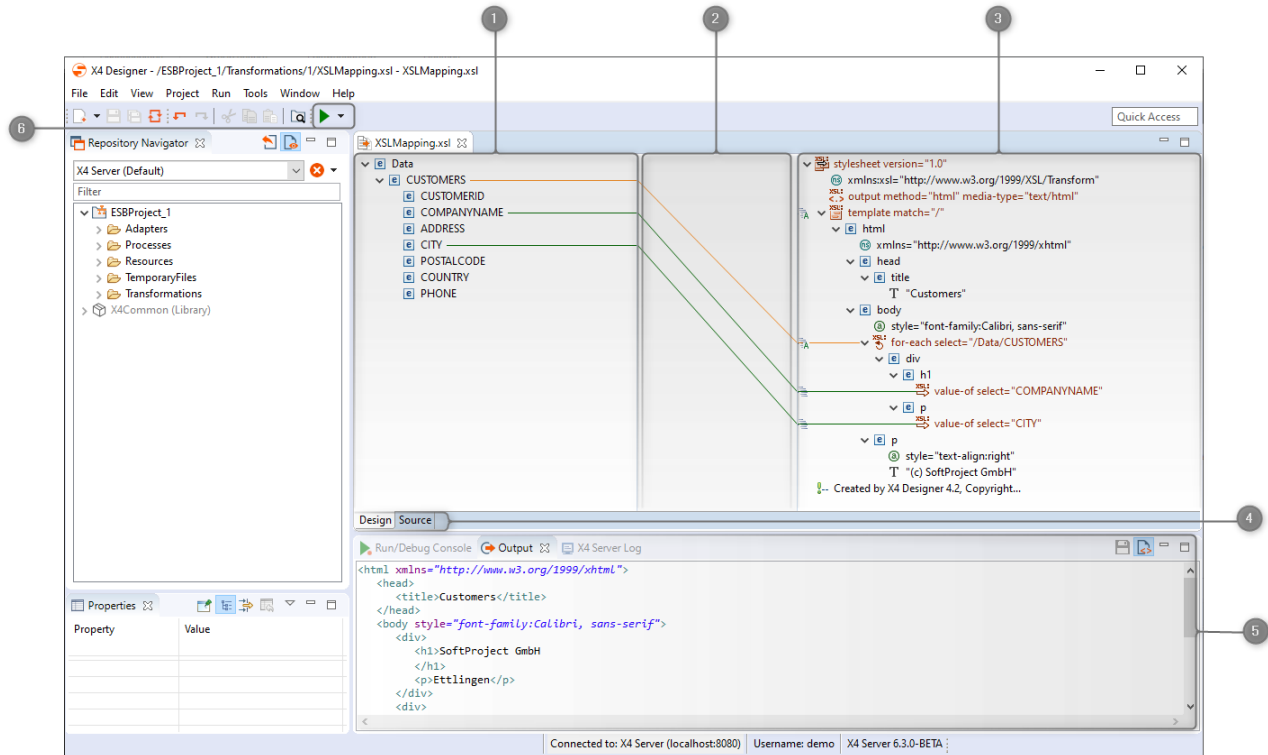
Bei einem XSL-Mapping sind drei Dokumente involviert:

- XML-Quelldokument (oder ein XML-Schema mit der Strukturinformation)
- XSLT-Stylesheet (dieses wird im Mapping Editor erstellt)
- XML-Ausgabedokument



 Um den Mapping Editor zu starten, legen Sie entweder ein neues Mapping an oder öffnen Sie ein bereits vorhandenes XSLT-Mapping.




Aufbau des Mapping Editors

Der Mapping Editor ist in folgende Bereiche unterteilt:



<p>1 Quelldokument-Bereich</p>	<p>Hier wird das zu transformierende Quelldokument oder eine XML-Schema-Datei (.xsd) angezeigt.</p> <p>Über das Kontextmenü können Sie den gewählten Knoten folgendermaßen bearbeiten:</p> <ul style="list-style-type: none"> • Copy XPath for this element to Clipboard: XPath-Ausdruck zum entsprechenden Element in die Zwischenablage kopieren und an anderer Stelle verwenden • Assign Input: Input-XML-Dokument zuweisen • Remove Input: Input-XML-Dokument entfernen
<p>2 Verknüpfungsbereich</p>	<p>Hier werden die Verknüpfungen zwischen Knoten im Quelldokument und XSL-Stylesheet als verschiedenfarbige Linien visualisiert.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p>i Unter Tastenkombinationen im Mapping Editor finden Sie nützliche Tastenkombinationen für die Arbeit mit XSL-Stylesheets in der grafischen Ansicht des X4 Mapping Editors.</p> </div>

3	Stylesheet-Bereich	<p>Hier bearbeiten Sie den Hauptbestandteil eines Mappings, das XSL-Stylesheet. Das Stylesheet wird in einer Baumstruktur angezeigt und bietet über das Kontextmenü Zusatzfunktionen für die XSL-Bearbeitung.</p> <p>Die verschiedenen Bestandteile des XSL-Stylesheets werden in der Baumansicht mit Symbolen und farblichen Hervorhebungen dargestellt:</p> <ul style="list-style-type: none"> • Knoten mit XSL-Anweisungen besitzen eine braune Schriftfarbe und ein entsprechendes Symbol, z. B.  <code>for-each select="Data/Customers"</code> • Knoten mit einem XPath-Ausdruck besitzen ein entsprechendes XPath-Symbol (). Um den XPath-Ausdruck zu bearbeiten, klicken Sie auf dieses Symbol. • Virtuelle Knoten (aus der Quellstruktur übernommene, aber noch nicht im Stylesheet explizit vorhandene Knoten) werden grau dargestellt. <p>Über das Kontextmenü können Sie im Stylesheet-Bereich den gewählten Knoten folgendermaßen bearbeiten:</p> <ul style="list-style-type: none"> • Edit XSL Element: XSL-Element bearbeiten • Rename: Knoten umbenennen • Element: Element einfügen • Attribute: Attribut einfügen • Edit: (bei Processing Instructions, Namespace-Deklarationen): Knoteninhalt bearbeiten • XSL Element: XSL-Element einfügen <ul style="list-style-type: none"> • as Child: als Kind-Knoten • Before: als Knoten vor dem markierten Knoten • After: als Knoten hinter dem markierten Knoten • Add other nodes: Andere Knoten einfügen (Namespace, Text, Comment, Processing Instruction) • Add here: (nur bei virtuellen Knoten): Virtuellen Knoten in tatsächlichen Knoten umwandeln • Add X4 Processing Instructions: X4-spezifische Processing Instructions hinzufügen, um Parameter dynamisch zu setzen. <ul style="list-style-type: none"> • Condition State für Status-Informationen • Log Info für Log-Nachrichten • Dynamic Parameter zum Setzen von dynamischen Werten • No Result für leere Mapping-Ausgaben • Add X4 Placeholder: X4-spezifische Platzhalter einfügen • Add Parent: Eltern-Element (übergeordnetes Element) einfügen • Cut/Copy/Paste: Auswahl ausschneiden, kopieren und einfügen • Delete: Knoten löschen • Expand/Collapse: Unterelemente des Knotens aufklappen bzw. zuklappen • Assign Virtual Nodes: Virtuellen Knoten aus einem XML-Dokument einfügen
---	--------------------	--

		<ul style="list-style-type: none"> • Remove Virtual Node: Eingefügten virtuellen Knoten entfernen <div style="border: 1px solid green; padding: 10px; margin-top: 10px;"> <p>✓ Mit dem XSL Element Creation Wizard lassen sich XSL-Mappings einfach und schnell erstellen. Der Wizard bietet neben der kompletten Palette an XSLT 2.0-Elementen auch die Möglichkeit, diese bequem über das Kontextmenü anzulegen.</p> <p>Über den Wizard lässt sich festlegen, ob das neue Element vor, nach oder als Kind des ausgewählten Elements angelegt werden soll. Funktionen wie Autovervollständigung oder die Anzeige aller verfügbaren und erlaubten XSL-Elemente ermöglichen schnelleres Arbeiten.</p> <p>Zudem können die Attribute des XSL-Elements gesetzt bzw. nach dem Anlegen bearbeitet werden. Zwingend erforderliche Attribute werden dabei automatisch übernommen, optionale Attribute können je nach Bedarf ausgewählt und befüllt werden.</p> <p>Bei Attributen, die mehrere Optionen zur Auswahl haben, wird eine Liste mit den verfügbaren Werten bereitgestellt.</p> </div>
4	Source- und Design-Ansicht	<p>Der Mapping Editor verfügt über zwei Ansichten, zwischen denen Sie über zwei Registerkarten wechseln können:</p> <ul style="list-style-type: none"> • Design: In der dreispaltigen Design-Ansicht wird das XSLT-Stylesheet als Baum dargestellt. Hier können die Knoten des Quelldokuments den Knoten im XSLT-Stylesheet grafisch zugeordnet werden. • Source: In dieser Quellcode-Ansicht kann das XSLT-Stylesheet direkt im XML-Editor bearbeitet werden.
5	Output	<p>Das Ergebnis von lokal ausgeführten XSL-Mappings wird in der Output-Sicht angezeigt.</p> <p>Über die Symbolleiste lassen sich zusätzlich folgende Aktionen vornehmen:</p> <ul style="list-style-type: none"> •  <i>Save content of output view:</i> Daten aus der Output-Sicht speichern •  <i>Format the content of the output view:</i> Daten aus der Output-Sicht formatieren
6	Transformation ausführen	<p>Über  Run Transformation wird das XSL-Mapping lokal ausgeführt und das Transformationsergebnis in der Output-Sicht (5) angezeigt.</p>

1.4.3.5.3 Tastenkombinationen im Mapping Editor

Nützliche Tastenkombinationen für die Arbeit mit XSL-Stylesheets in der grafischen Ansicht des X4 Mapping Editors

Aktionen per Drag&Drop

Werte mit dem Zielknoten verknüpfen mit <code>xsl:value-of</code>	Knoten im Quelldokument per Drag&Drop auf den Zielknoten ziehen	
Mehrere Werte im Zielknoten verknüpfen mit <code>xsl:value-of</code>	<p>Alt-Taste drücken und alle gewünschten Knoten im Quelldokument auf den Zielknoten ziehen</p> <p>Ergebnis: im <code>select</code>-Attribut des erzeugten <code>xsl:value-of</code>-Elements werden alle verknüpften Elemente nacheinander hinterlegt und können anschließend mit einer XPath-Funktion wie <code>concat()</code> oder logischen Operatoren verknüpft werden.</p> <p><i>Beispiel:</i> <code><xsl:value-of select="Element1 Element2"></code></p>	
Knoten kopieren mit <code>xsl:copy-of</code>	Strg -Taste drücken und Knoten im Quelldokument auf den Zielknoten ziehen	
For-Each-Schleife zu Knoten erstellen mit <code>xsl:for-each</code>	Shift -Taste drücken und Knoten im Quelldokument auf den Zielknoten ziehen	
	Funktion	Tastenkombination
Basis-Funktionen	Element bearbeiten	Enter
	XSL-Transformation ausführen	F9
	Element einfügen	Einfg
	Attribut einfügen	Alt+Einfg
	Text einfügen	Strg+Einfg
	Umschließendes Eltern-Element einfügen	Strg+Shift+Einfg
	Element umbenennen	Alt+Shift+R
	Markierten Knoten entfernen	Entf
	Code formatieren/einrücken (in der Source-Ansicht)	Strg+Shift+F
Knoten kopieren	Markierten Knoten als vorangehenden Knoten einfügen	Strg+Alt+Nach oben
	Markierten Knoten als nachfolgenden Knoten einfügen	Strg+Alt+Nach unten
	Markierten Knoten als vorangehenden Eltern-Knoten einfügen	Strg+Alt+Nach links

Knoten verschieben	Markierten Knoten als vorangehenden Knoten verschieben	Alt+Nach oben
	Markierten Knoten als nächsten Knoten verschieben	Alt+Nach unten
	Markierten Knoten als vorangehenden Eltern-Knoten verschieben	Alt+Shift+Nach links
Im Stylesheet-Baum navigieren	Zum Eltern-Element wechseln	Strg+Shift+Nach links
	Zum nächsten Geschwister-Knoten wechseln	Strg+Shift+Nach unten
	Zum vorangehenden Element wechseln	Strg+Shift+Nach oben
	Verbundene Knoten in Quelldokument-Bereich und Stylesheet-Bereich markieren	Shift+Eingf
	Letzte Markierung wiederherstellen	Alt+Shift+Nach unten

1.4.3.6 Report Generator



Report Generator: Transformiert ein XML-Dokument mithilfe eines XSLT-Stylesheets in ein XSL-FO-Dokument. Dieses wird im selben Prozess-Schritt über den XSL-FO-Prozessor *Apache FOP 1.0* in ein seitenorientiertes Ausgabemedium umgewandelt. Mögliche Ausgabeformate für Reports sind PDF, PDF/A und PostScript.

Eigenschaften

Document	Pfad zur verknüpften Datei im Repository
Custom Parameters	Benutzerdefinierte Parameter; siehe Benutzerdefinierte Parameter verwenden
Report type	<p>Ausgabeformat des Reports</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>PDF</i>: PDF-Dokument (Portable Document Format) • <i>PDF/A</i>: PDF-Dokument im Archivierungsformat <i>PDF/A-1b</i> gemäß <i>ISO 19005-1</i>; siehe Hinweis unten • <i>PS</i>: PostScript-Datei <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p>i Für die PDF/A-Option fordert Apache FOP 1.0 für jeden Textblock innerhalb des FO-Dokuments explizit die Festlegung einer Schriftart über das Attribut <code>font-family</code>; zudem müssen sämtliche verwendeten Schriftarten auf dem System während der PDF/A-Erzeugung verfügbar sein.</p> </div>

Reports erstellen mit XSL-FO

Mit dem Report-Baustein können Sie zur Laufzeit eines Prozesses dynamisch Dokumente für die Druckausgabe generieren. Ein Report ist ein spezielles XSL-Mapping, das XML-Daten transformiert und ein XML-Dokument mit XSL-FO-Anweisungen ausgibt. Das erzeugte XSL-FO-Dokument wird der X4 Suite direkt in das angegebene Ausgabeformat umgewandelt und an den nachfolgenden Prozess-Schritt weitergegeben.

Die X4 Suite verwendet Apache FOP als XSL-FO-Prozessor zum Verarbeiten der Reports.

Über das XSL-FO-Element `<fo:external-graphic src="url('Pfad/zur/Datei.png')">` können Grafikdateien (z. B. GIF, JPEG, PNG, SVG) ins Report-Dokument eingebunden werden. Um Grafiken aus dem Repository der X4 Suite einzubinden, können Sie deren Pfad mit einer `xstore:/`-URL angeben, z. B.:

```
<fo:external-graphic src="url('xstore://Project/Folder/File.png')"  
  content-width="10mm" content-height="10mm"/>
```

Mit den optionalen Attributen `content-width` und `content-height` können Sie Breite und Höhe des Bilds im Ausgabedokument bestimmen.

- ❗ Alternativ können Sie binäre Grafikdateien auch als Base64-kodierte Zeichenkette in den Report nach folgendem Muster einbetten; siehe [Base64 Converter](#):

```
<fo:external-graphic src="url(data:;base64,Base64-kodierte Bilddaten)"/>
```

Bei SVG-Bildern kann zudem deren Quellcode direkt in den Report eingebettet werden, z. B.:

```
<fo:instream-foreign-object>  
  <svg:svg xmlns:svg="http://www.w3.org/2000/svg"  
    height="1mm" width="40mm">  
    <svg:line y1="0" y2="0" x1="0" x2="40mm"  
      style="stroke:#0000FF;stroke-width:0.5pt" />  
    </svg:svg>  
  </fo:instream-foreign-object>
```

Achten Sie hierbei darauf, dass alle SVG-Elemente den Namespace `svg:` besitzen.

Beispiel

Im Folgenden ein Beispiel für ein sehr einfaches (minimales) XSL-FO-Dokument:

```
<fo:root xmlns:fo="http://www.w3.org/1999/XSL/Format">
  <fo:layout-master-set>
    <fo:simple-page-master master-name="simple" page-height="29.7cm"
      page-width="21cm" margin-left="2.5cm" margin-right="2.5cm">
      <fo:region-body margin-top="3cm"/>
    </fo:simple-page-master>
  </fo:layout-master-set>
  <fo:page-sequence master-reference="simple">
    <fo:flow flow-name="xsl-region-body">
      <fo:block>Hello World</fo:block>
    </fo:flow>
  </fo:page-sequence>
</fo:root>
```

1.4.4 Drawing

Folgende Zeichnungselemente können Sie in technischen Prozessen verwenden:

1.4.4.1 Connector



Connector: Ermöglicht die logische Verbindung von beliebigen Elementen.




Connector-Verbindungslinien ausschließlich zur Visualisierung von logischen Zusammenhängen!



Bei der Ausführung des Geschäftsprozesses über den X4 Server werden ausschließlich Prozessbausteine in den Daten- und Kontrollfluss mit einbezogen, die in BPM-Prozessen über Sequenzfluss-Linien, siehe Sequence Flow, bzw. in X4-Prozessen über Transition-Übergangslinien verbunden sind, siehe [Transition](#).

Eigenschaften


Label	Freitextfeld zur Beschriftung der Verbindungslinie (zur übersichtlicheren Gestaltung bzw. Dokumentation)
Line Routing	Linienführung Mögliche Werte: <ul style="list-style-type: none"> • <i>Angled</i>: Angewinkelte Linie • <i>Straight</i>: Gerade Linie (Standard)
Line Type	Linientyp Mögliche Werte: <ul style="list-style-type: none"> • <i>Dash line</i>: Gestrichelte Linie (Standard) • <i>Solid line</i>: Durchgezogene Linie • <i>Dot line</i>: Gepunktete Linie

Line Color	<p>Farbe der Verbindungslinie; Um die Farbe zu ändern, die Eigenschaft markieren und auf  klicken.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>RGB(RRR, GGG, BBB)</i>: 24-bit-RGB-Farbcode (über den Farbauswahldialog wählbar) • <i>RGB(0, 0, 0)</i>: Schwarz (Standard)
Line Source Decorator und Line Target Decorator	<p>Symbol für den Linienanfang</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Circle</i>: Kreis (Standard) • <i>Empty arrow</i>: Ungefüllter Pfeil • <i>Association arrow</i>: Offener Pfeil • <i>Arrow</i>: Gefüllter Pfeil • <i>Inverted arrow</i>: Umgekehrter Pfeil • <i>Diamond</i>: Gefüllte Raute • <i>Empty diamond</i>: Ungefüllte Raute • <i>Plain</i>: Kein Symbol

1.4.4.2 Rectangle und Ellipse

 Rectangle	Zeichnet ein beliebiges Rechteck
 Ellipse	Zeichnet eine beliebige Ellipse

Eigenschaften



Line Color	<p>Rahmenfarbe; um die Farbe zu ändern, die Eigenschaft markieren und auf  klicken.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>RGB(RRR, GGG, BBB)</i>: 24-bit-RGB-Farbcode (über den Farbauswahldialog wählbar) • <i>RGB(0, 0, 0)</i>: Schwarz (Standard)
------------	---


Line Width	Linienbreite des Zeichnungselements Mögliche Werte: <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 1: 1 Pixel breite Linie (Standard)
Hide Line	Rahmenlinie ausblenden
Fill Color	Füllfarbe des Zeichnungselements; Um die Farbe zu ändern, die Eigenschaft markieren und auf  klicken. Mögliche Werte: <ul style="list-style-type: none"> • <i>RGB(RRR, GGG, BBB)</i>: 24-bit-RGB-Farbcode (über den Farbauswahldialog wählbar) • <i>RGB(255, 255, 255)</i>: Weiß (Standard)
Transparent	Zeichnungselements als transparent kennzeichnen. Durch Aktivieren der Eigenschaft, wird die Eigenschaft Fill Color aufgehoben.

1.4.4.3 Textbox

Abc Textbox: Fügt einen Textrahmen ein, dessen Inhalt Sie per Doppelklick in einem Dialogfenster bearbeiten und formatieren können.

Eigenschaften

Hyperlink	Hinterlegte Verlinkung für das Textfeld. Zum Verlinken eine gültige URL eintragen.  Mit Strg + Klick lässt sich die eingetragene URL aufrufen.
Text	Inhalt des Textrahmens; um den Text zu ändern und zu formatieren, die Eigenschaft markieren und auf  klicken.
Rotation	Drehung des Textrahmens Mögliche Werte: <ul style="list-style-type: none"> • 0: Keine Drehung (Standard) • 90: Um 90° gegen den Uhrzeigersinn drehen • 180: Um 180° drehen • 270: Um 270° gegen den Uhrzeigersinn drehen



Fill Color	<p>Füllfarbe des Textfeldes; Um die Farbe zu ändern, die Eigenschaft markieren und auf  klicken.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • $RGB(RRR, GGG, BBB)$: 24-bit-RGB-Farbcode (über den Farbauswahldialog wählbar) • $RGB(255, 255, 255)$: Weiß (Standard)
Transparent	Textfeld als transparent setzen. Durch Aktivieren der Eigenschaft, wird die Eigenschaft Fill Color aufgehoben.

1.4.4.4 Polygon





Polygon: Zeichnet ein Polygon mit beliebig vielen Eckpunkten.


Eigenschaften

Line Color	<p>Rahmenfarbe; um die Farbe zu ändern, die Eigenschaft markieren und auf  klicken.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • $RGB(RRR, GGG, BBB)$: 24-bit-RGB-Farbcode (über den Farbauswahldialog wählbar) • $RGB(0, 0, 0)$: Schwarz (Standard)
Line Width	<p>Linienbreite des Zeichnungselements</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 1: 1 Pixel breite Linie (Standard)
Fill Color	<p>Füllfarbe des Zeichnungselements; um die Farbe zu ändern, die Eigenschaft markieren und auf  klicken.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • $RGB(RRR, GGG, BBB)$: 24-bit-RGB-Farbcode (über den Farbauswahldialog wählbar) • $RGB(255, 255, 255)$: Weiß (Standard)
Transparent	Transparente Füllung (keine Füllfarbe verwenden). Durch Aktivieren der Eigenschaft, wird die Eigenschaft Fill Color aufgehoben.

1.4.4.5 Polyline und Line

 Polyline	Zeichnet eine Linie mit beliebig vielen Wegpunkten
 Line	Zeichnet eine gerade Linie

Eigenschaften

Line Color	<p>Linienfarbe; um die Farbe zu ändern, die Eigenschaft markieren und auf  klicken.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • $RGB(RRR, GGG, BBB)$: 24-bit-RGB-Farbcode (über den Farbauswahldialog wählbar) • $RGB(0, 0, 0)$: Schwarz (Standard)
Line Width	<p>Linienbreite des Zeichnungselements</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 1: 1 Pixel breite Linie (Standard)


1.4.4.6 Image



Image: Fügt einen Bildrahmen ein, mit dem sich eine Grafikdatei vom Typ .gif, .jpg, .png oder .bmp einbinden lässt.

 Das eingefügte Bild wird im Prozess gespeichert.

Eigenschaften

Line Color	<p>Linienfarbe des Bildrahmens; Um die Farbe zu ändern, die Eigenschaft markieren und auf  klicken.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • $RGB(RRR, GGG, BBB)$: 24-bit-RGB-Farbcode (über den Farbauswahldialog wählbar) • $RGB(0, 0, 0)$: Schwarz (Standard)
------------	---

Line Width	Linienbreite des Zeichnungselements Mögliche Werte: <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 1: 1 Pixel breite Linie (Standard)
Image File	Pfad der verknüpften Grafikdatei

1.5 Tastenkombinationen im Process Editor

Alle Tastenkombinationen für die Modellierung von IT-Geschäftsprozessen im Process Editor

	Funktion	Tastenkombination oder Aktion
Prozess ausführen und debuggen	Prozess ausführen	F9
	Prozess auf Fehler prüfen (und ggf. aktiven Debugging-Vorgang abbrechen)	F4
	Beim Debuggen Einzelschritt ausführen und anzeigen (auch in Subprozessen)	F5
	Beim Debuggen Prozess-Schritt ausführen (ohne Schritte des Subprozess anzuzeigen)	F6
	Beim Debuggen aus Subprozess herausspringen	F7
	Debugging-Vorgang anhalten	Strg+F8
	Debugging fortsetzen	F8
	Debugging abbrechen	Strg+F2
Prozessbausteine und Übergänge einfügen	Prozessbaustein zwischen zwei verbundenen Bausteinen aus dem Repository einfügen	Datei per Drag&Drop aus dem Repository auf den Übergang ziehen; Wenn die Linie des Übergangs dick wird, loslassen.
	Prozessbaustein von der Übergangslinie herunterziehen	Alt-Taste drücken und Prozessbaustein auf der Zeichnungsfläche verschieben
	Prozessbaustein entfernen und Übergangslinie behalten	Alt+Entf
	Prozessbaustein durch eine Datei aus dem Repository ersetzen	Datei per Drag&Drop aus dem Repository auf den zu ersetzenden Prozessbaustein ziehen


	Prozessbaustein hinter einem unverbundenen Baustein einfügen und dabei automatisch einen Übergang herstellen	Unverbundenen Prozessbaustein markieren, Shift-Taste drücken und die Datei aus dem Repository hinter den markierten Prozessbaustein ziehen
Elemente bearbeiten	Alles auf der Zeichnungsfläche auswählen	Strg+A
	Prozessbaustein ausschneiden	Strg+X
	Prozessbaustein kopieren	Strg+C
	Prozessbaustein einfügen	Strg+V
	Prozessbaustein entfernen	Entf
	Bearbeitungsschritt rückgängig machen	Strg+Z
	Bearbeitungsschritt wiederherstellen	Strg+Y
	Prozessbaustein bearbeiten	Auf Symbol doppelklicken
	Editor schließen	Strg+W
	Bearbeitetes Element speichern	Strg+S
	Datei im Repository umbenennen	F2
	Datei im Repository verschieben	Datei per Drag&Drop in den Ordner ziehen
	Neu laden	F5
Ansicht bearbeiten	Prozessdiagramm vergrößern (zoomen)	Strg++
	Prozessdiagramm verkleinern	Strg+-

2 Mit ESB-Projekten arbeiten

Das X4 Repository ist in Projekten organisiert. Projekte besitzen eine Dateisystem-Struktur und können beliebige Ordner, Prozesse, Prozessbausteine und andere Dateien enthalten.

2.1 Die Struktur von ESB-Projekten

ESB-Projekte haben eine vordefinierte und nicht veränderbare Ordnerstruktur, die beim Anlegen eines neuen ESB-Projektes automatisch angelegt wird.

 Die automatisch angelegten Ordner und Unterordner können nicht gelöscht, verschoben oder umbenannt werden.

Adapters	Hier können beliebig viele Adapter angelegt werden.
Processes	Hier können beliebig viele technische Prozesse (.wrf) angelegt werden.
Resources	Hier können neben Ordnern auch Ressourcen wie Bilder, Text- und XML-Dokumente sowie weitere Dateien abgelegt werden.
TemporaryFiles	Hier werden temporär verwendete Dateien abgelegt.
Transformations	Hier können Transformationen (.xsl) sowie Reports (.rep) abgelegt werden
<Projekt>.wsinc	Web service-Definition, die automatisch beim Anlegen des Projektes angelegt und nach dem Projekt benannt wird, z. B. <i>ESBProject_Documentation.wsinc</i> .

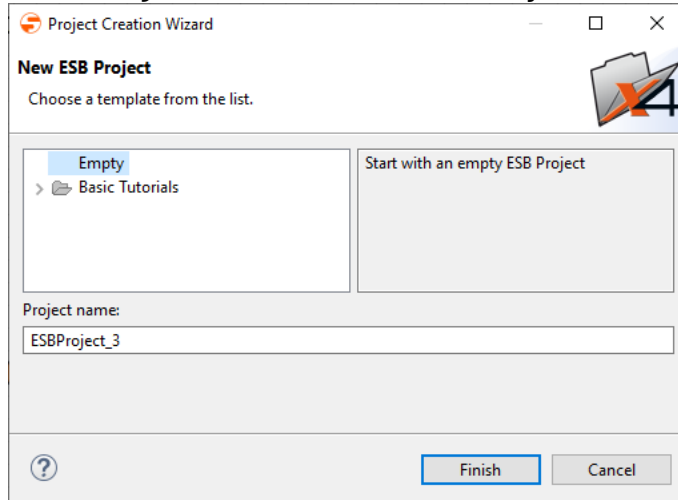
 Ein Projekt kann lediglich eine Web service-Definition enthalten.

2.2 ESB-Projekt anlegen


ESB-Projekte lassen sich auf verschiedene Arten anlegen:

- Über **New ESB Project** auf der Welcome-Seite des X4 Designers
- Über das Kontextmenü **New > ESB Project** im Repository Navigator
- Über das Menü **File > New > ESB Project**

1. Im X4 Designer Menü **File > New> ESB Project** wählen.




2. Leere Projektvorlage oder eine vorhandene Projektvorlage wählen.

 Es können auch Projekte auf Basis eines Templates aus einer älteren Version der X4 Suite erstellt werden. Diese werden während des Erstellens automatisch migriert.

3. In **Project name** den Namen des Projekts eingeben.

4. **Finish** klicken, um das Projekt anzulegen.

Ein leeres ESB-Projekt mit der vordefinierten Struktur wird im Repository Navigator angelegt.

 Über das Kontextmenü können nun die verschiedenen Elemente eines ESB-Projektes angelegt werden. Die verfügbaren Optionen ändern sich je nachdem an welcher Stelle in der Baumstruktur das Kontextmenü aufgerufen wird.

2.3 Projekte verwalten

Erfahren Sie hier, wie Sie Projekte im Repository Navigator verwalten.

2.3.1 Repository-Elemente importieren und exportieren

Hier erfahren Sie, wie Sie Projekte sowie verschiedene Ressourcen aus dem lokalen Dateisystem ins Repository importieren oder umgekehrt Repository-Dateien ins lokale Dateisystem exportieren.

Importieren via Drag&Drop

Aus dem Dateisystem (z. B. vom Desktop oder dem *Windows Explorer*) können Sie eine oder mehrere markierte Dateien per Drag & Drop in den Repository Navigator und damit ins aktuell verbundene Repository importieren. Falls eine Datei gleichen Namens am Ziel bereits vorhanden ist, werden Sie aufgefordert, einen neuen Namen einzugeben.

ⓘ Beachten Sie:

- Ordnerstrukturen werden beim Import nicht unterstützt, verwenden Sie hierzu den entsprechenden Wizard.
- Bei größeren Dateiübertragungen (insgesamt über 1 MB) wird ein Bestätigungsdialog eingeblendet, da möglicherweise Verzögerungen bei der Übertragung sowie Arbeitsspeicher-Probleme auftreten können.
- Importierte Dateien, die Sonder-/Leerzeichen oder Umlaute im Dateinamen enthalten, müssen zunächst umbenannt werden, bevor Sie diese im X4 Designer öffnen können.
- Projekte verfügen über eine vordefinierte Struktur, sodass die eingefügten Dateien nur im jeweils vordefinierten Ordner abgelegt werden können, siehe auch [Die Struktur von ESB-Projekten](#)

Import Wizard oder Export Wizard für Dateien und Projekte nutzen

ⓘ Beachten Sie

- Projekte verfügen über eine vordefinierte Struktur, sodass die zu importierenden Dateien nur im jeweils vordefinierten Ordner abgelegt werden können, siehe auch [Die Struktur von ESB-Projekten](#)
- Projekte können lediglich auf oberster Ebene und in der vordefinierten Projektstruktur importiert werden, siehe auch [Die Struktur von ESB-Projekten](#)
- Projekte können nur dann exportiert werden, wenn die oberste Projektebene ausgewählt ist.

1. Menü **Projekt > Import** bzw. **Projekt > Export** aufrufen.
Der **Resource Import Wizard** bzw. der **Repository Resource Export Wizard** wird aufgerufen.
2. Resource auswählen.
 - *Import:* Über **Browse** die zu importierende Datei (*.zip) oder das zu importierende Projekt wählen und in **New project/file name** den entsprechenden Namen eingeben.
 - *Export:* Über **Browse** den gewünschten Zielordner für den Export angeben.
3. **Finish** klicken.

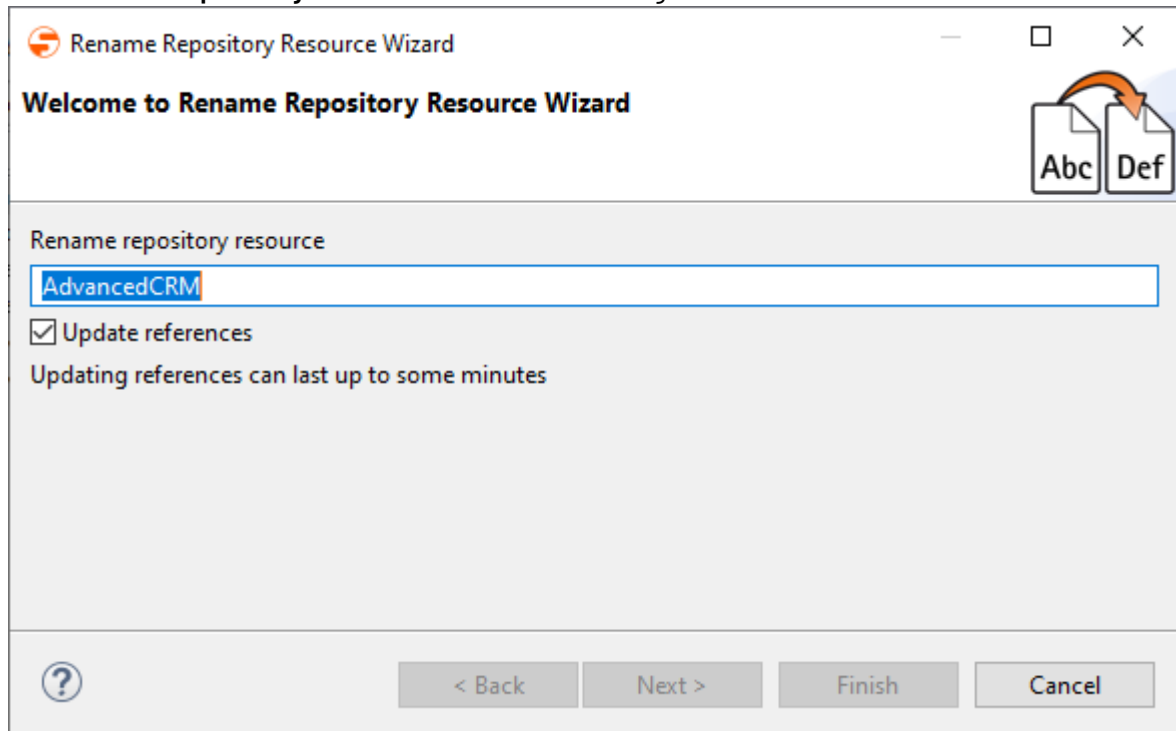
2.3.2 Repository-Elemente intelligent umbenennen

Wenn Sie Dateien im Repository umbenennen möchten, können Sie über den **Rename Repository Resource Wizard** die Referenzen in den Prozessen automatisch aktualisieren lassen.

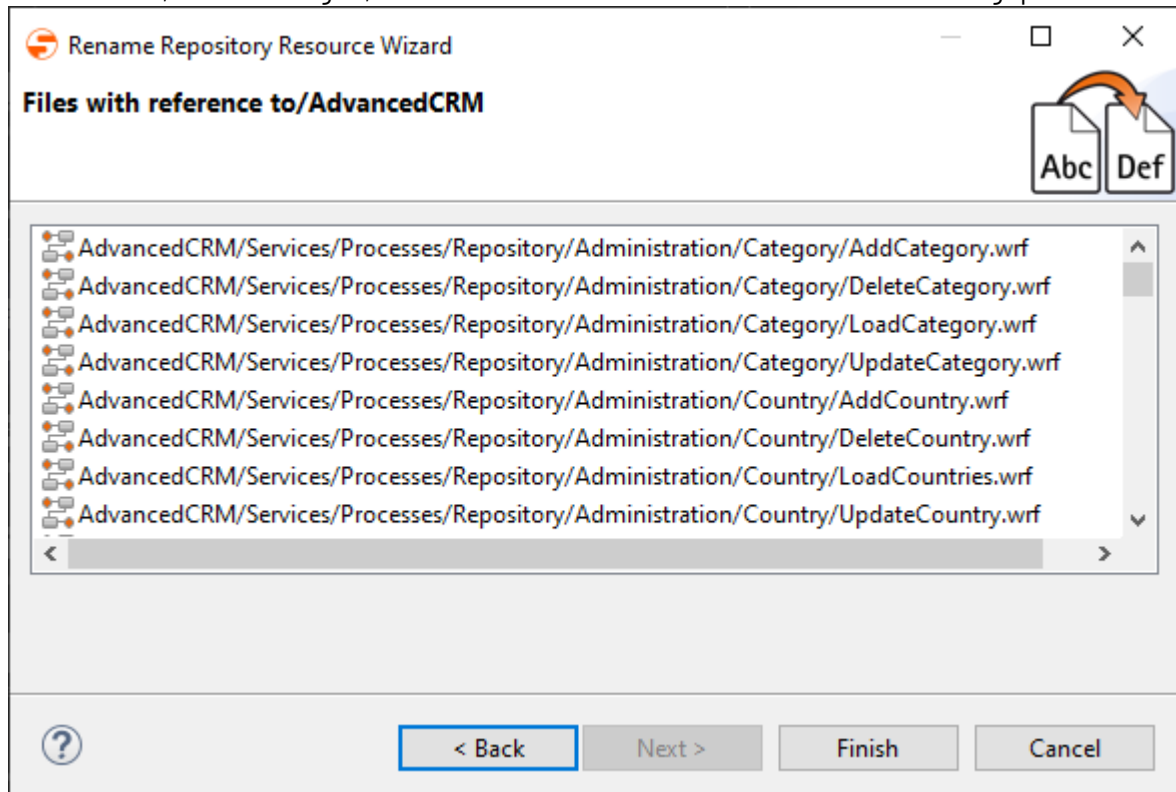
1. Auf das gewünschte Element im Repository Navigator rechtsklicken, um das Kontextmenü zu öffnen.

2. **Refactor/Rename** wählen.

Der **Rename Repository Resource Wizard** wird nun geöffnet.



3. In **Rename repository resource** einen neuen Dateinamen eingeben.
4. **Next** klicken, um anzuzeigen, welche Referenzen auf die umbenannte Datei angepasst werden.




5. **Finish** klicken, um die gewünschten Änderungen durchzuführen.
Die betroffenen Dateien werden nun umbenannt und die entsprechenden Referenzen zu Prozessen, Ressourcen oder Komponenten angepasst.

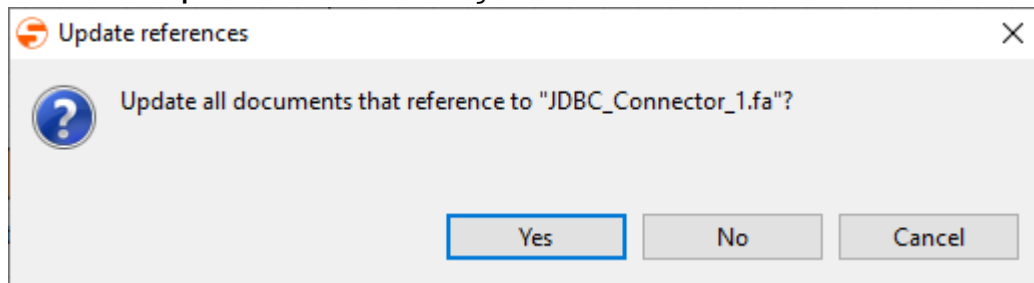
2.3.3 Repository-Elemente intelligent verschieben

Sie können Elemente innerhalb vom Repository verschieben. Dabei können die Referenzen auf die verschobene Datei in anderen Prozessen auf Wunsch automatisch angepasst werden.

1. Ressource an die gewünschte Stelle im Repository verschieben.

 Projekte verfügen über eine vordefinierte Struktur, sodass die eingefügten Dateien nur im jeweils vordefinierten Ordner abgelegt werden können, siehe auch [Die Struktur von ESB-Projekten](#)

Das Fenster **Update references** wird geöffnet.




2. Wenn sämtliche Referenzen auf die verschobene Datei bzw. den Ordner geändert werden sollen:

- **Yes** klicken.

Das Fenster **ConfirmMove** wird geöffnet. Hier wird angezeigt, von welchen Elementen im Repository die verschobene Datei verwendet wird.

- **OK** klicken, um die gewünschten Änderungen durchzuführen.

 Beim Verschieben von XSL-Mappings mit referenzierten Input- und Output-XML-Dokumenten/Schemas müssen diese Dateireferenzen manuell angepasst werden, falls sich deren relative Pfade geändert haben.

3. Wenn die Referenzen nicht aktualisiert werden sollen: **No** klicken.

Tipp

Wenn Sie die **Strg-/Alt**-Tasten drücken, können Sie auch mehrere Dateien markieren und verschieben. Für jede Datei wird dann das Fenster **Update references** aufgerufen. Dort erhalten Sie zusätzlich folgende Möglichkeiten:

- **Yes To All:** Referenzen für alle zu verschiebenden Dateien aktualisieren.
- **No To All:** Alle Referenzen nicht aktualisieren.

2.3.4 Repository-Referenzen reparieren

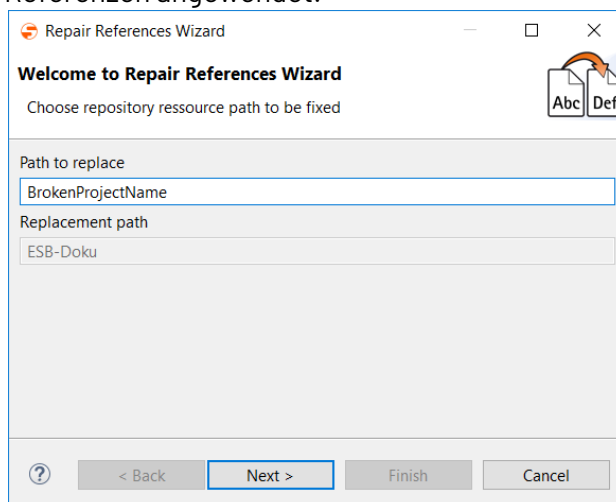
Wenn in einem Prozessdiagramm Repository-Elemente falsch referenziert sind – etwa weil ein Projekt oder Ordner umbenannt wurde oder von einer veralteten Vorlage erstellt wurde – so können Sie diese Referenzen über den **Repair References Wizard** reparieren lassen. Dabei werden die Pfadangaben für Prozessbausteine automatisch auf den tatsächlichen Pfad im Repository-Projekt angepasst.

1. Öffnen Sie einen technischen Prozess, Geschäftsprozess oder eine Prozesslandkarte in dem Referenzen auf Repository-Bausteine fehlerhaft (ausgegraut) sind und ermitteln Sie in der **Properties**-Sicht den inkorrekten Pfadbestandteil für einen markierten Prozessbaustein.

Beispiel

In einem Prozessdiagramm sind die Prozessbausteine inkorrekt referenziert: Deren Repository-Pfade verweisen auf das nicht vorhandene Projekt . Diese Pfadangaben sollen auf den tatsächlichen Projektnamen ProjectName korrigiert werden.

2. Auf das betroffene Projekt oder den betroffenen Ordner im Repository Navigator rechtsklicken, um das Kontextmenü zu öffnen.
3. **Repair References** wählen, um den **Repair References Wizard** aufzurufen.
Im Feld **Replacement path** wird nun das markierte Projekt bzw. der gewünschte Ordnername angezeigt. Dieser Pfadbestandteil wird im nächsten Schritt automatisch auf die fehlerhafte Referenzen angewendet.

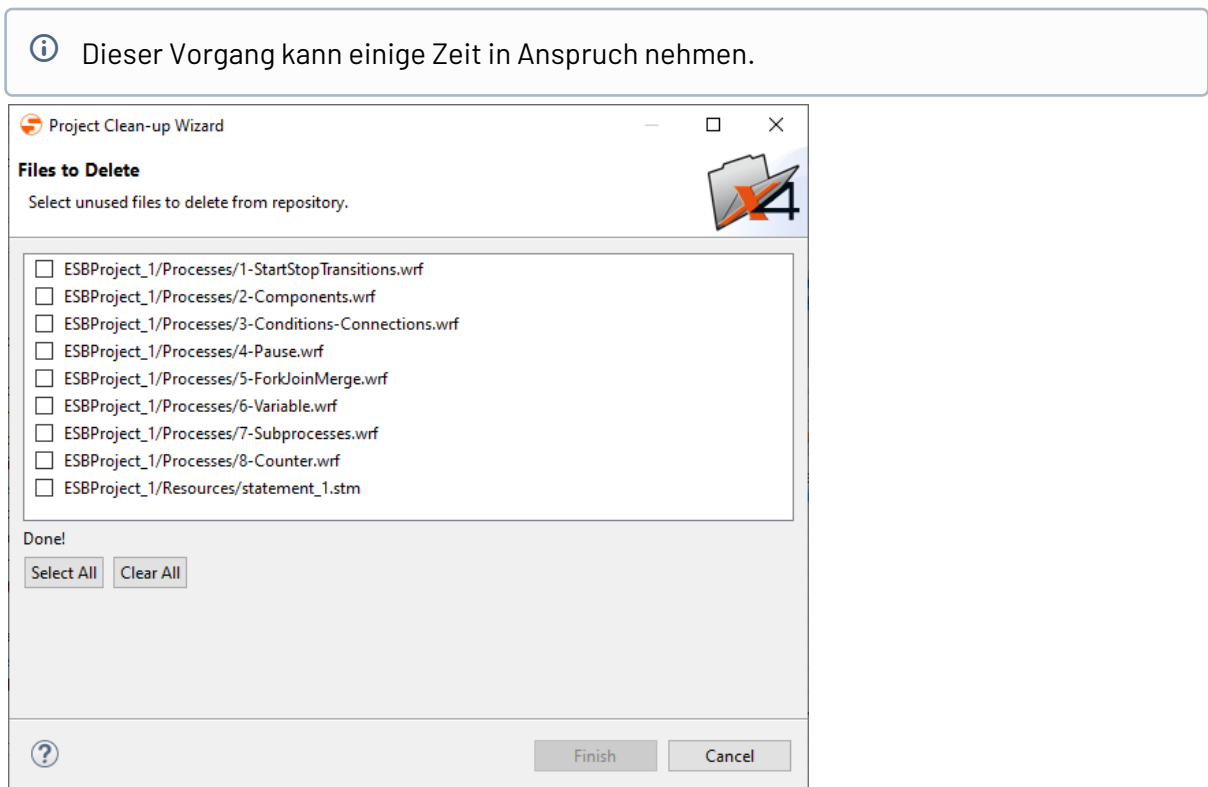


4. In **Path to replace** den fehlerhaften Projekt- oder Ordnernamen angeben, den Sie in Schritt 1 ermittelt haben.
5. **Next** klicken, um eine Liste von Prozessen anzuzeigen, welche die fehlerhaften Referenzen aufweisen und angepasst werden sollen.
6. **Finish** klicken, um die gewünschten Änderungen durchzuführen.

2.3.5 Nicht verwendete Ressourcen löschen

Wenn im Repository nicht verwendete Ressourcen vorhanden sind, so können Sie diese über den **Project Clean-up Wizard** auflisten und auf Knopfdruck entfernen.

1. Auf das betroffene Projekt im Repository Navigator rechtsklicken, um das Kontextmenü zu öffnen.
2. **Clean-up X4 Project** wählen, um den **Project Clean-up Wizard** aufzurufen.
Alle nicht verwendeten Dateien werden nun aufgelistet.



3. Gewünschte Ressourcen wählen.

⚠ Achtung: Möglicher Datenverlust!

Beachten Sie, dass es möglich ist, dass Dateien angezeigt werden, die z. B. als dynamischer Parameter in einem XSL-Mapping referenziert werden. Löschen Sie nur Dateien, von denen Sie sicher sind, dass Sie diese nicht mehr benötigen.

4. **Finish** klicken, und das endgültige Löschen anschließend mit **Yes** bestätigen.
Die Ressourcen wurden nun gelöscht.

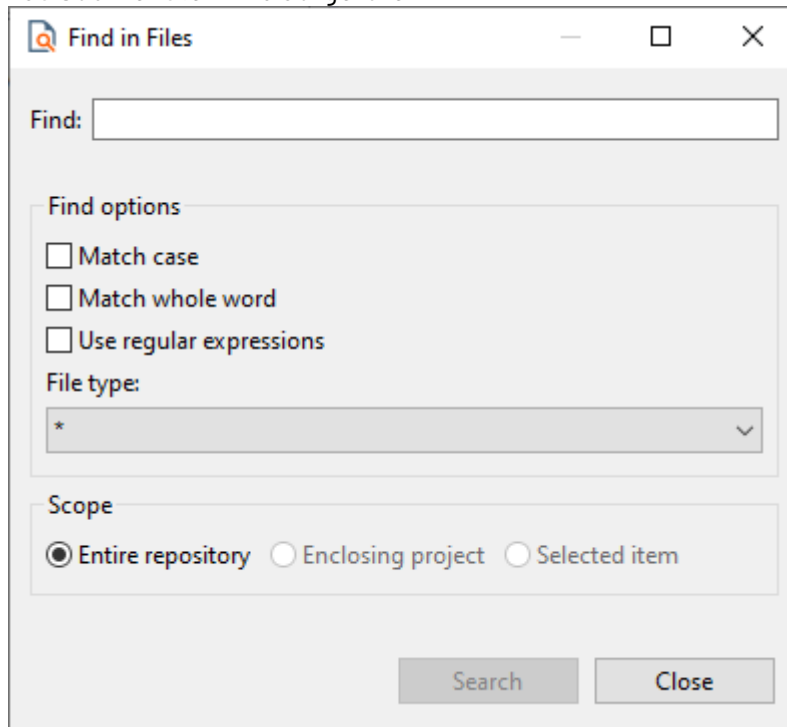
2.3.6 Repository-Elemente durchsuchen

Hier erfahren Sie, wie Sie innerhalb des Repository mit der Funktion **Find in Files** nach Texten, Regulären Ausdrücken und UTF-8-Symbolen suchen können.

i Bei der Suche werden lediglich Dateiinhalte durchsucht. Dateinamen werden bei der Suche nicht berücksichtigt.

1. Menü **Edit > Find in Files** aufrufen.


Das Suchfenster wird aufgerufen.



2. Bei **Find** Zeichen eingeben, nach dem gesucht werden soll.
3. Suchoptionen festlegen:
 - **Match Case:** Groß-/Kleinschreibung bei der Suche beachten.
 - **Match whole word:** Nach dem vollständigen Wort suchen.
 - **Use regular expression:** Nach regulären Ausdrücken suchen.
 - **File type:** Dateityp angeben, der durchsucht werden soll, z. B. bpm oder fa.

 Mit der Einstellung * werden alle Dateitypen durchsucht.

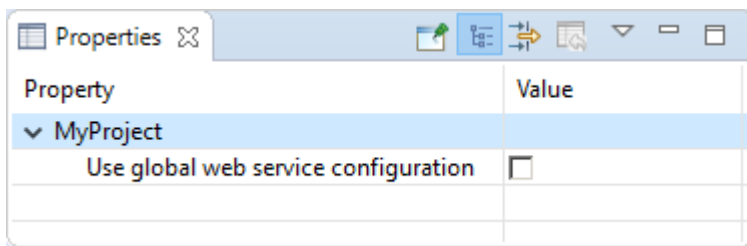
4. Umfang der Suche festlegen:
 - **Entire repository:** Im gesamten Repository suchen.
 - **Enclosing project:** Die Suche im ausgewählte Projekt ausführen.
 - **Selected item:** Suche auf ausgewählte Elemente beschränken.
5. **Search** klicken.

 Sie können auch anhand von Adapter-Parametern suchen, sodass beispielsweise alle Adapter mit einem bestimmten Parameter angezeigt werden.

2.3.7 Globale Webservice-Konfiguration verwenden

Die Bereitstellung von Webservices in Projekten kann über die Projekteigenschaft `Use global web service configuration` gesteuert werden.

Die Eigenschaft lässt sich in der **Properties**-Sicht unten links setzen, sobald das Projekt ausgewählt ist:



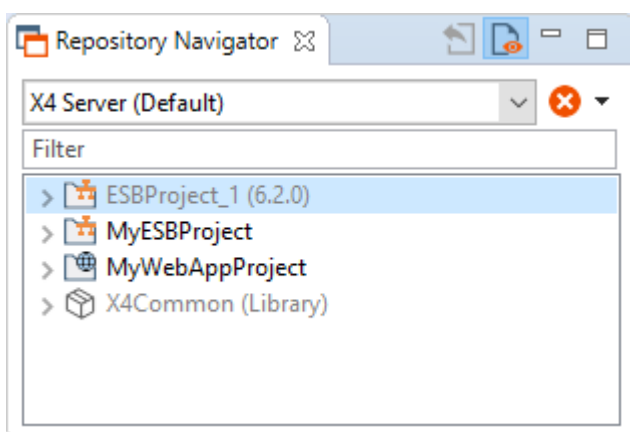
❗ Webservices werden ab Version 6.3.0 der X4 Suite ohne Aufnahme in die `restconfig.xml` bereitgestellt. Die `restconfig.xml` ist damit in Neuinstallationen nicht mehr standardmäßig verfügbar und wird erst beim Definieren eines Webservices unter `<X4>\Server\X4DB\0` angelegt.

✅ Weitere Informationen zur Webservice-Bereitstellung finden Sie im Abschnitt [Webservices via HTTP\(S\) bereitstellen](#).

2.3.8 Veraltete Projekte migrieren

Veraltete Projekte werden im Repository Navigator besonders gekennzeichnet und werden lediglich in einer Read-only-Ansicht angezeigt. Um die Projekte weiter zu bearbeiten, bietet der X4 Designer eine Funktion zur Migration des Projektes in die aktuelle X4 Suite-Version.

Dazu das gewünschte Projekt im Repository Navigator wählen und über das Kontextmenü **Migrate Project** oder das Menü **Project > Migrate Project** das Projekt migrieren.



⚠ Veraltete Projekte lassen sich ab Version 6.0.0 der X4 Suite in die aktuelle X4 Suite-Version migrieren. Projekte mit einer Version älter als 6.0.0, müssen zunächst manuell auf Version 6.0.0 aktualisiert werden.

3 Prozesse anlegen und modellieren

3.1 Prozess anlegen

Mit dem Process Editor können Sie technische Prozessdiagramme grafisch modellieren. Sie können sowohl leere Bausteine aus der Palette als auch Dateien aus dem Repository auf die Zeichnungsfläche ziehen und in der **Properties**-Sicht deren jeweiligen Eigenschaften konfigurieren, um einen ausführbaren Prozess zu erstellen.

1. Innerhalb des Ordners **Processes** z.B. über das Kontextmenü **New > Process** einen neuen Prozess anlegen.

The screenshot shows the 'Process Creation Wizard' dialog box. The title bar says 'Process Creation Wizard'. The main heading is 'Process File Name' with the instruction 'Select template from the list and enter a file name.' There are two radio buttons: 'Blank' (selected) and 'Start with a blank process'. Below these is a text field for 'File name:' containing 'process_1.wrf'. At the bottom are buttons for '?', '< Back', 'Next >', 'Finish' (highlighted), and 'Cancel'.

2. Leere Prozessvorlage oder eine vorhandene Prozessvorlage wählen.
3. In **File name** den Prozessnamen eingeben.
❗ Leerzeichen und Umlaute sind hierbei nicht erlaubt!
4. **Next** klicken, um weitere Prozessinformationen einzugeben.


The screenshot shows the 'Process Creation Wizard' dialog box at the 'Process Information' step. The title bar says 'Process Creation Wizard'. The main heading is 'Process Information' with the instruction 'Enter a process title and description.' There is a text field for 'Process title:' containing 'process_1' and a larger text area for 'Process description:'. At the bottom are buttons for '?', '< Back', 'Next >', 'Finish' (highlighted), and 'Cancel'.

5. In **Process title** einen Prozesstitel eingeben.
Der Prozesstitel wird im Prozessdiagramm angezeigt und kann in der *Properties*-Sicht über die Eigenschaft `Title` verändert und bei Bedarf über Eigenschaft `Show Title` ein- und ausgeblendet werden.

6. Unter **Process description** ggf. eine Prozessbeschreibung hinterlegen.
7. **Finish** klicken, um den Prozess anzulegen.
Ein leeres Prozessdiagramm wird unter der Dateinamensendung `.wrf` im Repository gespeichert und im Process Editor geöffnet.
8. Prozess wie gewünscht modellieren:
 - Übergang herstellen, um Start und Stop zu verbinden.
 - Mit dem Mauszeiger über den Start-Baustein fahren, um einen kleinen Pfeil anzuzeigen.
 - Pfeil anklicken und zum Stop-Baustein ziehen, um eine Übergangslinie zu erstellen.
 - Symbole aus der **Palette** oder dem **Repository Navigator** per Drag&Drop auf die Verbindungslinie ziehen.

i Tipp



Über das Menü **Project > Validate Project** lässt sich der Prozess über einen integrierten Prozess-Validator auf Gültigkeit prüfen.

Wenn Sie Dateien aus dem Repository in Ihren Prozess einbinden und diese in der **Properties**-Sicht entsprechend konfigurieren, können Sie den modellierten Prozess über  ausführen.

3.2 Eigenschaften von Prozessen (.wrf)




- i** Technische Prozesse (`.wrf`) werden über den Process Editor grafisch modelliert und können auf dem X4 Server ausgeführt werden, wenn sämtliche Bausteine entsprechend parametrisiert und korrekt mit Repository-Dateien verknüpft wurden.



- i** Die Eigenschaften jedes Prozesses lassen sich in der **Properties**-Sicht des Process Editors bearbeiten, wenn im Process Editor ein freier Bereich der Zeichnungsfläche angeklickt wird. Standardmäßig werden die gebräuchlichsten Eigenschaften angezeigt, über die Symbolleiste der Properties-Sicht lassen sich jedoch zusätzliche Einstellungen vornehmen:

-  *Show Categories*: Eigenschaften nach Kategorien geordnet anzeigen
-  *Show Advanced Properties*: Erweiterte Eigenschaften anzeigen

Eigenschaften

Documentation	
Title	Freitextfeld für den Prozesstitel
Show Title	Prozesstitel durch Aktivieren bzw. Deaktivieren des Auswahlfeldes ein- und ausblenden
Description	Prozessbeschreibung; Kann bei Bedarf geändert werden

Version	Projektversion  Wenn keine Version gesetzt ist, wird der Zeitstempel als Version genommen.
Project Name	Projektname; Kann bei Bedarf geändert werden
Created	Erstelldatum des Prozesses
Created by	Name des Prozess-Erstellers
Last Update	Datum der letzten Änderung
Last Update by	Name des letzten Bearbeiters
Metadata	
Manage Metadata	Metadaten verwalten, z. B. hinzufügen, löschen oder bearbeiten; Dazu die Eigenschaft markieren und auf  klicken, siehe Prozess-Metadaten hinzufügen
Technical	
Schema Declaration	Schema für Input und Output des Prozesses
Location	Vollständiger relativer Pfad des Prozesses innerhalb des X4 Repositorys; Wird automatisch gesetzt.
Can Stop	Geschäftsprozess darf während der Ausführung angehalten werden  Diese Eigenschaft wird nicht an Subprozesse vererbt. Mögliche Werte: <ul style="list-style-type: none"> • <i>True</i>: für normale Prozesse (Standard) • <i>False</i>: für kritische Prozesse, die zu Ende laufen müssen
Stop on error	Im Fehlerfall die Prozess-Ausführung anhalten Mögliche Werte: <ul style="list-style-type: none"> • <i>True</i>: Geschäftsprozess-Ausführung abbrechen, wenn ein Fehler auftritt (Standard) • <i>False</i>: Fehlerhaften Prozess weiter ausführen (sofern möglich)
In service	Prozess aktiv oder inaktiv setzen. Werden inaktive Prozesse direkt ausgeführt, so erhalten sie die Prozess-ID -1. Subprozesse, also Prozesse, die in anderen Prozessen aufgerufen werden, können unabhängig von dieser Eigenschaft immer ausgeführt werden. Mögliche Werte: <ul style="list-style-type: none"> • <i>True</i>: Prozess kann ausgeführt werden (Standard) • <i>False</i>: Prozess kann nicht ausgeführt werden (nur als Subprozess)

Schedule	<p>Schedule-Werkzeug, das zeitgesteuertes Ausführen von Prozessen auf dem X4 Server ohne X4 Designer ermöglicht; Um einen Zeitplan einzurichten, die Eigenschaft markieren und auf  klicken.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Disabled</i>: Kein Zeitplan wurde festgelegt (Standard) • <i>Enabled</i>: Ein Zeitplan für die Ausführung des Prozesses wurde hinterlegt
Instance limit	<p>Maximale Anzahl von Instanzen dieses Prozesses, die parallel ausgeführt werden dürfen</p> <div style="border: 1px solid #fde725; padding: 10px; margin: 10px 0;"> <p> Beachten Sie!</p> <p>Instance limit wird ignoriert, wenn der Prozesse als Subprozess aufgerufen wird.</p> </div> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 0: Keine Begrenzung (Standard)
Owner	Freitextfeld zur Benennung eines Prozess-Verantwortlichen

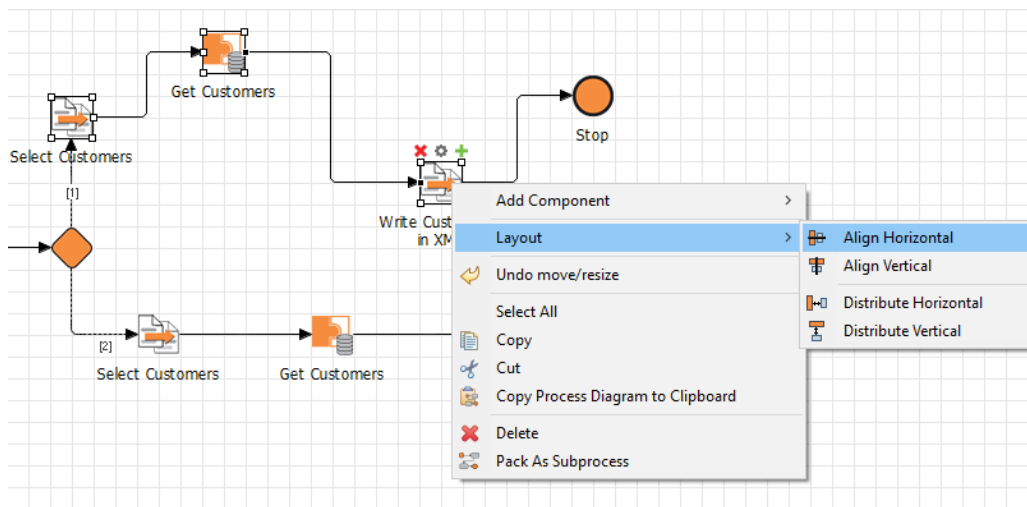
3.3 Prozessbausteine ausrichten

Prozessbausteine von technischen Prozessen und Geschäftsprozessen können über den Kontextmenü-Eintrag **Layout** auf einfache Weise aneinander ausgerichtet werden. Damit schaffen Sie schnell und einfach eine saubere, strukturierte Prozessübersicht.

Dazu die gewünschten Prozesskomponenten auswählen und über Rechtsklick das Kontextmenü aufrufen. Anschließend **Layout** und die gewünschte Option wählen.

Folgende Optionen stehen für das Ausrichten der Prozesskomponenten zur Verfügung

- **Align Horizontal**: Komponenten anhand des Mittelwertes horizontal ausrichten
- **Align Vertical**: Komponenten anhand des Mittelwertes vertikal ausrichten
- **Distribute Horizontal**: Komponenten gleichmäßig horizontal verteilen
- **Distribute Vertical**: Komponenten gleichmäßig vertikal verteilen



i Für die Optionen **Align Horizontal** und **Align Vertical** müssen mindestens zwei Prozesskomponenten ausgewählt sein, für die Optionen **Distribute Horizontal** und **Distribute Vertical** mindestens drei Komponenten.

3.4 Prozesse validieren

Sowohl Geschäftsprozesse (.bpm) als auch technische Prozesse (.wrf) können über einen integrierten Prozess-Validator auf Gültigkeit geprüft werden. Dabei kann die Validierung auf Basis von standardmäßig hinterlegten oder benutzerdefinierten Validierungsregeln durchgeführt werden.

Die Prozess-Validierung kann über folgende Menüs geöffnet werden:

- Kontextmenü > **Validate** bei Rechtsklick auf Prozesse im Repository
- Menü **Project > Validate Project**

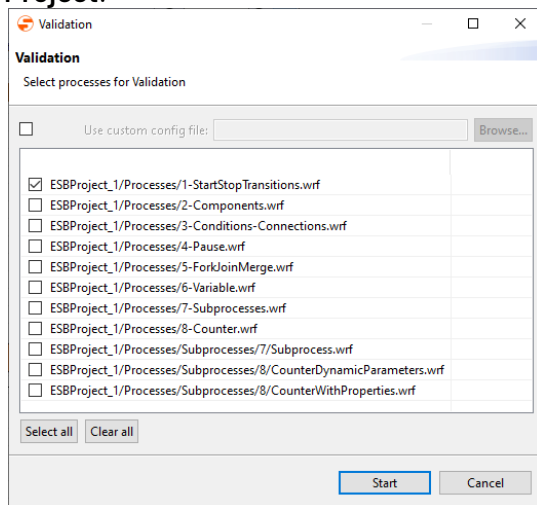
Bei der Prozess-Validierung werden drei Schweregrade unterschieden:

- **Error**: Fehler, welche die Ausführung des Prozesses verhindern
- **Warning**: Fehler, die eine fehlerhafte Ausführung des Prozesses verursachen können
- **Information**: Informationen zum Zustand eines Prozesses

3.4.1 Prozess-Validierung durchführen

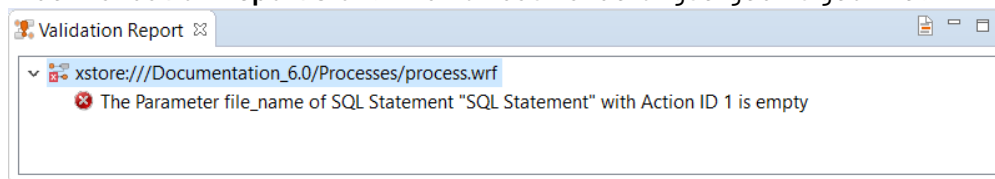
Um einen Prozess auf Gültigkeit zu prüfen, gehen Sie wie folgt vor:

1. Öffnen Sie den Dialog für die Prozess-Validierung, z.B. über das Menü **Project > Validate Project**.



2. Die zu validierenden Prozesse auswählen.
3. Über **Use custom config file** und **Browse**, ggf. eine Konfiguration für eigene Validierungsregeln hochladen, siehe [Benutzerdefinierte Validierungsregeln](#).
4. Mit **Start** die Validierung beginnen.

In der **Validation Report** Sicht wird nun das Validierungsergebnis geöffnet.



- Mit Doppelklick auf den Prozesspfad wird der entsprechende Prozess im X4 Designer geöffnet.
- Mit Doppelklick auf ein Validierungsergebnis werden die Eigenschaften der entsprechende Prozesskomponente in der *Properties-Sicht* geöffnet.

3.4.2 Standardmäßig hinterlegte Validierungsregeln

Folgende Validierungsregeln sind bereits standardmäßig hinterlegt:

Validierungsregel	Beschreibung	Schwere
Action Parameter Validation	Sind alle Aktionen korrekt parametrierung?	Error
Condition Validation	Sind alle Zweige eines Condition-Baust eine gefüllt?	Warning
Connection Validation	Sind alle Pfade geschlossen?	Error

3.4.3 Benutzerdefinierte Validierungsregeln

Es können auch eigene Validierungsregeln definiert werden. Diese müssen als XSLT-Mapping mit einer vorgegebenen Struktur vorliegen.

Beispiel: Mapping-Beispiel für benutzerdefinierte Validierungsregeln

```

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="2.0">
  <xsl:output media-type="text/xml" method="xml"></xsl:output>
  <xsl:template match="/">
    <xsl:param name="subprocessCount" select="count(Workflow/Actions/Action[Module/
text() = 'ixworkflow'])"></xsl:param>
    <xsl:param name="componentCount" select="count(Workflow/Actions/Action)"></xsl:pa
ram>
    <xsl:param name="textCount" select="count(Workflow/Graphics/svg/text)"></xsl:para
m>
    <step>
      <xsl:choose>
        <xsl:when test="$componentCount > 100">
          <warn>The number of process components is <xsl:value-of select="$componentC
ount"></xsl:value-of>. Processes with more than 100 components may probably be
unclear and error-prone. Please check, if you can outsource parts of the process as
subprocesses.</warn>
          </xsl:when>
          <xsl:otherwise>
            <info>The number of process components is <xsl:value-of select="$componentC
ount"></xsl:value-of>
            </info>
          </xsl:otherwise>
        </xsl:choose>
      <xsl:choose>
        <xsl:when test="$textCount > 0">
          <xsl:if test="$componentCount div $textCount > 5">
            <warn>The ratio of process components to text components is greater than
5:1. It is <xsl:value-of select="$componentCount div $textCount"></xsl:value-of>:1.</
warn>
            </xsl:if>
          </xsl:when>
          <xsl:otherwise>
            <info>The process doesn't contain any text components.</info>
          </xsl:otherwise>
        </xsl:choose>
      <xsl:choose>
        <xsl:when test="$subprocessCount > 0">
          <xsl:if test="( $componentCount - $subprocessCount ) div $subprocessCount
> 10">
            <warn>The ratio of process components to subprocesses is greater than
10:1. It is <xsl:value-of select="( $componentCount - $subprocessCount ) div
$subprocessCount"></xsl:value-of>:1</warn>
            </xsl:if>
          </xsl:when>
          <xsl:otherwise>
            <info>The process doesn't contain any subprocesses.</info>
          </xsl:otherwise>
        </xsl:choose>
      </step>
    </xsl:template>
  </xsl:stylesheet>

```

Für die Prozess-Validierung muss eine XML-Datei mit folgendem Aufbau übergeben werden. Das zu verwendende Mapping wird hier anhand der xstore-URL referenziert:

```
<validationConfig>
  <default active="true"/>
  <custom_xslt active="true" name="processMetrics">
    xstore://Projekt/Unterverzeichnis/customValidationMetrics.xsl
  </custom_xslt>
</validationConfig>
```

3.5 Prozessbausteine parametrieren

Erfahren Sie hier, wie Bausteine von technischen Prozessen parametriert bzw. konfiguriert werden, wie innerhalb eines technischen Prozesses Platzhalter verwendet werden und wie sich Parameter dynamisch setzen lassen.

3.5.1 Parametrierungs-Möglichkeiten

Hier erhalten Sie einen Überblick über die Möglichkeiten, je nach Anwendungszweck Prozessbausteine zu parametrieren.

Repository-Dateien konfigurieren	Prozessbaustein-Dateien aus dem Repository können per Doppelklick im jeweiligen Editor geöffnet werden (z. B. Adapter werden im Adapter-Editor geöffnet, XML-Dokumente im XML-Editor). Änderungen werden in der entsprechenden Datei im Repository gespeichert.
Eigenschaften bearbeiten	Markierte Prozessbausteine eines geöffneten Prozesses können Sie in der Properties -Sicht anpassen (z. B. die Eigenschaft <code>Operation</code> setzen). Diese Bausteinkonfiguration wird im jeweiligen Prozess gespeichert.
Benutzerdefinierte Parameter mit Platzhaltern verwenden	<p>Für manche Prozessbausteine lassen sich benutzerdefinierte Parameter (Custom Parameters) definieren, siehe Benutzerdefinierte Parameter verwenden. Diese Konfiguration wird im jeweiligen Prozess gespeichert.</p> <p>Als benutzerdefinierte Parameterwerte können Sie eine Reihe von vordefinierten Platzhaltern verwenden, die zur Prozesslaufzeit gesetzt werden. Platzhalter sind z. B. <code>%PID%</code> (aktuelle Prozessinstanz-ID) oder <code>%NOW%</code> (aktuelles Datum und Uhrzeit), siehe System-Platzhalter.</p>

Dynamische Parameter erzeugen	<p>Innerhalb eines Prozesses können Sie in Abhängigkeit von verarbeiteten Daten mit dynamischen Parametern verschiedene Prozessbausteine ansteuern. Dynamische Parameter werden in XSL-Mappings in Form von Processing Instructions abgelegt und vom darauf folgenden Prozessbaustein (z. B. einem Adapter) interpretiert, siehe Parameter dynamisch setzen.</p> <p>Die Konfiguration wird nur während der Prozesslaufzeit dynamisch verändert.</p>
--------------------------------------	---



Beachten Sie folgende Priorisierung bei der Parametrierung:



- Benutzerdefinierte Parameter (Custom Parameters) überschreiben die Konfiguration von Prozessbausteinen im Repository (beispielsweise Funktions-Adapter) während der Prozesslaufzeit.
- Dynamische Parameter überschreiben gesetzte benutzerdefinierte Parameter und die Konfiguration von Prozessbausteinen im Repository (beispielsweise Funktions-Adapter) zur Prozesslaufzeit.

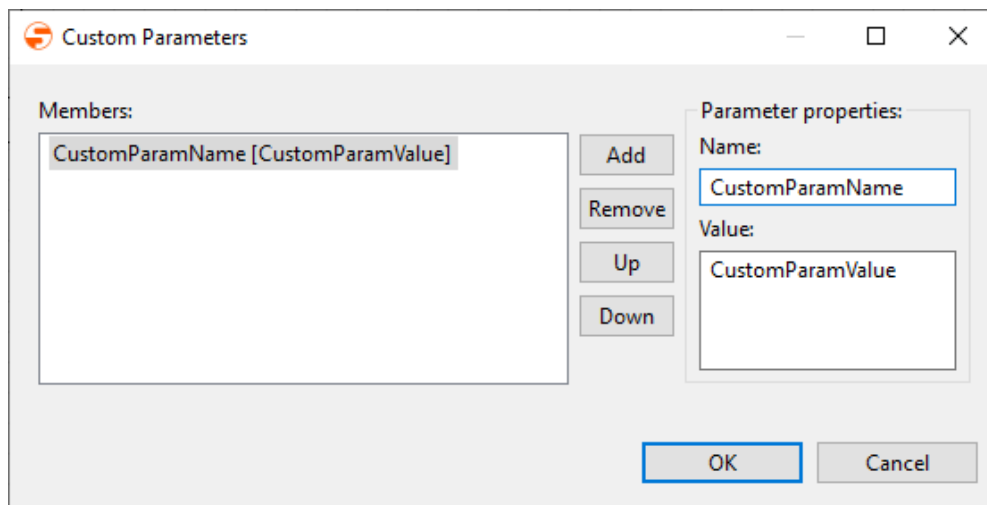
3.5.2 Benutzerdefinierte Parameter verwenden

Für bestimmte Prozessbausteine eines Prozesses (für XSL-Mappings, PDF-Creator-Bausteine und Adapter) lassen sich benutzerdefinierte Parameter (Custom Parameters) definieren.

Als benutzerdefinierte Parameterwerte können Sie eine Reihe von vordefinierten Platzhalter verwenden, die zur Laufzeit des Prozesses gesetzt werden. Platzhalter sind z. B. %PID% (aktuelle Prozessinstanz-ID) oder %NOW% (aktuelles Datum und Uhrzeit), siehe [System-Platzhalter](#)

Um benutzerdefinierte Parameter zu definieren, gehen Sie wie folgt vor:

1. Fenster **Custom Parameters** aufrufen:
 - Prozessbaustein im Prozess markieren.
 - In der **Properties**-Sicht auf Custom Parameters klicken, um  anzuzeigen.
 -  klicken, um das Fenster **Custom Parameters** aufzurufen.



2. In **Name** den Namen des benutzerdefinierten Parameters eingeben.
 - In XSL-Mappings und Reports können Sie den hier gesetzten Namen als globalen Parameter ansprechen und den zugehörigen Wert übergeben, z. B. einen benutzerdefinierten Parameter PDFPath mit `<xsl:param name="PDFPath"/>`.
 - Bei Adaptern und Datenbankverbindungen: Die exakte (technische) Bezeichnung des zu überschreibenden Parameters verwenden
3. In **Value** den zu überschreibenden Parameterwert eingeben.
 Hier können Sie Platzhalter einfügen, die beim Ausführen des Prozesses durch die jeweils aktuellen Werte ersetzt werden, z. B. %PID% durch die aktuelle Prozessinstanz-ID.
 Für eine vollständige Liste der verfügbaren Platzhalter, siehe [System-Platzhalter](#).
4. **OK** klicken, um die benutzerdefinierten Parameter im Prozess zu speichern.

*Funktionen in Fenster **Custom Parameters**:*

- **Add**: Weitere benutzerdefinierte Parameter hinzufügen
- **Remove**: Ausgewählten Parameter aus der Liste löschen
- **Up / Down**: Reihenfolge der Parameter wechseln

3.5.3 Parameter dynamisch setzen

Innerhalb eines technischen Prozesses können Sie in Abhängigkeit von verarbeiteten Daten mit sogenannten dynamischen Parametern Prozessbausteine ansteuern und ggf. bereits gesetzte Parameter zur Prozesslaufzeit überschreiben.

Dynamische Parameter werden in XSL-Mappings als Processing Instruction erzeugt und vom darauf folgenden Prozessbaustein (beispielsweise einem Adapter) verarbeitet. Hierbei haben Sie mehrere Möglichkeiten: Sie können ein einzelnes XSL-Daten-Mapping einsetzen, das XML-Daten transformiert und gleichzeitig dynamisch Processing Instruction-Definitionen enthält. Sie können aber auch separates XSL-Mapping erstellen, das nur die Processing Instructions generiert.


3.5.3.1 Parameter via XSL-Mapping setzen

1. XSL-Mapping im technischen Prozess vor dem Adapter oder der Datenbank-Verbindung positionieren.
2. Processing Instruction im XSL-Mapping nach folgendem Muster definieren:

```
<xsl:processing-instruction name="DynamicParameter">ParameterName="ParameterValue "</xsl:processing-instruction>
```
3. `ParameterName` durch den gewünschten Parameternamen ersetzen.
4. `ParameterValue` durch den gewünschten Wert ersetzen, meist in Form eines XSL-Befehls.
5. Wenn das XSL-Mapping ausschließlich dynamische Parameter erzeugen soll: Zusätzlich mit

```
<xsl:processing-instruction name="NoResult"/>
```

 festlegen, dass das XSL-Mapping kein Ergebnis-Dokument erzeugt.

 Das XSL-Mapping muss in jedem Fall ein Wurzel-Element generieren, damit das XSL-Mapping ausgeführt werden kann.

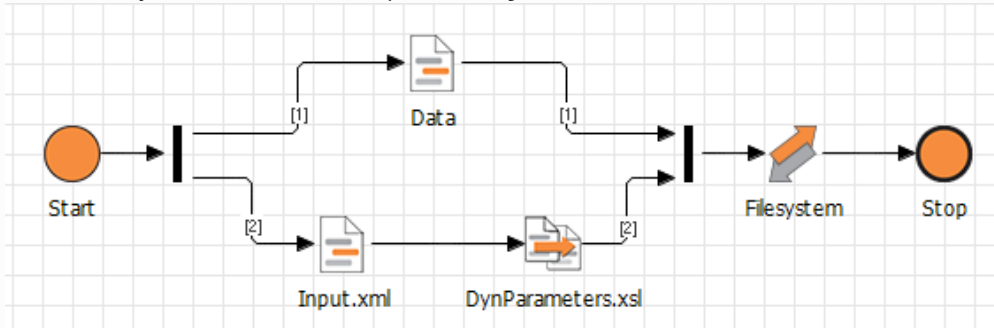
3.5.3.2 Dynamisch parametrieren ohne Datentransformation

Wenn ein Input-Dokument direkt an einen dynamisch parametrierten Adapter übergeben werden soll, ist es erforderlich, eine Gabelung (mithilfe des Fork-Bausteins) in das Diagramm des technischen Prozesses einzufügen:

1. In einen Ausführzweig ein XSL-Mapping setzen.
2. In einen zweiten Ausführzweig den zu übergebenden Prozessbaustein setzen.
3. Den Adapter direkt hinter den Join-Baustein setzen.

❗ Beispiel

Ein aus einem XML-Dokument ausgelesener Dateiname soll als dynamischer Parameter an einen Filesystem Transfer Adapter übergeben werden:



Im Beispiel besteht das XSL-Mapping aus folgendem Quelltext:

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/" version="1.1">
  <xsl:output method="xml" media-type="text/xml"/>
  <xsl:template match="/">
    <xsl:processing-instruction name="DynamicParameter">documentname="
      <xsl:value-of select="data"/>
    </xsl:processing-instruction>
    <xsl:processing-instruction name="NoResult"/>
    <root/>
  </xsl:template>
</xsl:stylesheet>
```

3.5.3.3 Dynamische Parameter mit DynamicParameter

Mit XSL-Mappings können Sie die Konfiguration von Adaptern und Datenbank-Verbindungen zur Laufzeit beeinflussen. Innerhalb des XSL-Mappings wird eine Logik implementiert, die in Abhängigkeit von den verarbeiteten Daten benutzerdefinierte Parameterwerte ausgibt. Diese benutzerdefinierten Parameterwerte können von einem nachfolgenden Adapter-, Transfer-Adapter- oder Datenbank-Baustein verarbeitet werden und überschreiben bereits gesetzte Werte.

Parameterwerte erzeugen

Um dynamische Werte innerhalb von technischen Prozessen zu setzen, können Sie mithilfe von XSL-Mappings Processing Instructions erzeugen, die an den nachfolgenden Prozess-Schritt übergeben werden und entsprechend interpretiert werden können.

```
<xsl:processing-instruction name="DynamicParameter">
  parameterName="parameterValue"</xsl:processing-instruction>
```

Die Ausgabe des XSL-Mappings:

```
<?DynamicParameter parameterName="parameterValue" ?>
```

Parameter und Parameterwerte

Im XSL-Mapping können Sie beliebig viele unterschiedliche dynamische Parameter mit jeweils einem zugehörigen Wert definieren:

- `parameterName`: Zu überschreibender Parametername
- `parameterValue`: Zu überschreibender Parameterwert; immer innerhalb von Anführungszeichen

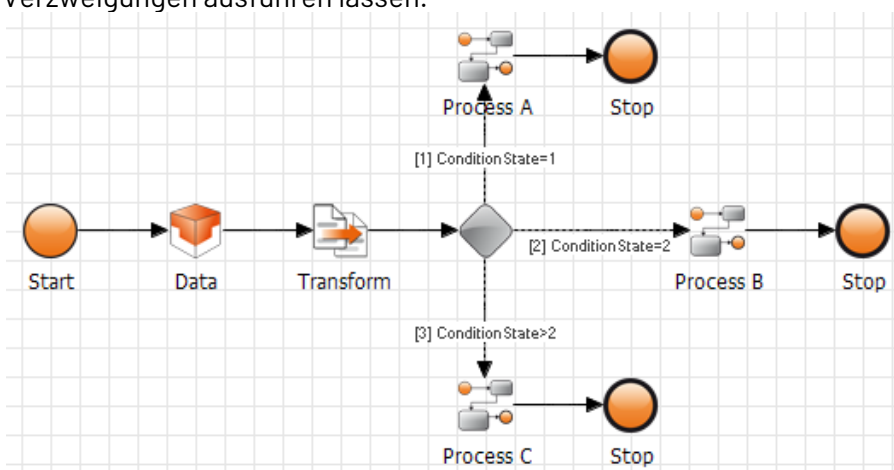
i Beachten Sie, dass das XSL-Mapping ein Wurzel-Element generieren muss, damit das XSL-Mapping ausgeführt und ein gültiges XML-Dokument erzeugt werden kann.

Beispiel

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">
  <xsl:output encoding="UTF-8" method="xml" indent="yes" version="1.0"/>
  <xsl:template match="/">
    <xsl:processing-instruction
      name="DynamicParameter">key="i"</xsl:processing-instruction>
    <root/>
  </xsl:template>
</xsl:stylesheet>
```

3.5.4 Status mit ConditionState dynamisch setzen

Mit einem XSL-Mapping können Sie abhängig von verarbeiteten Daten zusätzlich Ausführ-Statuswerte als Processing Instruction erzeugen, die innerhalb des technischen Prozesses von einem Condition-Baustein ausgewertet werden. Je nach Status können Sie so entsprechend definierte Verzweigungen ausführen lassen.



1. Leeres XSL-Mapping in einem technischen Prozess vor einem Condition-Baustein positionieren.
2. Processing Instruction(s) im XSL-Mapping nach folgendem Muster definieren:

```
<xsl:processing-instruction name="ConditionState">value</xsl:processing-instruction>
```

3. *value* durch einen Wert z.B. aus dem verarbeiteten XML-Dokument ersetzen.

Beispiel: `<xsl:value-of select="data"/>`

i Zulässige Statuswerte *ganze Zahlen (Integer)* sowie negative Zahlen (z. B. *-1*) für einen Fehlerstatus.

Nicht zulässig sind folgende Status-Nummern, da diese systemintern reserviert sind:

- *-999* (Status-ID für Stop-Baustein)
- *999* (Status-ID für noch nicht ausgeführte Prozessbausteine)
- *997* (bewirkt die sofortige Beendigung des gesamten technischen Prozesses)
- *998* (wird vom Save Point-Mechanismus verwendet und bewirkt die sofortige Beendigung des gesamten Prozesses)

Empfehlung: Verwenden Sie für benutzerdefinierte Statuswerte Zahlen größer als 1000, um mögliche Überschneidungen mit anderen Statuswerten zu vermeiden.

4. Sicherstellen, dass das XSL-Mapping ein Wurzel-Element generiert, damit das XSL-Mapping ausgeführt und ein gültiges XML-Dokument erzeugt werden kann.

Das XSL-Mapping gibt nun *Processing Instructions* nach dem Muster `<?ConditionState value ?>` aus.

5. Im nachfolgenden *Condition*-Baustein für jede Verzweigung einen Status definieren, siehe [Bedingungen erstellen und verwenden](#).
6. Wenn Sie im weiteren Verlauf des Prozesses erneut *ConditionStates* erzeugen: die vorherigen *Processing Instructions* zunächst aus dem Input-XML-Dokument z.B. per `<xsl:copy-of>` entfernen, um unerwartetes Verhalten bei der Prozessausführung zu vermeiden.

i Problemlösung: Das XSL-Mapping lässt sich nicht ausführen?

Möglicherweise ist der verarbeitete Wert einer der unzulässigen Statuswerte *-999*, *999*, *997*, *998* oder ist eine Zeichenkette. Oder das vom XSL-Mapping erzeugte XML-Dokument enthält kein Wurzel-Element.

- Verwenden Sie nur zulässige Statuswerte (siehe oben).
- Fügen Sie z. B. ein leeres Element `<root/>` hinter die *ConditionState*-Definitionen ins XSL-Mapping ein.

3.6 Platzhalter verwenden

Wenn technische Prozesse für variable Umgebungsbedingungen gerüstet sein sollen, können Platzhalter die Prozessmodellierung deutlich vereinfachen – vor allem benutzerdefinierte Platzhalter, die es erlauben, beliebige Schlüssel/Wertepaare, z.B. für technische Umgebungsvariablen wie Pfade oder für fachliche Parameter wie Grenzwerte für Prüfungen, systemweit zu definieren. Zur Laufzeit des technischen Prozesses werden diese Platzhalter durch zuvor definierte Werte ersetzt. Damit lassen sich technische Prozesse so modellieren, dass sie in verschiedenen Umgebungsbedingungen

ausgeführt werden können, etwa für den Betrieb in Staging-Systemen. Ohne weitere Anpassung der Prozessmodelle lassen sich technische Prozesse so über externe Parameter steuern.

Benutzerdefinierten Platzhalter werden nach fachlichen oder technischen Kriterien gruppiert, um gezielt eingesetzt werden zu können. So können beispielsweise nur die tatsächlich für einen Prozess relevanten Platzhalter verwendet werden.


3.6.1 System-Platzhalter

Die X4 Suite stellt eine Reihe von Platzhaltern bereit, die Sie in *Eigenschaften und Parametern* von Prozessbausteinen einsetzen können. Während der Ausführung des Prozesses werden die Platzhalter durch die jeweiligen Werte ersetzt und von der Prozess-Engine verarbeitet.

Folgende Platzhalter werden in der X4 Suite bereitgestellt:

<code>%ACTION_ID%</code>	Gibt die laufende Nummer des aktuell ausgeführten Prozess-Schrittes aus. <i>Beispiel: 3</i>
<code>%CALLING_PROCESS%</code>	Gibt den relativen Repository-Pfad und den Dateinamen des Prozesses aus, der den aktuellen Prozess aufgerufen hat. Dieser Platzhalter wird nur umgewandelt, wenn der aktuelle Prozess von einem anderen Prozess direkt als Subprozess aufgerufen wurde. <i>Beispiel: Project/Folder/ProcessCaller.wrf</i>
<code>%CALLING_PROCESS_NAME%</code>	Gibt den Dateinamen des Prozesses ohne Dateinamensendung aus, der den aktuellen Prozess direkt als Subprozess aufgerufen hat. Dieser Platzhalter wird nur umgewandelt, wenn der aktuelle Prozess von einem anderen Prozess als Subprozess aufgerufen wurde. <i>Beispiel: ProcessCaller</i>
<code>%CALLING_PROCESS_PATH%</code>	Gibt den relativen Repository-Pfad des Prozesses aus, der den aktuellen Prozess direkt als Subprozess aufgerufen hat. <i>Beispiel: Project/Folder/</i>
<code>%CURRENT_PROCESS%</code>	Gibt den relativen Repository-Pfad und den Dateinamen des aktuellen Prozesses aus. <i>Beispiel: Project/Folder/Process.wrf</i>
<code>%CURRENT_PROCESS_NAME%</code>	Gibt den Dateinamen des aktuellen Prozesses ohne Dateinamensendung aus. <i>Beispiel: Process</i>

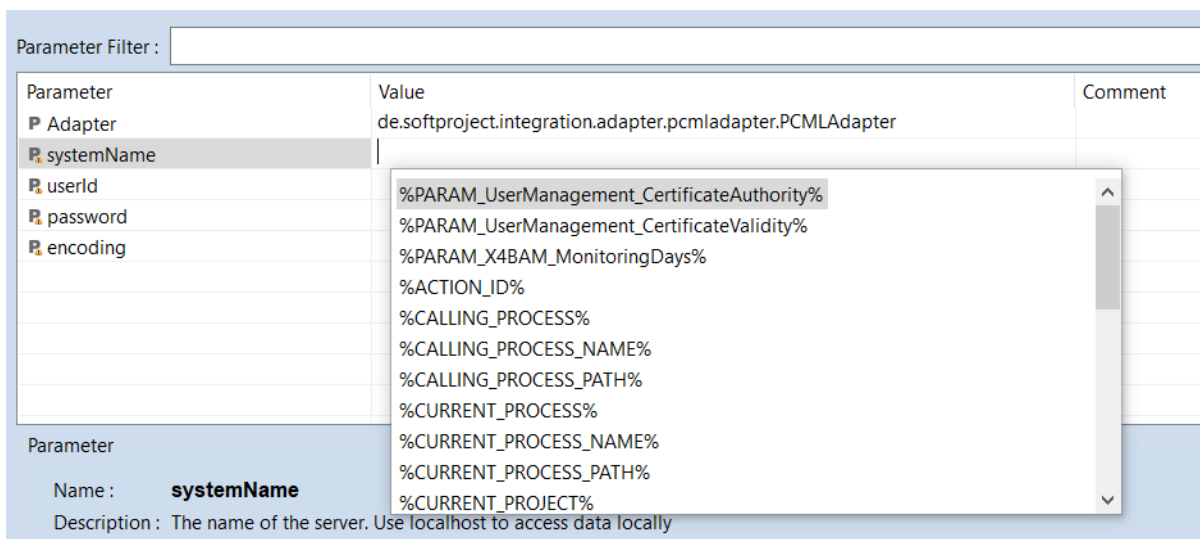
<code>%CURRENT_PROCESS_PATH%</code>	<p>Gibt den relativen Repository-Pfad des aktuellen Prozesses aus.</p> <p>Dieser Platzhalter wird nur umgewandelt, wenn der aktuelle Prozess von einem anderen Prozess als Subprozess aufgerufen wurde.</p> <p><i>Beispiel: Project/Folder/</i></p>
<code>%CURRENT_PROJECT%</code>	<p>Gibt den Namen des aktuellen Repository-Projekts aus.</p> <p><i>Beispiel: Project</i></p>
<code>%HOSTNAME%</code>	<p>Gibt den Host-Namen des <i>X4 Servers</i> aus.</p> <p><i>Beispiel: sp-ws-40</i></p>
<code>%HTTP_PORT%</code>	<p>Gibt den HTTP-Port aus, der in der Konfigurationsdatei <code>x4config.xml</code> im Element <code><webContainerURL/></code> festgelegt wurde.</p> <p><i>Beispiel: 8080</i></p>
<code>%INSTANCE%</code>	<p>Gibt den Namen der aktuellen <i>X4 Server</i>-Instanz aus, der in der Konfigurationsdatei <code>x4config.xml</code> im Element <code><instanceName></code> festgelegt wurde.</p> <p><i>Beispiel: X4</i></p>
<code>%IPADDRESS%</code>	<p>Gibt die IP-Adresse des <i>X4 Servers</i> aus.</p> <p><i>Beispiel: 192.168.0.1</i></p>
<code>%NOW%</code>	<p>Gibt das aktuelle Datum und die Uhrzeit im ISO-Format <code>yyyy-MM-ddThh:mm:ss</code> aus.</p> <p><i>Beispiel: 2010-02-13T15:52:51</i></p>
<code>%NOW_PLAIN%</code>	<p>Gibt das aktuelle Datum und die Uhrzeit im Format <code>yyyyMMddhhmmss</code> aus.</p> <p><i>Beispiel: 20100213155251</i></p>
<code>%PID%</code>	<p>Gibt die ID der aktuellen Prozess-Instanz als ganze Zahl aus, die innerhalb einer <i>X4 Server</i>-Instanz eindeutig ist. Subprozesse werden mit derselben Prozess-ID ausgeführt, nicht jedoch Prozesse, die über den <i>Process Starter</i> gestartet wurden.</p> <p><i>Beispiel: 1260192816371</i></p> <p>Wenn mehrere Instanzen eines <i>X4 Servers</i> parallel auf einem Applikations-Server ausgeführt werden, empfiehlt sich, statt <code>%PID%</code> den eindeutigen Platzhalter <code>%UID%</code> zu verwenden.</p>

%START_PROCESSES%	<p>Gibt den relativen Repository-Pfad und den Dateinamen des Prozesses aus, der ursprünglich gestartet wurde.</p> <p>Wird der aktuelle Prozess als Subprozess eines anderen Prozesses gestartet, so wird dessen Pfad und Dateiname angezeigt.</p> <p><i>Beispiel: Project/Folder/Process.wrf</i></p>
%START_PROCESSES_NAME%	<p>Gibt den Dateinamen des ursprünglich gestarteten Prozesses ohne Dateinamensendung aus.</p> <p>Wird der aktuelle Prozess als Subprozess gestartet, so wird der Dateiname des ursprünglichen gestarteten Prozesses angezeigt, der den aktuellen Prozess-Kontext erzeugt hat.</p> <p> Beachten Sie, dass bei einem Prozessaufruf mittels <i>Process Starter</i> ein neuer Prozess-Kontext erzeugt wird.</p> <p><i>Beispiel: Process</i></p>
%START_PROCESSES_PATH%	<p>Gibt den relativen Repository-Pfad des Prozesses aus, der ursprünglich gestartet wurde.</p> <p>Wird der aktuelle Prozess als Subprozess eines anderen Prozesses gestartet, so wird dessen Pfad angezeigt.</p> <p><i>Beispiel: Project/Folder/</i></p>
%TIME_MS%	<p>Gibt die Zahl der Millisekunden seit 1. Januar 1970, 0:00 Uhr aus.</p> <p><i>Beispiel: 1260197571880</i></p>
%UID%	<p>Gibt den Namen der <i>X4 Server</i>-Instanz %INSTANCE% und die angehängte aktuelle Prozess-ID (%PID%) aus.</p> <p>Damit lässt sich eine Prozess-Instanz eindeutig identifizieren, wenn auf demselben Applikations-Server mehrere <i>X4 Server</i>-Instanzen parallel ausgeführt werden.</p> <p><i>Beispiel: X41260192816371</i></p>
%USER_COLLECTION%	<p>Gibt den Pfad (absolut oder relativ) zum Repository-Ordner des aktuellen Benutzers in der X4DB aus; die so genannte "<i>User Collection</i>".</p> <p>Dieser Wert wird aus der zentralen Konfigurationsdatei <i>X4config.xml</i> von Element <code><repositoryPath></code> unverändert übernommen, der dort auch relativ definiert sein kann, siehe <i>X4 Server - Repository-Konfiguration</i></p> <p><i>Beispiel: C:\X4\X4DB\1 oder X4DB/1</i></p>

<code>%USER_COLLECTION_ABSOLUTE%</code>	Gibt den absoluten Pfad zum Repository-Ordner des aktuellen Benutzers in der x4DB aus; die so genannte "User Collection" <i>Beispiel: C:\X4\X4DB\1</i>
<code>%USER_ID%</code>	Gibt die ID des aktuellen Benutzers aus, in dessen Repository-Ordner sich der Prozess befindet ("user collection"). Seit X4-Version 4.0 ist die User-ID keine ganze Zahl mehr, sondern eine Zeichenkette. <i>Beispiel:1</i>
<code>%WORKING_DIR_ABSOLUTE%</code>	Gibt den absoluten Pfad des aktuellen Arbeitsverzeichnisses aus, in dem der Applikations-Server gestartet wurde. <i>Beispiel: C:\X4</i>

Verwendung von System-Platzhaltern in Adaptern

Platzhalter können sowohl über *Custom Parameters* in den Adapter-Eigenschaften als auch bei allen Textfeldern im Adapter Editor selbst eingesetzt werden. Um die Eingabe zu erleichtern, wird hierzu über Strg + Leertaste eine Autovervollständigungs-Funktion angeboten. Anschließend kann der gewünschte Platzhalter aus einer Drop-down-Liste ausgewählt werden.



Platzhalter für Webservices

Innerhalb des Webservice Configuration Editors können zusätzliche Platzhalter für die Nutzung von Webservices aktiviert werden. Damit lassen sich Parameter für diese ReSTful-Webservice- oder SOAP-Webservice-Operation verarbeiten und Informationen aus dem HTTP-Request als Platzhalter bereitstellen (als effizientere Alternative zu Input Mode HTTP-MetaXml).

Mögliche Werte:

- Disabled: Keine Platzhalter aus dem HTTP-Request im Kontext des technischen Prozesses bereitstellen (Standard)

- **Enabled:** Platzhalter aus HTTP-Request-Informationen im Kontext des technischen Prozesses bereitstellen

The screenshot shows the configuration window for an HTTP ReST operation in X4 ESB. On the left, a 'Tree' panel shows a project 'Documentation_6.0' with a sub-project 'HTTP ReST' containing a 'GET' operation. The main area is titled 'HTTP ReST Operation' and contains two sections: 'Access Restriction' and 'Operation Configuration'.

Access Restriction:

- Request Method: GET
- Realms: (empty field) with 'Inherit from parent node' checked.
- Guest Allowed: false with 'Inherit from parent node' checked.
- Token Return: NONE with 'Inherit from parent node' checked.
- Login Failure Ret: NONE with 'Inherit from parent node' checked.
- App Permission: (empty field) with 'Inherit from parent node' checked.
- Action Permissio: (empty field) with 'Inherit from parent node' checked.
- WADL Allowed: false with 'Inherit from parent node' checked.

Operation Configuration:

- Process Path: (empty field) with a 'Browse...' button.
- Input Mode: HTTP-MetaXml
- Output Mode: HTTP-MetaXml
- Placeholders: Enabled

Folgende Platzhalter mit Informationen aus dem HTTP-Request stehen bei Option **Enabled** innerhalb des X4-Prozesskontexts zur Verfügung, z.B. innerhalb von XSL-Mappings:

- REQUEST_PARAMETER_<Name>: Gewünschter Parameter (Groß-/Kleinschreibung wird beachtet!)
- REQUEST_HEADER_<NAME>: Gewünschter HTTP-Header (nur Großschreibung!)
- REQUEST_METHOD
- REQUEST_SCHEME
- REQUEST_PATH
- REQUEST_FULL_PATH
- REQUEST_REMOTE_ADDRESS
- REQUEST_REMOTE_PORT
- REQUEST_REMOTE_USER
- REQUEST_LOCAL_ADDRESS
- REQUEST_LOCAL_PORT
- REQUEST_SERVER_NAME
- REQUEST_SERVER_PORT

i Diese Platzhalter können wie alle Standard-Platzhalter verwendet werden, lassen sich jedoch nicht überschreiben und sind nicht in der Verwaltung für Benutzerdefinierte Platzhalter sichtbar.

3.6.2 Benutzerdefinierte Platzhalter verwenden

Benutzerdefinierte Platzhalter in technischen Prozessen verwenden

Um benutzerdefinierte Platzhalter in technischen Prozessen verwenden zu können, müssen die zu verwendenden Platzhalter zunächst in den aktuellen Prozess-Kontext eingelesen werden. Dafür gibt es folgende Möglichkeiten:

- Über den [Custom Placeholder Initializer](#) Adapter mit der Operation Load. Im Adapter-Parameter groups können die gewünschten Platzhaltergruppen durch Komma getrennt genannt werden.
- Benutzerdefinierten Platzhalter über den [Custom Placeholder Storage Manage](#) zur Prozesslaufzeit verwalten.

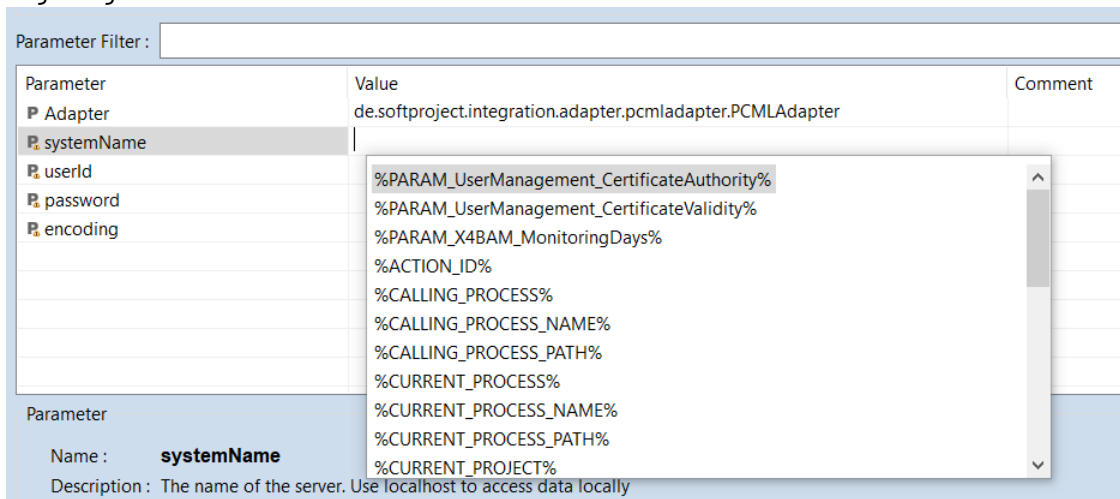
Benutzerdefinierte Platzhalter in Adapter-Parametern verwenden

Analog zu den System-Platzhaltern lassen sich auch benutzerdefinierte Platzhalter als Parameterwerte mit %PARAM_Gruppe_Name% in Adapter-Konfigurationen einsetzen, siehe auch [System-Platzhalter](#).

P MQEnviroment.port %PARAM_X4Environment_MQPort%

Benutzerdefinierte Platzhalter in Transfer-Adapter verwenden

Bei Transfer-Adaptoren wird für die Eigenschaft `File name` über die Tastenkombination Strg+Leertaste eine Vorschlagsliste mit den System- und Benutzerdefinierten Platzhaltern angezeigt.



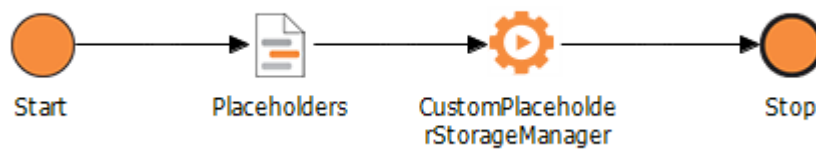
Benutzerdefinierte Platzhalter in XSL-Parametern verwenden

Analog zu den System-Platzhaltern lassen sich auch benutzerdefinierte Platzhalter als Parameter mit %PARAM_Gruppe_Name% in XSL-Stylesheets verwenden.

Beispiele für die Nutzung

Benutzerdefinierte Platzhalter mit dem Custom Placeholder Storage Manager anlegen

Folgendes Prozessbeispiel zeigt, wie benutzerdefinierte Platzhalter über den [Custom Placeholders Storage Manager](#) Adapter angelegt werden können. Der technische Prozess liest dabei ein XML-Dokument mit benutzerdefinierten Platzhaltern ein und stellt diese über den Adapter bereit.



Input-XML-Dokument bereitstellen

Das Input-XML-Dokument muss nach folgendem Schema aufgebaut sein:

Beispiel: Input-XML mit vordefinierten Platzhaltern

```

<Placeholder>
  <Group name="X4Environment">
    <Key name="MQHost" type="String">123.456.789.000</Key>
    <Key name="MQPort" type="Integer">1415</Key>
    <Key name="MQChannel" type="String">Dev</Key>
  </Group>
  <Group name="Thresholds">
    <Key name="MinMessages" type="Integer">0</Key>
    <Key name="MaxMessages" type="Integer">10</Key>
  </Group>
</Placeholder>
  
```

Erläuterung:

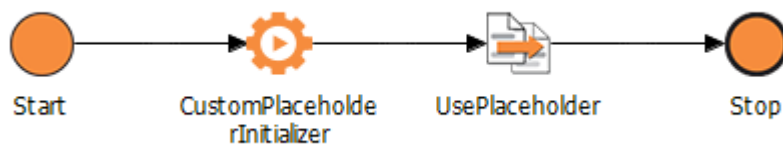
- Das Wurzelement `Placeholder` kann beliebig viele `Group`-Elemente enthalten.
- Ein `Group`-Element stellt jeweils eine Platzhaltergruppe dar und kann beliebig viele `Key`-Elemente enthalten.
- Das Attribut `name` innerhalb eines `Group`-Elements definiert den Namen der Platzhaltergruppe.
- Ein `Key` steht jeweils für ein Platzhalter und enthält folgende Attribute und Inhalte:
 - `name`: Name des Platzhalters
 - `type`: Platzhaltertyp `String`, `Boolean`, `Integer`, `Decimal` oder `Date` (ISO-Date-Format `YYYY-MM-DD`)
 - Wert des Platzhalters entsprechend dem Platzhaltertyp

Technischen Prozess modellieren

1. Neuen technischen Prozess über das Menü **New > Process** anlegen.
2. Input-XML-Datei (siehe Beispiel) in den Prozess ziehen und in den Eigenschaften die Operation *Read* wählen.
3. Den [Custom Placeholder Storage Manager](#) in den Prozess ziehen und die Adapter-Operation *AddKeys* wählen.
4. Prozess speichern und anschließend ausführen.
Die Platzhalter wurden nun angelegt und können sowohl im [Benutzerdefinierte Platzhalter und -gruppen verwalten](#) als auch in der Administration-Oberfläche des [X4 Control Centers](#) eingesehen und bearbeitet werden.

Benutzerdefinierte Platzhalter in XSL-Mappings verwenden

Folgendes Beispiel zeigt die Verwendung von benutzerdefinierten Platzhaltern in Mappings. Vor ihrer Verwendung muss die relevante Gruppe von benutzerdefinierten Platzhaltern in den aktuellen Prozess-Kontext, z.B. über den [Custom Placeholder Initializer](#) geladen werden. Alternativ können die Platzhalter auch über die Prozess-Eigenschaften Placeholder Groups und Max Age geladen werden.



Mapping erstellen

Das XSL-Mapping muss nach folgendem Schema aufgebaut sein. Die zu verwendenden benutzerdefinierten Platzhalter müssen zunächst im Stylesheet unter ihrem voll qualifizierten Namen deklariert werden. Anschließend lassen sie sich wie gewohnt verwenden.

Beispiel: XSL-Mapping

```

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="2.0">
  <xsl:output media-type="text/xml" method="xml"></xsl:output>
  <xsl:param name="PARAM_X4Environment_MQHost"></xsl:param>
  <xsl:param name="PARAM_Thresholds_MinMessages"></xsl:param>
  <xsl:param name="PARAM_Thresholds_MaxMessages"></xsl:param>
  <xsl:template match="/">
    <Result>
      <X4Environment>
        <MQHost><xsl:value-of select="$PARAM_X4Environment_MQHost"></xsl:value-
of></MQHost>
      </X4Environment>
      <Thresholds>
        <MinMessages><xsl:value-of select="$PARAM_Thresholds_MinMessages"></xsl:v
alue-of></MinMessages>
        <MaxMessages><xsl:value-of select="$PARAM_Thresholds_MaxMessages"></xsl:v
alue-of></MaxMessages>
      </Thresholds>
    </Result>
  </xsl:template>
</xsl:stylesheet>
  
```

Technischen Prozess modellieren

1. Neuen technischen Prozess über das Menü **New > Process** anlegen.

2. Den **Custom Placeholder Initializer** in den Prozess ziehen und die Adapter-Operation *Load* wählen.
3. Das XML-Mapping (siehe Beispiel) in den Prozess ziehen.
4. Prozess speichern und anschließend ausführen.
Die Platzhalter wurden nun geladen und können verwendet werden.

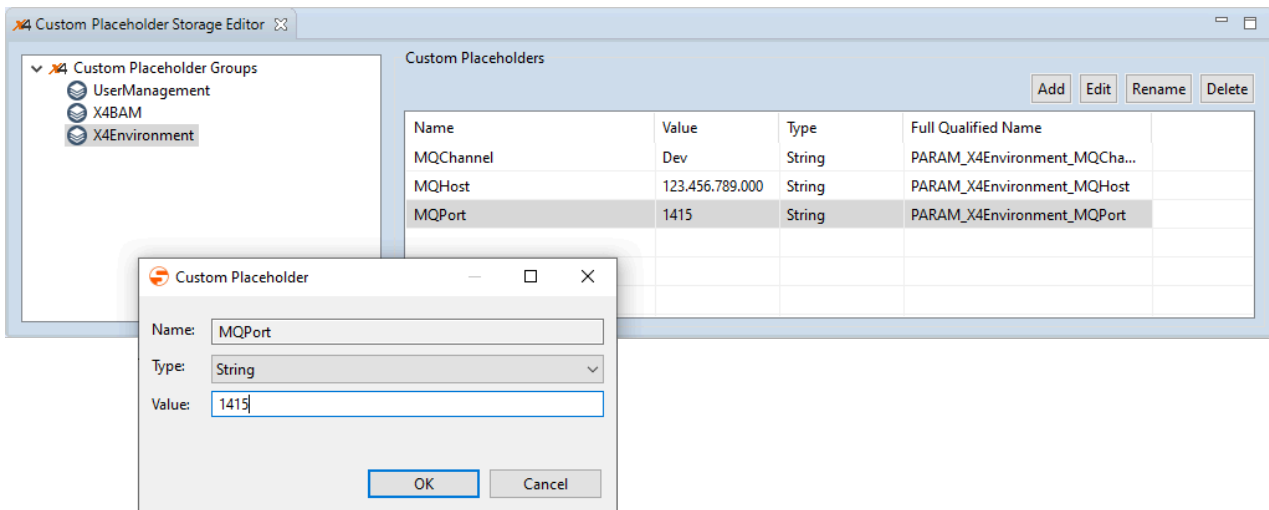
3.6.3 Benutzerdefinierte Platzhalter und -gruppen verwalten

Es gibt verschiedene Möglichkeiten, um benutzerdefinierte Platzhalter anzulegen und zu verwalten:

3.6.3.1 Platzhalter über den Custom Placeholder Storage Editor verwalten

Benutzerdefinierte Platzhalter lassen sich manuell über den Custom Placeholder Storage Editor definieren. Dabei können beliebige Platzhaltergruppen mit benutzerdefinierten Platzhaltern erstellt und direkt im X4 Designer verwaltet werden. Diese lassen sich anschließend wie System-Platzhalter in den Werkzeugen des X4 Designers (Adapter Editor, Mapping Editor, Process Editor) nutzen, siehe [Benutzerdefinierte Platzhalter verwenden](#).

i Um den Custom Placeholder Storage Editor aufzurufen, wählen Sie das Menü **Tools > Manage Custom Placeholders**.



✓ Über Rechtsklick auf einen benutzerdefinierten Platzhalter lässt sich dessen Kontextmenü öffnen. Mit **Copy Full Qualified Name to Clipboard** kann anschließend dessen vollqualifizierter Name kopiert und an anderer Stelle eingefügt werden.

3.6.3.1.1 Platzhaltergruppen im Custom Placeholder Storage Editor

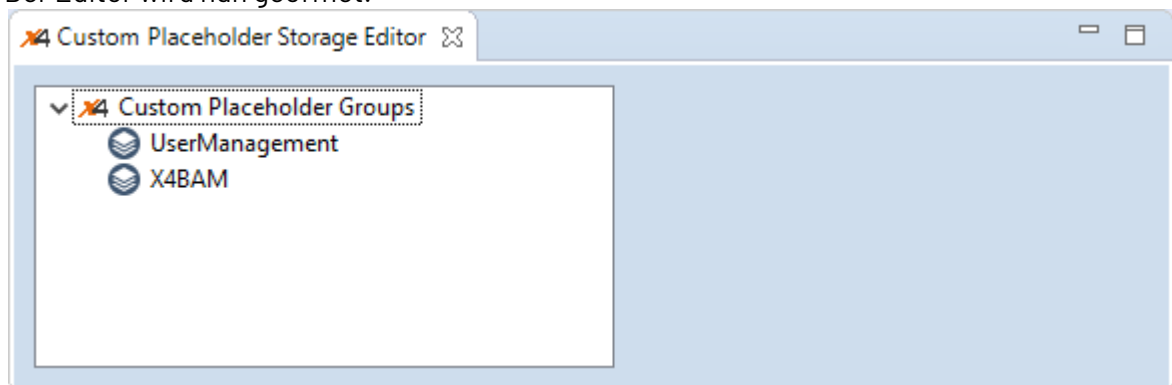
Folgende Platzhaltergruppen sind im *Custom Placeholder Storage Editor* standardmäßig vorhanden und dürfen nicht gelöscht werden:

Platzhaltergruppe	Benutzerdefinierte Platzhalter	Standardeinstellungen	Nutzung
X4BAM	<i>MonitoringDays</i>	<ul style="list-style-type: none"> Name: MonitoringDays Value: -1 (es gibt keine zeitliche Einschränkung für die Anzeige von Prozessausführungen) Type: Integer Full Qualified Name: PARAM_X4BAM_MonitoringDays 	Globaler zeitlicher Filter für die Einschränkung der Anzeige in der Monitoring-Oberfläche
UserManagement	<i>CertificateAuthority</i>	<ul style="list-style-type: none"> Name: CertificateAuthority Value: SoftProject Type: String Full Qualified Name: PARAM_UserManagement_CertificateAuthority 	Gibt den Namen des Zertifikat-Ausstellers an <i>i</i> Diese Information wird in den Zertifikat-Metadaten nach dessen Erstellung angezeigt.
	<i>CertificateValidity</i>	<ul style="list-style-type: none"> Name: CertificateValidity Value: 365 Type: Integer Full Qualified Name: PARAM_UserManagement_CertificateValidity 	Gibt die Gültigkeitsdauer des Zertifikats in Tagen an <i>i</i> Diese Information wird in den Zertifikat-Metadaten nach dessen Erstellung angezeigt.

3.6.3.1.2 Neue Platzhaltergruppe mit Platzhaltern anlegen

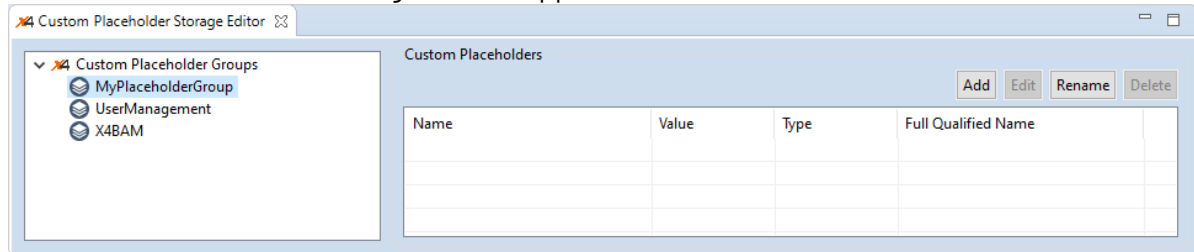
1. Den *Custom Placeholder Storage Editor* über das Menü **Management > Custom Placeholder Storage Editor** öffnen.

Der Editor wird nun geöffnet.

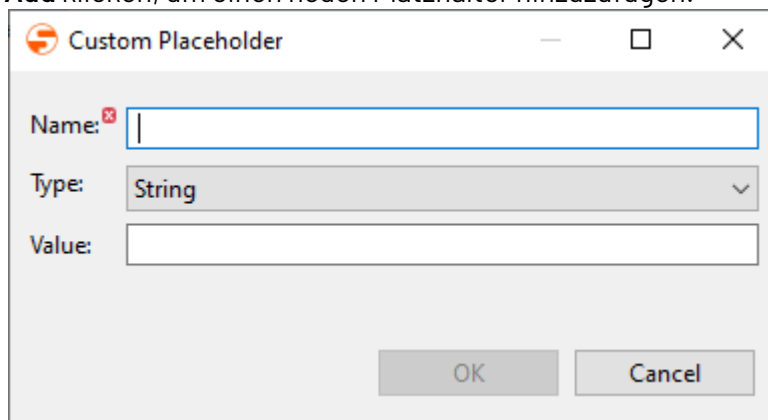



2. Auf **Custom Placeholder Groups** rechtsklicken, um das Kontextmenü zu öffnen.
3. **Add Group** im Kontextmenü wählen.

4. Namen für die Platzhaltergruppe eingeben und mit **OK** bestätigen.
Die neue Gruppe wurde nun hinzugefügt und erscheint als neuer Knoten unterhalb des Wurzelknotens.
5. Um neue Platzhalter hinzuzufügen, die Gruppe auswählen.



6. **Add** klicken, um einen neuen Platzhalter hinzuzufügen.



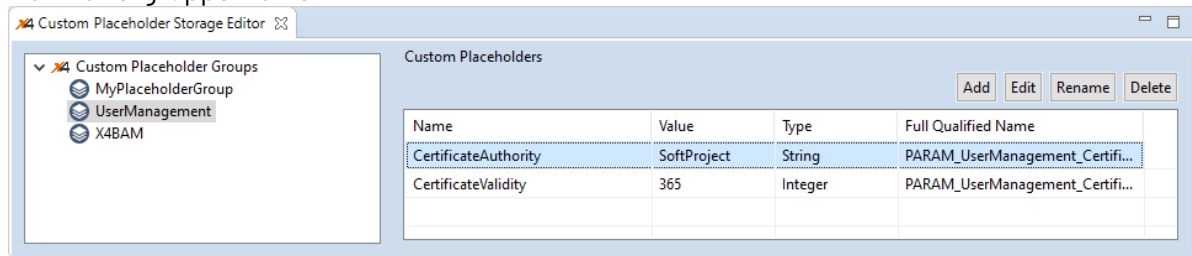
7. Platzhalter-Informationen eingeben:
 - Unter **Name**, den Platzhalter-Namen
 Folgende Zeichen sind hierbei erlaubt: `^[a-zA-Z][a-zA-Z0-9.\-]*$`
 - In **Type**, mit String, Boolean, Integer, Decimal oder Date den Platzhaltertyp spezifizieren
 - Unter **Value**, den Wert des Platzhalters
8. **OK** klicken, um den Platzhalter hinzuzufügen.
Der Platzhalter wurde nun angelegt. Der vollqualifizierte Name des Platzhalters wird dabei automatisch angelegt.

- ✓ Über Rechtsklick auf einen benutzerdefinierten Platzhalter lässt sich dessen Kontextmenü öffnen. Mit **Copy Full Qualified Name to Clipboard** kann anschließend dessen vollqualifizierter Name kopiert und an anderer Stelle eingefügt werden.

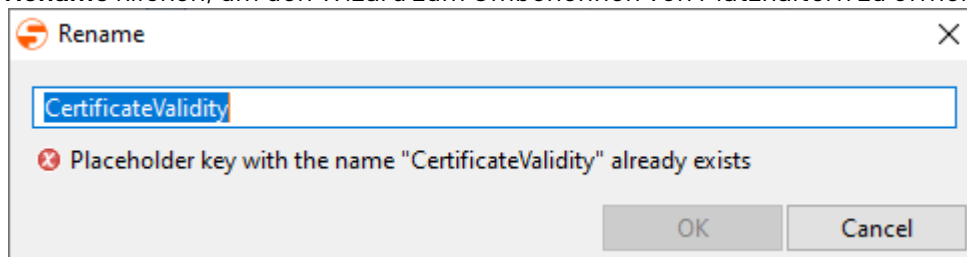
3.6.3.1.3 Platzhalter umbenennen

Um einen einzelnen Platzhalter umzubenennen und dessen Referenzen dabei automatisch zu aktualisieren, gehen Sie wie folgt vor:

1. Im *Custom Placeholder Storage Editor* den gewünschten Platzhalter in der entsprechenden Platzhaltergruppe wählen.



2. **Rename** klicken, um den Wizard zum Umbenennen von Platzhaltern zu öffnen.

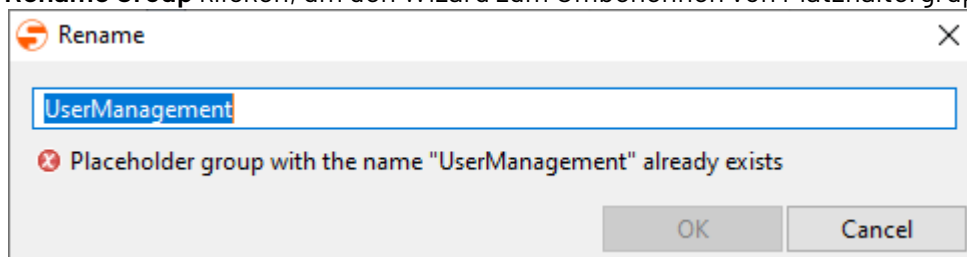


3. Neuen Platzhalternamen eingeben.
4. **OK** klicken, um die gewünschten Änderungen durchzuführen.

3.6.3.1.4 Platzhaltergruppe umbenennen

Um eine Platzhaltergruppe umzubenennen und deren Referenzen dabei automatisch zu aktualisieren, gehen Sie wie folgt vor:

1. Im *Custom Placeholder Storage Editor* auf die gewünschte Platzhaltergruppe rechtsklicken.
2. **Rename Group** klicken, um den Wizard zum Umbenennen von Platzhaltergruppen zu öffnen.



3. Neuen Namen eingeben.
4. **OK** klicken, um die gewünschten Änderungen durchzuführen.

3.6.3.2 Platzhalter über den Custom Placeholder Storage Manager Adapter verwalten

Mit dem *Custom Placeholder Storage Manager* Adapter lassen sich benutzerdefinierte Platzhalter anlegen und verwalten. Dabei bietet der Adapter verschiedene Operationen zur Verwaltung benutzerdefinierter Platzhalter, etwa zum Hinzufügen oder Löschen, aber auch zum Import und Export von Platzhaltern oder Platzhaltergruppen.


3.6.3.3 Platzhalter im X4 Control Center steuern

Während des Betriebs können Fachadministratoren im X4 Control Center die globale Platzhalterkonfiguration jederzeit einsehen und anpassen. Neue Prozess-Instanzen werden anschließend mit geänderten Parametern ausgeführt


3.7 XSL-Hilfsfunktionen

Welche Hilfsfunktionen von der X4 Suite zur Verwendung innerhalb von XSL-Mappings bereitstellt werden, erfahren Sie in diesem Abschnitt.

3.7.1 TimeHelper (Veraltet)

 Die Datums-/Uhrzeitfunktionen der Java-Klasse `de.softproject.xsl.TimeHelperDate` zur Verwendung innerhalb von XSL-Mappings sind veraltet und werden in künftigen Versionen der X4 Suite abgekündigt.


Die Java-Klasse `de.softproject.xsl.TimeHelperDate` befindet sich innerhalb der X4 Server-Bibliothek `x4-client.jar`
 (<X4>\Server\<wildfly>\modules\system\layers\base\de\softproject\x4\client\main).


 Ein XSL-Mapping, das diese Hilfsklasse verwendet, kann nicht in der Transformations-Vorschau des Mapping Editor ausgeführt werden.

3.7.1.1 Datum/Uhrzeit-Methoden





addDay <i>java.util.Date</i> date, <i>int</i> amount)	Fügt einem <code>java.util.Date</code> -Objekt eine beliebige ganzzahlige Anzahl an Tagen hinzu und gibt ein Objekt vom Typ <code>static java.util.Date</code> zurück
addHour <i>java.util.Date</i> date, <i>int</i> amount)	Fügt einem <code>java.util.Date</code> -Objekt eine beliebige ganzzahlige Anzahl an Stunden hinzu und gibt ein Objekt vom Typ <code>static java.util.Date</code> zurück
addMillisecond <i>java.util.Date</i> date, <i>int</i> amount)	Fügt einem <code>java.util.Date</code> -Objekt eine beliebige ganzzahlige Anzahl an Millisekunden hinzu und gibt ein Objekt vom Typ <code>static java.util.Date</code> zurück
addMinute <i>java.util.Date</i> date, <i>int</i> amount)	Fügt einem <code>java.util.Date</code> -Objekt eine beliebige ganzzahlige Anzahl an Minuten hinzu und gibt ein Objekt vom Typ <code>static java.util.Date</code> zurück

addMonth (<i>java.util.Date</i> date, <i>int</i> amount)	Fügt einem <i>java.util.Date</i> -Objekt eine beliebige ganzzahlige Anzahl an Monaten hinzu und gibt ein Objekt vom Typ <i>static java.util.Date</i> zurück
addSecond (<i>java.util.Date</i> date, <i>int</i> amount)	Fügt einem <i>java.util.Date</i> -Objekt eine beliebige ganzzahlige Anzahl an Sekunden hinzu und gibt ein Objekt vom Typ <i>static java.util.Date</i> zurück
addYear (<i>java.util.Date</i> date, <i>int</i> amount)	Fügt einem <i>java.util.Date</i> -Objekt eine beliebige ganzzahlige Anzahl an Jahren hinzu und gibt ein Objekt vom Typ <i>static java.util.Date</i> zurück
asMilliseconds (<i>java.util.Date</i> date)	Gibt ein <i>java.util.Date</i> -Objekt als Zeit in Millisekunden seit dem 1.1.1970 als Zahl vom Typ <i>static long</i> zurück
canParse (<i>java.lang.String</i> format, [<i>java.lang.String</i> timezone,] <i>java.lang.String</i> value)	Überprüft das Format einer Datums mit optionalem Parameter <i>timezone</i> für die Zeitzone und gibt ein Objekt vom Typ <i>static boolean</i> zurück  Hier wird lediglich das Format und nicht die Gültigkeit des Ergebnisses überprüft.
canParseDefaultFormat ([<i>java.lang.String</i> timezone,] <i>java.lang.String</i> value)	Überprüft, ob das verwendete Datumsformat mit optionalem Parameter <i>timezone</i> für die Zeitzone dem Standardformat entspricht und gibt ein Objekt vom Typ <i>static boolean</i> zurück
createDefaultFormat ([<i>java.lang.String</i> timezone])	Gibt das aktuelle Datum (mit optionalem Parameter <i>timezone</i> für eine spezifische Zeitzone) als Standardformat vom Typ <i>static java.text.SimpleDateFormat</i> aus
createFormat (<i>java.lang.String</i> format, [<i>java.lang.String</i> timezone])	Gibt das aktuelle Datum (mit optionalem Parameter <i>timezone</i> für eine spezifische Zeitzone) in einem Format vom Typ <i>static java.text.SimpleDateFormat</i> aus  Mehr Informationen zu den Datums-/Zeitformaten siehe http://docs.oracle.com/javase/7/docs/api/java/text/SimpleDateFormat.html
differenceInMilliseconds (<i>java.util.Date</i> from, <i>java.util.Date</i> to)	Berechnet den Unterschied zwischen zwei <i>java.util.Date</i> - Objekten in Millisekunden und gibt eine Zahl vom Typ <i>static long</i> zurück
format (<i>java.lang.String</i> format, [<i>java.lang.String</i> timezone,] <i>java.util.Date</i> date)	Formatiert ein <i>java.util.Date</i> - Objekt mit optionalem Parameter <i>timezone</i> für die Zeitzone in ein Objekt vom Typ <i>static java.lang.String</i>

formatDefaultFormat (<i>[java.lang.String timezone,]</i> <i>java.util.Date date</i>)	Formatiert ein <code>java.util.Date</code> -Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone in ein Standardformat vom Typ <code>static java.lang.String</code>
fromMilliseconds (<i>long milli</i>)	Konvertiert eine Zeitangabe in Millisekunden in ein Objekt vom Typ <code>static java.util.Date</code>
getDay (<i>java.util.Date date, [</i> <i>java.lang.String timezone]</i>)	Ermittelt den Tag aus einem <code>java.util.Date</code> -Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone, und gibt diesen als Objekt vom Typ <code>static int</code> zurück
getDayOfWeek (<i>java.util.Date date, [</i> <i>java.lang.String timezone]</i>)	<p>Ermittelt den Wochentag aus einem <code>java.util.Date</code>-Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone, und gibt diesen als Objekt vom Typ <code>static int</code> zurück</p> <p> Die Zählung der Wochentage beginnt bei Sonntag mit 1:</p> <ul style="list-style-type: none"> • Sonntag: 1 • Montag: 2 • Dienstag: 3 • Mittwoch: 4 • Donnerstag: 5 • Freitag: 6 • Samstag: 7
getDayOfWeekInMonth (<i>java.util.Date date, [</i> <i>java.lang.String timezone]</i>)	Ermittelt den Wochentag im Monat (z.B. "der wievielte Dienstag im Monat") aus einem <code>java.util.Date</code> -Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone, und gibt diesen als ganze Zahl vom Typ <code>static int</code> zurück
getDayOfYear (<i>java.util.Date date, [</i> <i>java.lang.String timezone]</i>)	Ermittelt den Tag des Jahres aus einem <code>java.util.Date</code> -Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone, und gibt diesen als ganze Zahl vom Typ <code>static int</code> zurück
getHour (<i>java.util.Date date, [</i> <i>java.lang.String timezone]</i>)	Ermittelt die Stunden aus einem <code>java.util.Date</code> -Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone, und gibt diese als ganze Zahl vom Typ <code>static int</code> zurück
getMilliSecond (<i>java.util.Date date, [</i> <i>java.lang.String timezone]</i>)	Ermittelt die Millisekunden aus einem <code>java.util.Date</code> -Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone, und gibt diese als ganze Zahl vom Typ <code>static int</code> zurück
getMinute (<i>java.util.Date date, [</i> <i>java.lang.String timezone]</i>)	Ermittelt die Minuten aus einem <code>java.util.Date</code> -Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone, und gibt diese als ganze Zahl vom Typ <code>static int</code> zurück

getMonth() <i>java.util.Date</i> date, [<i>java.lang.String</i> timezone])	<p>Ermittelt den Monat aus einem <code>java.util.Date</code>-Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone, und gibt diesen als ganze Zahl vom Typ <code>static int</code> zurück</p> <p> Die Zählung der Monate beginnt mit 0:</p> <ul style="list-style-type: none"> • Januar: 0 • Februar: 1 • März: 2 • April: 3 • Mai: 4 • Juni: 5 • Juli: 6 • August: 7 • September: 8 • Oktober: 9 • November: 10 • Dezember: 11
getSecond() <i>java.util.Date</i> date, [<i>java.lang.String</i> timezone])	<p>Ermittelt die Sekunden aus einem <code>java.util.Date</code>-Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone, und gibt diese als ganze Zahl vom Typ <code>static int</code> zurück</p>
getWeekOfMonth() <i>java.util.Date</i> date, [<i>java.lang.String</i> timezone])	<p>Ermittelt die Woche des Monats aus einem <code>java.util.Date</code>-Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone, und gibt diese als ganze Zahl vom Typ <code>static int</code> zurück</p>
getWeekOfYear() <i>java.util.Date</i> date, [<i>java.lang.String</i> timezone])	<p>Ermittelt die Kalenderwoche aus einem <code>java.util.Date</code>-Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone, und gibt diese als ganze Zahl vom Typ <code>static int</code> zurück</p>
getYear() <i>java.util.Date</i> date, [<i>java.lang.String</i> timezone])	<p>Ermittelt das Jahr aus einem <code>java.util.Date</code>-Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone, und gibt dieses als ganze Zahl vom Typ <code>static int</code> zurück</p>
isAfter() <i>java.util.Date</i> date, <i>java.util.Date</i> reference)	<p>Überprüft, ob ein <code>java.util.Date</code>-Objekt vor einem Referenzobjekt desselben Typs liegt, und gibt ein Objekt vom Typ <code>boolean</code> zurück</p>
isBefore() <i>java.util.Date</i> date, <i>java.util.Date</i> reference)	<p>Überprüft, ob sich ein <code>java.util.Date</code>-Objekt zeitlich nach einem Referenzobjekt desselben Typs befindet, und gibt ein Objekt vom Typ <code>boolean</code> zurück</p>
now()	<p>Ermittelt das aktuelle Datum und gibt dieses als Objekt vom Typ <code>static java.util.Date</code> zurück</p>

parse() <i>java.lang.String format,</i> <i>[java.lang.String timezone,]</i> <i>java.lang.String value)</i>	Liest eine Zeitangabe in <code>java.lang.String</code> -Format mit optionalem Parameter <code>timezone</code> für die Zeitzone aus, und gibt ein Objekt vom Typ <code>static java.util.Date</code> zurück
parseDefaultFormat() <i>[java.lang.String timezone,]</i> <i>java.lang.String value)</i>	Liest eine Zeitangabe in <code>java.lang.String</code> -Format mit optionalem Parameter <code>timezone</code> für die Zeitzone aus, und gibt ein Objekt vom Typ <code>static java.util.Date</code> im Standardformat zurück
setBeginOfDay() <i>java.util.Date date, [</i> <i>java.lang.String timezone])</i>	Liefert für das eingegebene <code>java.util.Date</code> -Objekt (mit optionalem Parameter <code>timezone</code> für die Zeitzone) den Beginn des Tages als <code>static java.util.Date</code> - Objekt
setBeginOfMonth() <i>java.util.Date date, [</i> <i>java.lang.String timezone])</i>	Liefert für das eingegebene <code>java.util.Date</code> -Objekt (mit optionalem Parameter <code>timezone</code> für die Zeitzone) den Beginn des Monats als <code>static java.util.Date</code> - Objekt
setBeginOfYear() <i>java.util.Date date, [</i> <i>java.lang.String timezone])</i>	Liefert für das eingegebene <code>java.util.Date</code> -Objekt (mit optionalem Parameter <code>timezone</code> für die Zeitzone) den Beginn des Jahres als <code>static java.util.Date</code> - Objekt
setDay() <i>java.util.Date date,</i> <i>[java.lang.String timezone,]</i> <i>int amount)</i>	Liefert für das eingegebene <code>java.util.Date</code> -Objekt (mit optionalem Parameter <code>timezone</code> für die Zeitzone) den Tag des Monats als <code>static java.util.Date</code> -Objekt  Ist die eingegeben Zahl höher als die Anzahl der tatsächlichen Tage des Monats, wird die Differenz auf den/ die Folgemonat(e) angerechnet.
setDayOfYear() <i>java.util.Date date,</i> <i>[java.lang.String timezone,]</i> <i>int amount)</i>	Liefert für das eingegebene <code>java.util.Date</code> -Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone das Datum als <code>static java.util.Date</code> - Objekt
setEndOfDay() <i>java.util.Date date, [</i> <i>java.lang.String timezone])</i>	Liefert für das eingegebene <code>java.util.Date</code> -Objekt (mit optionalem Parameter <code>timezone</code> für die Zeitzone) das Ende des Tages als <code>static java.util.Date</code> - Objekt
setEndOfMonth() <i>java.util.Date date, [</i> <i>java.lang.String timezone])</i>	Liefert für das eingegebene <code>java.util.Date</code> -Objekt (mit optionalem Parameter <code>timezone</code> für die Zeitzone) das Ende des Monats als <code>static java.util.Date</code> - Objekt
setEndOfYear() <i>java.util.Date date, [</i> <i>java.lang.String timezone])</i>	Liefert für das eingegebene <code>java.util.Date</code> -Objekt (mit optionalem Parameter <code>timezone</code> für die Zeitzone) das Ende des Jahres als <code>static java.util.Date</code> - Objekt

setHour (<i>java.util.Date</i> date, <i>[java.lang.String</i> timezone,] <i>int</i> amount)	<p>Liefert für das eingegebene <code>java.util.Date</code>-Objekt (mit optionalem Parameter <code>timezone</code> für die Zeitzone) die Anzahl der Stunden als static <code>java.util.Date</code>- Objekt</p> <p> Ist die eingegebene Zahl größer als 24, wird die Stundendifferenz auf den/die Folgetag(e) angerechnet.</p>
setMillisecond (<i>java.util.Date</i> date, <i>[java.lang.String</i> timezone,] <i>int</i> amount)	<p>Liefert für das eingegebene <code>java.util.Date</code>-Objekt (mit optionalem Parameter <code>timezone</code> für die Zeitzone) die Anzahl der Millisekunden als static <code>java.util.Date</code>-Objekt</p>
setMinute (<i>java.util.Date</i> date, <i>[java.lang.String</i> timezone,] <i>int</i> amount)	<p>Liefert für das eingegebene <code>java.util.Date</code>-Objekt (mit optionalem Parameter <code>timezone</code> für die Zeitzone) die Anzahl der Minuten als static <code>java.util.Date</code>-Objekt</p> <p> Ist die eingegebene Zahl größer ist als 60, wird die Differenz auf die Folgestunde(n) angerechnet.</p>
setMonth (<i>java.util.Date</i> date, <i>[java.lang.String</i> timezone,] <i>int</i> amount)	<p>Liefert für das eingegebene <code>java.util.Date</code>-Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone den Monat als static <code>java.util.Date</code>- Objekt</p> <p> Die Zählung der Monate beginnt mit 0:</p> <ul style="list-style-type: none"> • Januar: 0 • Februar: 1 • März: 2 • April: 3 • Mai: 4 • Juni: 5 • Juli: 6 • August: 7 • September: 8 • Oktober: 9 • November: 10 • Dezember: 11 <p>Ist die eingegebene Zahl größer als 11, wird die Differenz auf das/die Folgejahr(e) angerechnet.</p>
setSecond (<i>java.util.Date</i> date, <i>[java.lang.String</i> timezone,] <i>int</i> amount)	<p>Liefert für das eingegebene <code>java.util.Date</code>-Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone die Sekunden als static <code>java.util.Date</code>- Objekt</p> <p> Ist die eingegebene Zahl größer ist als 60, wird die Differenz auf die Folgeminute(n) angerechnet.</p>

setYear() <i>java.util.Date</i> date, [<i>java.lang.String</i> timezone,] <i>int</i> amount)	Liefert für das eingegebene <code>java.util.Date</code> -Objekt mit optionalem Parameter <code>timezone</code> für die Zeitzone das Jahr als <code>static java.util.Date</code> -Objekt
---	---

3.7.1.2 Beispiel

In diesem Beispiel werden stellvertretend für alle Methoden Variablen mit je einem Methodenaufruf initialisiert und deren Rückgabewert in einem entsprechend benannten XML-Element ausgegeben. Hierfür werden im Kopf des XSL-Stylesheets zunächst die Namensräume `x4time` für die TimeHelper-Klasse sowie `sdf` fürs Java-SimpleDateFormat definiert. Zudem werden in Variable `date` und `from_date` zwei Variablen mit einem `java.util.Date`-Objekt für den weiteren Gebrauch als Beispieldaten initialisiert.

```

<xsl:stylesheet xmlns:sdf="java.text.SimpleDateFormat"
xmlns:x4time="de.softproject.xsl.TimeHelperDate" xmlns:xsl="http://www.w3.org/1999/
XSL/Transform" version="1.0">
  <xsl:output indent="yes" method="xml" media-type="text/xml"></xsl:output>
  <xsl:template match="/">

    <!-- Get date/time values -->
    <xsl:variable select="x4time:now()" name="date"></xsl:variable>
    <xsl:variable select="x4time:addDay($date, 1)" name="from_date"></xsl:variable>

    <Output>
      <!-- Output method return values -->
      <now><xsl:value-of select="$date"></xsl:value-of></now>

      <!-- method addDay -->
      <xsl:variable select="x4time:addDay($date, 8)" name="addDayDate"></
xsl:variable>
      <addDay type="date">
        <xsl:value-of select="$addDayDate"></xsl:value-of>
      </addDay>

      <!-- method asMilliseconds -->
      <xsl:variable select="x4time:asMilliseconds($date)" name="asMillisecondsDate"></
xsl:variable>
      <asMilliseconds type="date">
        <xsl:value-of select="$asMillisecondsDate"></xsl:value-of>
      </asMilliseconds>

      <!-- method canParse with Timezone-->
      <xsl:variable select="x4time:canParse('yyyy-MM-dd', 'CET','2013.07.111')>
name="canParseWithTimezone"></xsl:variable>
      <canParse type="withTimezone">
        <xsl:value-of select="$canParseWithTimezone"></xsl:value-of>
      </canParse>

      <!-- method canParseDefaultFormat -->
      <xsl:variable select="x4time:canParseDefaultFormat('2013-07-111 16:56:45.123')>
name="canParseDefaultFormatSimple"></xsl:variable>
      <canParseDefaultFormat type="simple">
        <xsl:value-of select="$canParseDefaultFormatSimple"></xsl:value-of>
      </canParseDefaultFormat>

      <!-- method createDefaultFormat with timezone -->
      <xsl:variable select="x4time:createDefaultFormat('PST')>
name="sdf_default_withTimezone"></xsl:variable>
      <createDefaultFormatTz type="withTimezone">
        <xsl:value-of select="sdf:format($sdf_default_withTimezone, $date)"></
xsl:value-of>
      </createDefaultFormatTz>

      <!-- method createFormat with timezone -->
      <xsl:variable select="x4time:createFormat('yyyy-MMM-dd EE','CET')>
name="sdf_simpleWithTimezone"></xsl:variable>
      <createFormatTz type="withTimezone">

```

```

        <xsl:value-of select="sdf:format($sdf_simpleWithTimezone, $date)"></
xsl:value-of>
    </createFormatTz>

    <!-- method differenceInMilliseconds: date to date -->
    <xsl:variable select="x4time:differenceInMilliseconds($date, $from_date)"
name="differenceInMillisecondsDate"></xsl:variable>
    <differenceInMilliseconds type="date">
        <xsl:value-of select="$differenceInMillisecondsDate"></xsl:value-of>
    </differenceInMilliseconds>

    <!-- method format with Timezone -->
    <xsl:variable select="x4time:format('yyyy-MM-dd z','CET',$date)"
name="formatWithTimezone"></xsl:variable>
    <formatTZ type="withTimezone">
        <xsl:value-of select="$formatWithTimezone"></xsl:value-of>
    </formatTZ>

    <!-- method formatDefaultFormat with timezone -->
    <xsl:variable select="x4time:formatDefaultFormat('PST',$date)"
name="formatDefaultFormatWithTimezone"></xsl:variable>
    <formatDefaultFormat type="withTimezone">
        <xsl:value-of select="$formatDefaultFormatWithTimezone"></xsl:value-of>
    </formatDefaultFormat>

    <!-- method fromMilliseconds -->
    <xsl:variable select="x4time:fromMilliseconds(1374483531161)"
name="fromMilliseconds"></xsl:variable>
    <fromMilliseconds type="date">
        <xsl:value-of select="$fromMilliseconds"></xsl:value-of>
    </fromMilliseconds>

    <!-- method getDayOfWeek with timezone -->
    <xsl:variable select="x4time:getDayOfWeek($date, PST)"
name="getDayOfWeekWithTimezone"></xsl:variable>
    <getDayOfWeekTZ type="withTimezone">
        <xsl:value-of select="$getDayOfWeekWithTimezone"></xsl:value-of>
    </getDayOfWeekTZ>

    <!-- method isAfter -->
    <xsl:variable select="x4time:isAfter($date, $from_date)" name="isAfter"></
xsl:variable>
    <isAfter type="date">
        <xsl:value-of select="$isAfter"></xsl:value-of>
    </isAfter>

    <!-- method now -->
    <xsl:variable select="x4time:now()" name="now"></xsl:variable>
    <now type="date">
        <xsl:value-of select="$now"></xsl:value-of>
    </now>

    <!-- method parse with timezone -->
    <xsl:variable select="x4time:parse('yyyy-MM-dd','PST','2013-07-22')"
name="parseWithTimezone"></xsl:variable>
    <parseTZ type="withTimezone">

```

```

        <xsl:value-of select="$parseWithTimezone"></xsl:value-of>
    </parseTZ>

    <!-- method parseDefaultFormat with timezone -->
    <xsl:variable select="x4time:parseDefaultFormat('PST','2013-07-22
05:24:30.376')" name="parseDefaultFormatWithTimezone"></xsl:variable>
    <parseDefaultFormatTZ type="withTimezone">
        <xsl:value-of select="$parseDefaultFormatWithTimezone"></xsl:value-of>
    </parseDefaultFormatTZ>

    <!-- method setBeginOfDay -->
    <xsl:variable select="x4time:setBeginOfDay($date)" name="setBeginOfDay"></
xsl:variable>
    <setBeginOfDay type="date">
        <xsl:value-of select="$setBeginOfDay"></xsl:value-of>
    </setBeginOfDay>


    <!-- method setDay with timezone -->
    <xsl:variable select="x4time:setDay($date, 32, PST)"
name="setDayDateWithTimezone"></xsl:variable>
    <setDayTZ type="WithTimezone">
        <xsl:value-of select="$setDayDateWithTimezone"></xsl:value-of>
    </setDayTZ>

    <!-- method setEndOfDay -->
    <xsl:variable select="x4time:setEndOfDay($date)" name="setEndOfDay"></
xsl:variable>
    <setEndOfDay type="date">
        <xsl:value-of select="$setEndOfDay"></xsl:value-of>
    </setEndOfDay>

    <!-- method setMinute with Timezone -->
    <xsl:variable select="x4time:setMinute($date, 72)" name="setMinuteDate"></
xsl:variable>
    <setMinuteTZ type="date">
        <xsl:value-of select="$setMinuteDate"></xsl:value-of>
    </setMinuteTZ>
</Output>
</xsl:template>
</xsl:stylesheet>


```

3.7.2 WorkingTimeCalculator (Veraltet)

 Die Arbeitszeit-Berechnungsfunktionen der Java-Klasse `de.softproject.xml.WorkingTimeCalculator` zur Verwendung innerhalb von XSL-Mappings sind veraltet und werden in künftigen Versionen der X4 Suite abgekündigt. Ab Version 6.0.0 der X4 Suite steht hierfür der Adapter [Working Time Calculator](#) zur Verfügung.

Die Java-Klasse `de.softproject.xml.WorkingTimeCalculator` befindet sich innerhalb der X4 Server-Bibliothek `x4-`

```
client.jar(
<X4>\Server\<wildfly>\modules\system\layers\base\de\softproject\x4\client\main).
```

 Ein XSL-Mapping, das diese Hilfsklasse verwendet, kann nicht in der Transformations-Vorschau des Mapping Editors ausgeführt werden.




3.7.2.1 Methoden zum Definieren der Arbeitszeit


Bevor die unten genannten Methoden für die Arbeitszeitberechnung verwendet werden können, ist es erforderlich, für das Objekt `WorkingTimeCalculator` die gewünschten Arbeitszeiten (Beginn und Ende des Arbeitstages, Anzahl der Wochenarbeitsstage, Urlaubstage etc.) über folgende Methoden zu definieren.

setHolidays <i>java.lang.String holidays</i>	Setzt für ein <code>WorkingTimeCalculator</code> -Objekt einen oder mehrere Feiertage als Zeichenkette gemäß <code>java.lang.String</code> nach dem Muster <code>yyyy-mm-dd, yyyy-mm-dd, ...</code> und gibt ein Objekt vom Typ <code>WorkingTimeCalculator</code> zurück <i>Beispiel:</i> <code>setHolidays('2013-12-24, 2013-12-25, 2013-12-26')</code> , <code>setHolidays('2016-4-15, 2016-4-16, 2016-4-18')</code>
setSpecialDays <i>java.lang.String specialDays</i>	Setzt für ein <code>WorkingTimeCalculator</code> -Objekt einen Tageszeitraum als Zeichenkette gemäß <code>java.lang.String</code> nach dem Muster <code>yyyy-mm-dd hh:MM-hh:MM, yyyy-mm-dd hh:MM-hh:MM, ...</code> und gibt ein Objekt vom Typ <code>WorkingTimeCalculator</code> zurück Dieser Tageszeitraum wird in jedem Fall als Arbeitszeit berechnet, besitzt also Vorrang gegenüber Feiertagen und Arbeitszeiten. <i>Beispiel:</i> <code>setSpecialDays('2010-12-06 11:30-12:0, 2010-01-06 10:30-15:30')</code>
setWorkingDayBegin <i>int hr,</i> <i>int min</i>	Setzt für ein <code>WorkingTimeCalculator</code> -Objekt den Beginn des Arbeitstags mit Stunden und Minuten jeweils als Ganzzahl und gibt ein Objekt vom Typ <code>WorkingTimeCalculator</code> zurück <i>Beispiel:</i> <code>setWorkingDayBegin(9,0)</code>
setWorkingDayBegin <i>java.lang.String hr,</i> <i>java.lang.String min</i>	Setzt für ein <code>WorkingTimeCalculator</code> -Objekt den Beginn des Arbeitstags mit Stunden und Minuten jeweils als Zeichenkette vom Typ <code>java.lang.String</code> und gibt ein Objekt vom Typ <code>WorkingTimeCalculator</code> zurück <i>Beispiel:</i> <code>setWorkingDayBegin('09', '00')</code>

setWorkingDayEnd (<i>int</i> hr, <i>int</i> min)	Setzt für ein <code>WorkingTimeCalculator</code> -Objekt das Ende des Arbeitstags mit Stunden und Minuten jeweils als Ganzzahl und gibt ein Objekt vom Typ <code>WorkingTimeCalculator</code> zurück <i>Beispiel:</i> <code>setWorkingDayEnd(17,0)</code>
setWorkingDayEnd (<i>java.lang.String</i> hr, <i>java.lang.String</i> min)	Setzt für ein <code>WorkingTimeCalculator</code> -Objekt das Ende des Arbeitstags mit Stunden und Minuten jeweils als Zeichenkette vom Typ <code>java.lang.String</code> und gibt ein Objekt vom Typ <code>WorkingTimeCalculator</code> zurück <i>Beispiel:</i> <code>setWorkingDayEnd('17','00')</code>
setWorkingDays (<i>java.lang.String</i> days)	Setzt für ein <code>WorkingTimeCalculator</code> -Objekt eine Liste der Arbeitstage pro Woche als Zeichenkette vom Typ <code>java.lang.String</code> nach dem Muster <code><Tagnummer>,<Tagnummer>,...</code> und gibt ein Objekt vom Typ <code>WorkingTimeCalculator</code> zurück Wochentage werden gemäß <code>java.util.Calendar</code> als numerische Liste angegeben, beginnend bei <code>1</code> mit dem Sonntag. <i>Beispiel:</i> <code>setWorkingDays('2,3,4,5,6')</code>

3.7.2.2 Arbeitszeit-Berechnungsfunktionen

addWorkingTimeMinutes(<i>WorkingTimeCalculator</i> wtc, <i>java.util.Date</i> from, <i>int</i> minutes)	<p>Fügt einem <code>java.util.Date</code>-Objekt eine beliebige ganzzahlige Anzahl an Minuten innerhalb der definierten Arbeitszeiten hinzu und gibt ein Objekt vom Typ <code>static java.util.Date</code> zurück</p> <p><i>Voraussetzung:</i> ein Objekt <code>WorkingTimeCalculator</code> wurde bereits korrekt initialisiert und als erstes Argument innerhalb einer XSL-Variable übergeben, siehe Methoden zum Definieren der Arbeitszeit</p>
addWorkingTimeMinutesString(<i>WorkingTimeCalculator</i> wtc, <i>java.util.Date</i> from, <i>int</i> minutes)	<p>Fügt einem <code>java.util.Date</code>-Objekt eine beliebige ganzzahlige Anzahl an Minuten innerhalb der definierten Arbeitszeiten hinzu und gibt eine Zeichenkette vom Typ <code>java.lang.String</code> zurück</p> <p><i>Voraussetzung:</i> ein Objekt <code>WorkingTimeCalculator</code> wurde bereits korrekt initialisiert und als erstes Argument innerhalb einer XSL-Variable übergeben, siehe Methoden zum Definieren der Arbeitszeit</p> <p> Nur mit <i>Xalan 2.7.1</i> kompatibel, nicht mit <i>Saxon 9</i>!</p>
addWorkingTimeMinutesString(<i>WorkingTimeCalculator</i> wtc, <i>java.lang.String</i> from, <i>int</i> minutes)	<p>Fügt einer Zeichenkette gemäß <code>java.lang.String</code> eine beliebige ganzzahlige Anzahl an Minuten innerhalb der definierten Arbeitszeiten hinzu und gibt eine Zeichenkette vom Typ <code>java.lang.String</code> zurück</p> <p><i>Voraussetzung:</i> ein Objekt <code>WorkingTimeCalculator</code> wurde bereits korrekt initialisiert und als erstes Argument innerhalb einer XSL-Variable übergeben, siehe Methoden zum Definieren der Arbeitszeit</p> <p> Nur mit <i>Xalan 2.7.1</i> kompatibel, nicht mit <i>Saxon 9</i>!</p> <p><i>Beispiel:</i> <code>addWorkingTimeMinutesString('Thu Jul 11 14:29:24 CEST 2013', 'Mon Jul 18 16:04:55 CEST 2013', 200)</code></p>
dateDiff(<i>java.util.Date</i> from, <i>java.util.Date</i> to)	<p>Vergleicht zwei <code>java.util.Date</code>-Objekte miteinander und gibt die Differenz in Millisekunden als Zahl vom Typ <code>static long</code> zurück</p> <p> Nur mit <i>Xalan 2.7.1</i> kompatibel, nicht mit <i>Saxon 9</i>!</p>

dateDiffWorkingMinutes <i>WorkingTimeCalculator wtc,</i> <i>java.util.Date from,</i> <i>java.util.Date to)</i>	<p>Vergleicht zwei <code>java.util.Date</code>-Objekte miteinander und gibt die Differenz innerhalb der definierten Arbeitszeiten in Millisekunden als Zahl vom Typ <code>long</code> zurück</p> <p><i>Voraussetzung:</i> ein Objekt <code>WorkingTimeCalculator</code> wurde bereits korrekt initialisiert und als erstes Argument innerhalb einer XSL-Variable übergeben, siehe Methoden zum Definieren der Arbeitszeit</p> <p> Nur mit <i>Xalan 2.7.1</i> kompatibel, nicht mit <i>Saxon 9!</i></p>
getOstern (<i>double year</i>)	<p>Ermittelt das Osterdatum für ein Jahr, das als vierstellige Zahl vom Typ <code>double</code> übergeben wird, und gibt eine Zeichenkette vom Typ <code>java.lang.String</code> zurück</p> <p><i>Beispiel:</i> <code>getOstern(2014)</code></p>
getWorkingTimeMinutes <i>WorkingTimeCalculator wtc,</i> <i>java.util.Date from,</i> <i>java.util.Date to)</i>	<p>Ermittelt die Anzahl der Minuten innerhalb der definierten Arbeitszeiten, die zwischen einem Start- und einem Enddatum jeweils vom Typ <code>java.util.Date</code> liegen, und gibt eine Zahl vom Typ <code>double</code> zurück</p> <p><i>Voraussetzung:</i> ein Objekt <code>WorkingTimeCalculator</code> wurde bereits korrekt initialisiert und als erstes Argument innerhalb einer XSL-Variable übergeben, siehe Methoden zum Definieren der Arbeitszeit</p>
getWorkingTimeMinutes <i>WorkingTimeCalculator wtc,</i> <i>int fromYear,</i> <i>int fromMonth,</i> <i>int fromDay,</i> <i>int fromHr,</i> <i>int fromMin)</i>	<p>Ermittelt die Anzahl der Minuten innerhalb der definierten Arbeitszeiten, die zwischen einem Startdatum (definiert als Ganzzahlen für Jahr, Monat, Tag, Stunde und Minute) und dem aktuellen Zeitpunkt liegen, und gibt eine Zahl vom Typ <code>double</code> zurück</p> <p><i>Voraussetzung:</i> ein Objekt <code>WorkingTimeCalculator</code> wurde bereits korrekt initialisiert und als erstes Argument innerhalb einer XSL-Variable übergeben, siehe Methoden zum Definieren der Arbeitszeit</p>
getWorkingTimeMinutes <i>WorkingTimeCalculator wtc,</i> <i>int fromYear,</i> <i>int fromMonth,</i> <i>int fromDay,</i> <i>int fromHr,</i> <i>int fromMin,</i> <i>java.util.Date to)</i>	<p>Ermittelt die Anzahl der Minuten innerhalb der definierten Arbeitszeiten, die zwischen einem Startdatum (definiert als Ganzzahlen für Jahr, Monat, Tag, Stunde und Minute) und einem Enddatum vom Typ <code>java.util.Date</code> liegen, und gibt eine Zahl vom Typ <code>double</code> zurück</p> <p><i>Voraussetzung:</i> ein Objekt <code>WorkingTimeCalculator</code> wurde bereits korrekt initialisiert und als erstes Argument innerhalb einer XSL-Variable übergeben, siehe Methoden zum Definieren der Arbeitszeit</p>

getWorkingTimeMinutes <i>WorkingTimeCalculator</i> wtc, <i>int</i> fromYear, <i>int</i> fromMonth, <i>int</i> fromDay, <i>int</i> fromHr, <i>int</i> fromMin, <i>int</i> toYear, <i>int</i> toMonth, <i>int</i> toDay, <i>int</i> toHr, <i>int</i> toMin)	<p>Ermittelt die Anzahl der Minuten innerhalb der definierten Arbeitszeiten, die zwischen einem Start- und Enddatum (jeweils definiert als Ganzzahlen für Jahr, Monat, Tag, Stunde und Minute) liegen, und gibt eine Zahl vom Typ <code>double</code> zurück</p> <p><i>Voraussetzung:</i> ein Objekt <code>WorkingTimeCalculator</code> wurde bereits korrekt initialisiert und als erstes Argument innerhalb einer XSL-Variable übergeben, siehe Methoden zum Definieren der Arbeitszeit</p>
---	---

3.7.2.3 Beispiel

```
<xsl:stylesheet xmlns:x4time="de.softproject.xsl.TimeHelperDate"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:x4wtc="de.softproject.xsl.WorkingTimeCalculator"
  exclude-result-prefixes="x4time x4wtc" version="1.0">
  <xsl:output indent="yes" method="xml" media-type="text/xml"></xsl:output>
  <xsl:template match="/">
    <Output>
      <!-- Create new WorkingTimeCalculator object and store it in variable WTC -->
      <xsl:variable select="x4wtc:new()" name="WTC1"></xsl:variable>

      <!-- Get a java.util.date object with current date/time -->
      <xsl:variable select="x4time:now()" name="date"></xsl:variable>
      <now><xsl:value-of select="$date"></xsl:value-of></now>

      <!-- Set 2013 holidays to new year's eve and christmas -->
      <xsl:variable select="x4wtc:setHolidays($WTC1,'2013-1-1, 2013-12-25')"
name="WTC2"></xsl:variable>

      <!-- Set extra working time to December 24th from (Santa's X-mas delivery
shift) -->
      <xsl:variable select="x4wtc:setSpecialDays($WTC2,'2013-12-24 23:0-23:59')"
name="WTC3"></xsl:variable>

      <!-- Set working day begin time to 9:00 AM for the WTC object variable -->
      <xsl:variable select="x4wtc:setWorkingDayBegin($WTC3,9,0)" name="WTC4"></
xsl:variable>

      <!-- Set working day end time to 5:00 PM (17:00) for the WTC object variable --
>
      <xsl:variable select="x4wtc:setWorkingDayEnd($WTC4, 17, 0)" name="WTC5"></
xsl:variable>

      <!-- Set working days to Monday, Tuesday, Wednesday, Thursday, Friday for the
WTC object variable -->
      <xsl:variable select="x4wtc:setWorkingDays($WTC5,'2,3,4,5,6')" name="WTC"></
xsl:variable>

      <!-- Add 500 working time minutes to a date -->
      <xsl:variable select="x4wtc:addWorkingTimeMinutes($WTC, $date,500)"
name="addWorkingTimeMinutes"></xsl:variable>
      <addWorkingTimeMinutes><xsl:value-of select="$addWorkingTimeMinutes"></
xsl:value-of></addWorkingTimeMinutes>

      <!-- Get Easter date for year 2014 -->
      <xsl:variable select="x4wtc:getOstern(2014)" name="getOstern"></xsl:variable>
      <getOstern><xsl:value-of select="$getOstern"></xsl:value-of></getOstern>

      <!-- Calculate working minutes between two java.util.Date objects -->
      <xsl:variable select="x4wtc:getWorkingTimeMinutes($WTC,$date,
$addWorkingTimeMinutes)" name="getWorkingTimeMinutes"></xsl:variable>
      <getWorkingTimeMinutes type="date"><xsl:value-of
select="$getWorkingTimeMinutes"></xsl:value-of></getWorkingTimeMinutes>
```


```
</Output>
</xsl:template>
</xsl:stylesheet>
```

3.7.3 checkXML

Wie die XML-Prüfungsfunktion `checkXML` aus der Java-Klasse `de.softproject.xml.XSLHelper` innerhalb von XSL-Mappings verwendet wird

Die Java-Klasse `de.softproject.xml.XSLHelper` befindet sich innerhalb der X4 Server-Bibliothek `x4-client.jar`

(`<X4>\Server\<wildfly>\modules\system\layers\base\de\softproject\x4\client\main`).

 Ein XSL-Mapping, das diese Hilfsklasse verwendet, kann nicht in der Transformations-Vorschau des Mapping Editors ausgeführt werden.

3.7.3.1 Methodenbeschreibung

checkXML <i>java.lang.String URL</i>	<p>Prüft ein Dokument, das über eine beliebige gültige URL als Zeichenkette gemäß <code>java.lang.String</code>, referenziert wird, ob es wohlgeformtes XML ist und gibt einen Wert vom Typ <code>boolean</code> zurück</p> <p>Gültige URL-Protokolle sind u. a.:</p> <ul style="list-style-type: none"> • <code>file:///</code> für lokale Dateien • <code>xstore://</code> für Repository-Dokumente • <code>context:</code> für Variablen im Kontext der aktuellen X4-Prozessinstanz
--	---

3.7.3.2 Beispiel

Im Folgenden XSL-Mapping wird auf ein XML-Dokument, das in einer Prozessvariable `Var1` zur Laufzeit eines zugehörigen X4-Prozesses hinterlegt wurde, über das `context:`-URL-Protokoll zugegriffen. Mit der eingebundenen Java-Methode `checkXML` erfolgt dabei innerhalb des XSL-Mappings zunächst eine Prüfung auf Wohlgeformtheit, gefolgt von einer Verarbeitung und einer Fehlerbehandlung.


```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet exclude-result-prefixes="xslhelper" version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:xslhelper="de.softproject.xsl.XSLHelper">
  <xsl:output indent="yes" media-type="text/xml" method="xml"/>
  <xsl:template match="/">
    <Output>
      <!-- Checking if Var1 is XML or not -->
      <!-- If true, output it -->
      <xsl:choose>
        <xsl:when test="xslhelper:checkXML('context:Var1')">
          <!-- Define an XSL variable and assign a process variable "Var1",
               that is accessible as an external document via context: protocol -->
          <xsl:variable name="Var1Content" select="document('context:Var1')"/>
          <xsl:attribute name="isXML">
            <xsl:text>Yes</xsl:text>
          </xsl:attribute>
          <!-- Output XML content of Var1 -->
          <Data>
            <xsl:copy-of select="$Var1Content"/>
          </Data>
        </xsl:when>
        <xsl:otherwise>
          <xsl:attribute name="isXML">
            <xsl:text>No</xsl:text>
          </xsl:attribute>
        </xsl:otherwise>
      </xsl:choose>
    </Output>
  </xsl:template>
</xsl:stylesheet>

```

3.8 XSLT-Funktionen

In diesem Abschnitt werden die von XSLT 1.0 bereitgestellten Funktionen beschrieben, die Sie in XSL-Mappings einsetzen können.

 Mit [eckigen Klammern] markierte Argumente sind optional.

current ()	<p>Gibt den Inhalt der aktuellen Knotengruppe zurück.</p> <p>Dies entspricht im Wesentlichen der gebräuchlicheren Angabe <code><xsl:value-of select="." /></code>.</p>
-------------------	--


document (URI, [Knotengruppe])	<p>Öffnet das in Argument URI angegebene externe XML-Dokument und gibt dessen Knoten als Knotengruppe zurück.</p> <p>Optional kann in Argument Knotengruppe eine Basis angegeben werden, auf die sich relative XPath-Angaben beziehen können. Innerhalb der X4 Suite lässt sich ein XML-Dokument per xstore-Protokoll im Repository aufrufen.</p> <p>Beispiel:</p> <pre><xsl:value-of select="document('xstore://Project/test.xml')/element" /></pre> <p>gibt vom externen XML-Dokument test.xml die Knotengruppe ab Element element zurück.</p>
element-available ([Zeichenkette])	<p>Ermittelt, ob die XSLT-Anweisung in Argument Zeichenkette (mit Namespace-Angabe) vom XSLT-Prozessor unterstützt wird und gibt einen Booleschen Wert zurück.</p> <p>Um XSL-Mappings fehlertoleranter zu gestalten, kann so geprüft werden, ob der XSLT-Prozessor eine hersteller-/produktspezifische Anweisung oder eine Anweisung einer anderen XSLT-Version unterstützt.</p> <p>Beispiel:</p> <pre><xsl:if test="element-available('xsl:template')"></pre> <p>gibt den Booleschen Wert <i>true</i> zurück.</p>
format-number (Zahl, Format, [Name])	<p>Konvertiert eine Zahl in Argument Zahl in eine Zeichenkette und wendet eine Formatierungsfestlegung in Argument Format an. Diese Festlegungen entsprechen der Java 1.1-Klasse <code>decimal_format</code>, die beispielsweise 0 als Ziffer, # als Ziffer ohne 0 und . als Dezimaltrennzeichen definiert.</p> <p>Optional kann ein Argument Name verwendet werden, das den Namen eines Formats angibt, der über die XSLT-Anweisung <code>xsl:decimal-format</code> definiert wurde.</p> <p>Beispiel:</p> <pre><xsl:value-of select="format-number(1234, '#.00')"/></pre> <p>gibt <i>1234.00</i> zurück.</p>
function-available (Zeichenkette)	<p>Ermittelt, ob die XPath- oder XSLT-Funktion in Argument Zeichenkette (lediglich die Bezeichnung der Funktion, ggf. mit Namespace-Angabe) in der Funktionsbibliothek des XSLT-Prozessors vorhanden ist und gibt einen Booleschen Wert zurück.</p> <p>Um XSL-Mappings fehlertoleranter zu gestalten, kann so geprüft werden, ob der XSLT-Prozessor eine hersteller-/produktspezifische Funktion oder eine Funktion einer anderen XSLT-/XPath-Version unterstützt.</p> <p>Beispiel:</p> <pre><xsl:value-of select="function-available('format-number')"/></pre> <p>gibt <i>true</i> zurück.</p>

generate-id ([Knoten])	<p>Erzeugt für den aktuellen Knoten eine eindeutige ID als XML-Name und gibt diesen als Zeichenkette zurück.</p> <p>Optional kann im Argument Knoten ein Kontextknoten angegeben werden, für den eine ID erzeugt werden soll. Wird eine Knotengruppe angegeben, so wird für den ersten Knoten eine ID erstellt. Ist die Knotengruppe leer, wird eine leere Zeichenkette zurückgegeben.</p> <p>Beispiel:</p> <p><code><xsl:value-of select="generate-id(element)"/></code> gibt z. B. <i>d43e1</i> zurück.</p>
key (Schlüssel, Wert)	<p>Wählt einen in Argument Schlüssel angegeben Index aus, der mit einem <code><xsl:key></code>-Element angelegt wurde und dessen Wert in Argument Wert mit einem der Knoteninhalte übereinstimmt, und gibt eine Knotengruppe zurück.</p> <p>Beispiel:</p> <p>Folgendes XML-Dokument soll verarbeitet werden:</p> <pre data-bbox="336 954 1433 1137"><root> <element>Hello</element> <element>World</element> </root></pre> <p>Im XSLT-Stylesheet wird mit dem <code><xsl:key></code>-Element ein Index definiert, der über die Knotengruppe <i>root/element</i> definiert wurde und den aktuellen Knoteninhalt (.) indiziert. Über die Funktion <code>key()</code> wird darauf zugegriffen:</p> <pre data-bbox="336 1346 1433 1592"><xsl:key name="key1" match="root/element" use="."/> <xsl:template match="/"> <output> <xsl:value-of select="key('key1', 'Hello')"/> </output> </xsl:template></pre> <p>Das XSLT-Stylesheet gibt <code><output>Hello</output></code> zurück.</p>
system-property (Eigenschaft)	<p>Gibt die in Argument Eigenschaft angegebene XSLT-Prozessor-Information als Zeichenkette zurück.</p> <p>Unterstützte Eigenschaften sind mindestens: <i>xsl:version</i>, <i>xsl:vendor</i> und <i>xsl:vendor-url</i>.</p> <p>Beispiel:</p> <p><code><xsl:value-of select="system-property('xsl:vendor')"/></code> gibt je nach XSLT-Prozessor z. B. <i>SAXON 9.1.0.2 from Saxonica</i> zurück.</p>

unparsed-entity-uri (Name)	<p>Ermittelt den URI der in Argument Name angegebenen ungeparsten Entity (wird beispielsweise zum Referenzieren von Grafikdateien verwendet) und gibt die URI als Zeichenkette zurück.</p> <p>Die Entity muss in demselben Dokument wie der aktuelle Knoten definiert sein.</p>
-----------------------------------	---

3.9 XPath-Funktionen

Welche von XPath 1.0 bereitgestellten Funktionen Sie in XSL-Mappings einsetzen können

 Mit [eckigen Klammern] markierte Argumente sind optional.

3.9.1 Knotengruppen-Funktionen (node-set)

count (Knotengruppe)	Gibt die Anzahl der Knoten in Knotengruppe als ganze Zahl zurück
id (Id)	<p>Wählt den Knoten mit der in Id angegebenen eindeutigen ID aus.</p> <p>Beispiel: <code><xsl:value-of select="id('abc')/child::para"/></code> Wählt Kindknoten para des Knotens mit der eindeutigen ID <i>abc</i> aus.</p>
last ()	<p>Gibt die Positionsnummer des letzten Knotens im aktuellen Kontext als ganze Zahl zurück.</p> <p>Beispiel: <code><xsl:if test="position() = last()"></code> prüft, ob der aktuelle Knoten der letzte Knoten ist.</p>
local-name ([Knotengruppe])	<p>Gibt aus einem Knotennamen mit Namensraumangabe den lokalen Teil des Namens als Zeichenkette zurück.</p> <p>Optional kann als Argument eine Knotengruppe ausgewählt werden.</p>
name ([Knotengruppe])	<p>Gibt den Namen des aktuellen Knotens als Zeichenkette zurück.</p> <p>Optional kann als Argument eine Knotengruppe ausgewählt werden.</p>
namespace-uri ([Knotengruppe])	<p>Gibt den Namespace-URI (Uniform Resource Identifier) des aktuellen Element- oder Attributknotens als Zeichenkette zurück.</p> <p>Optional kann als Argument eine Knotengruppe ausgewählt werden.</p> <p>Beispiel: <code><xsl:value-of select="namespace-uri()"/></code> Gibt den Namespace-URI (sofern vorhanden) des aktuellen Knotens aus.</p>

position ()	Gibt die Positionsnummer des aktuellen Knotens als ganze Zahl zurück. Der erste Knoten erhält die Positionsnummer <i>1</i> .
--------------------	---

3.9.2 Boolesche Funktionen (boolean)

Boolesche XPath-Funktionen geben *true/false*-Werte zurück und können mit Vergleichsoperatoren verwendet werden.

boolean (Arg)	Prüft ein Argument <i>Arg</i> auf seine Wahrheit und gibt einen Booleschen Wert zurück. Alle Argumente, die eine positive Zahl, eine nicht-leere Zeichenkette oder eine nicht-leere Knotengruppe sind, werden als <i>true</i> interpretiert.
false ()	Gibt <i>false</i> zurück.
lang (Arg)	Gibt <i>true</i> zurück, wenn die in Argument <i>Arg</i> angegebene Sprache, die in einem <code>xml:lang</code> -Attribut angegeben wurde, mit der Sprachangabe des aktuellen Knotens mindestens zum Teil übereinstimmt. Beispiel: <code>lang("en")</code> gibt <i>true</i> zurück, wenn der aktuelle Knoten das Attribut <code>xml:lang="en"</code> besitzt oder wie in <code>xml:lang="en-US"</code> teilweise übereinstimmt.
not (Arg)	Gibt die Negation eines Booleschen Ausdrucks in <i>Arg</i> zurück. Beispiel: <code><xsl:if test="not(element > 1)"></code> gibt <i>true</i> zurück, wenn der Elementinhalt von <code>element</code> kleiner oder gleich <i>1</i> ist.
true ()	Gibt <i>true</i> zurück.

3.9.3 Number-Funktionen

XPath-Number-Funktionen geben ganze Zahlen zurück und können mit Vergleichsoperatoren verwendet werden.

ceiling (Arg)	Rundet das Argument <i>Arg</i> auf die nächsthöhere ganze Zahl auf und gibt diese zurück (z. B. <i>0.001</i> auf <i>1</i>). Falls nötig wird zunächst das Argument in eine Zahl konvertiert. Mathematisch korrekt ausgedrückt: Gibt die kleinste ganze Zahl zurück, die nicht kleiner ist als das Argument <i>Arg</i>
floor (Arg)	Rundet das Argument <i>Arg</i> auf die nächstniedrigere ganze Zahl ab und gibt diese zurück (z. B. <i>0.999</i> auf <i>0</i>). Falls nötig wird zunächst das Argument in eine Zahl konvertiert. Mathematisch korrekt ausgedrückt: Gibt die größte ganze Zahl zurück, die nicht größer ist als das Argument <i>Arg</i>

number ([Arg])	<p>Konvertiert den aktuellen Knoteninhalte in eine Zahl und gibt diese zurück.</p> <p>Optional kann ein Argument <i>Arg</i> eine Knotengruppe, eine Zeichenkette, einen Booleschen Wert oder ein anderes Objekt angeben, das konvertiert werden soll. Boolesche Werte <i>true</i> und <i>false</i> werden zu <i>1</i> und <i>0</i> konvertiert.</p> <p>Beispiel: Eine Zeichenkette <i>"3xy"</i> wird in <i>3</i> umgewandelt; <i>"xy"</i> als <i>NaN</i>(Not a Number) interpretiert.</p>
round (Arg)	<p>Rundet das Argument <i>Arg</i> zu einer ganzen positiven oder negativen Zahl auf oder ab und gibt diese zurück.</p> <p>Zahlen kleiner als <i>[Wert].5</i> werden abgerundet, bei Bruchgrößen größer oder gleich <i>[Wert].5</i> wird auf die nächsthöhere ganze Zahl aufgerundet.</p> <p>Beispiel: <i>0.49</i> wird zu <i>0</i> abgerundet, <i>-1.5</i> wird zu <i>-1</i> aufgerundet.</p>
sum (Knotengruppe)	<p>Bildet eine Gesamtsumme aller Knoteninhalte des Arguments Knotengruppe und gibt diese wenn möglich als Zahl zurück. Falls nötig werden zunächst die Knoteninhalte in Zahlen konvertiert.</p> <p>Können Werte nicht als Zahl interpretiert werden, dann werden diese als <i>NaN</i>(Not a Number) interpretiert.</p>

3.9.4 Zeichenketten-Funktionen (String)

XPath-String-Funktionen verarbeiten Objekte und geben Zeichenketten oder bei Zeichenketten-Prüfungen Boolesche Werte zurück.

concat (Arg1, Arg2 [, ...])	<p>Hängt mehrere Zeichenketten, die als Argumente <i>Arg1</i>, <i>Arg2</i> etc. übergeben werden, aneinander und gibt eine Zeichenkette zurück. Falls nötig werden zunächst die Argumente in Zeichenketten konvertiert.</p>
contains (Arg, Substring)	<p>Prüft, ob das Argument <i>Substring</i> in Argument <i>Arg</i> enthalten ist und gibt einen Booleschen Wert zurück. Falls nötig werden zunächst die Argumente in Zeichenketten konvertiert.</p> <p>Beispiel: <code><xsl:if test="contains('SoftProject', 'Project')"></code> gibt den Booleschen Wert <i>true</i> zurück.</p>
starts-with (Arg, Substring)	<p>Prüft, ob das Argument <i>Arg</i> mit dem Argument <i>Substring</i> beginnt und gibt einen Booleschen Wert zurück. Falls nötig werden zunächst die Argumente in Zeichenketten konvertiert.</p> <p>Beispiel: <code><xsl:if test="starts-with('SoftProject', 'Soft')"></code> gibt den Booleschen Wert <i>true</i> zurück.</p>

normalize-space ([Arg])	<p>Entfernt vom aktuellen Knoteninhalte die Leerzeichen am Anfang und Ende, ersetzt mehrfach aufeinanderfolgende Leerzeichen sowie Tabulatoren und Zeilenumbrüche durch ein einziges Leerzeichen und gibt den Knoteninhalte als Zeichenkette zurück.</p> <p>Optional kann ein Argument Arg verwendet werden, das falls nötig zunächst in eine Zeichenkette konvertiert wird.</p>
string ([Arg])	<p>Konvertiert den aktuellen Knoteninhalte in eine Zeichenkette und gibt diese zurück.</p> <p>Optional kann ein Argument Arg verwendet werden. Boolesche Werte werden als Zeichenkette <i>true</i> oder <i>false</i> ausgegeben, Kommazahlen werden nach Möglichkeit vereinfacht. Für erweiterte Konvertierungen von Zahlen zu Zeichenketten die XSLT-Funktion <code>format-numbers()</code> verwenden; siehe XSLT-Funktionen.</p> <p>Beispiel: <code><xsl:value-of select="string(4.00)"/></code> gibt die Zeichenkette <i>4</i> zurück.</p> <p>Prüfen, ob ein Wert eine Zahl ist: <code><xsl:if test="string(number(@value))!='NaN'"/></code>.</p>
string-length ([Arg])	<p>Gibt die Anzahl der Zeichen des aktuellen Knoteninhalts als ganze Zahl zurück. Falls nötig wird zunächst das Argument in eine Zeichenkette konvertiert.</p> <p>Optional kann ein Argument Arg verwendet werden, das falls nötig zunächst in eine Zeichenkette konvertiert wird.</p>
substring (Arg, Start [, Länge])	<p>Extrahiert aus der Zeichenkette in Argument Arg eine Teilzeichenkette, die ab Zeichen Start beginnt und eine Zeichenkette zurückgibt.</p> <p>Die Zählung der Zeichen beginnt bei 1. Optional kann das Argument Länge verwendet werden, das die maximale Länge der Teilzeichenkette angibt.</p>
substring-before (Arg, Substring)	<p>Prüft, ob die Zeichenkette in Arg vor der Teilzeichenkette in Argument Substring ein oder mehrere Zeichen besitzt und gibt diese als Zeichenkette zurück.</p> <p>Wenn Substring nicht in Arg enthalten ist, wird eine leere Zeichenkette zurückgegeben.</p> <p>Beispiel: <code><xsl:value-of select="substring-before('2008/04/01', '/')"/></code> gibt <i>2008</i> zurück.</p>
substring-after (Arg, Substring)	<p>Prüft, ob die Zeichenkette in Arg hinter der Teilzeichenkette in Argument Substring ein oder mehrere Zeichen besitzt und gibt die geänderte Zeichenkette zurück.</p> <p>Wenn Substring nicht in Arg enthalten ist, wird eine leere Zeichenkette zurückgegeben.</p> <p>Beispiel: <code><xsl:value-of select="substring-after('2008/04/01', '20')"/></code> gibt <i>08/04/01</i> zurück.</p>

translate (Arg, Substring 1, Substring 2)	<p>Sucht in Argument Arg nach der Zeichenkette in Argument Substring1, ersetzt diese durch die Zeichenkette in Argument Substring2 und gibt die geänderte Zeichenkette zurück. Falls nötig wird zunächst das Argument Arg in eine Zeichenkette konvertiert.</p> <p>Jeder Treffer von Substring1 wird durch Substring2 ersetzt.</p> <p>Beispiel: <code><xsl:value-of select="translate('hallo welt', 'abcdefghijklmnopqrstuvwxyz', 'ABCDEFGHIJKLMNOPQRSTUVWXYZ')"/></code> gibt HALLO WELT zurück.</p>
---	---

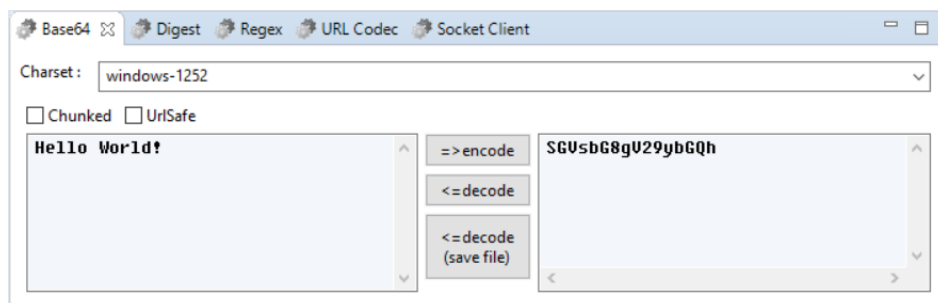
3.10 Entwickler-Hilfswerkzeuge

Eine Reihe von Entwickler-Hilfswerkzeugen unterstützen Prozess-Entwickler u.a. bei der schnellen URL-Kodierung, Base64-Konvertierung und der Erzeugung von Hash-Codes. In jeweils einer Sicht innerhalb des X4 Designers lassen sich damit die Funktionen des Eclipse-Plug-ins DevUtils grafisch nutzen.



Um das gewünschte Hilfswerkzeug kann über das Menü **View** aufgerufen werden.

Sicht	Erläuterung
Base64	Hier können Sie eine aus der Zwischenablage eingefügte Zeichenkette Base64-kodieren oder dekodieren



Verfügbare Optionen:

- **Charset:** Zeichensatz für die Zeichenkette
- **Chunked:** Base64-kodierte Zeichenkette blockweise ausgeben
- **UrlSafe:** Beim Base64-Kodieren nur Zeichen verwenden, die in URLs erlaubt sind



Beachten Sie:

- Die Schaltfläche **decode (save file)** speichert die dekodierte Base64-Zeichenkette als Datei in Ihrem lokalen Dateisystem.
- Derzeit lassen sich im X4 Designer nur Dateien aus Ihrem lokalen Dateisystem hineinziehen, jedoch keine Dateien, die per Drag & Drop aus dem **Repository Navigator** eingefügt wurden.

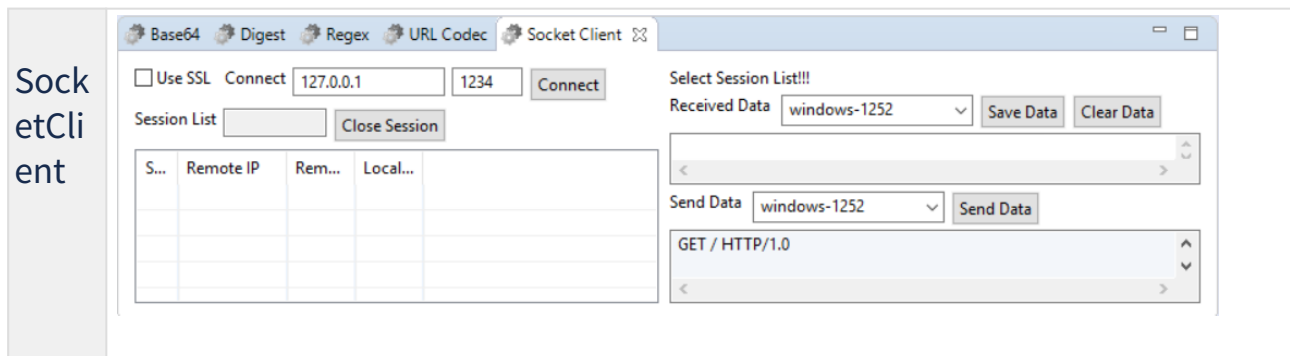
Digest Hier können Sie einen Hash-Wert für eine aus der Zwischenablage eingefügte Zeichenkette erzeugen

Verfügbare Optionen:

- **Charset:** Zeichensatz für die Zeichenkette
- **Digest:** Algorithmus wählen:
 - md5: MD5-Hash erzeugen (Standard)
 - sha: Mit dem Secure Hash Algorithm (SHA) einen Hash erzeugen
 - sha256: SHA-Hash mit 256 Bit erzeugen
 - sha384: SHA-Hash mit 384 Bit erzeugen
 - sha512: SHA-Hash mit 512 Bit erzeugen

Regex Hier können Sie einen regulären Ausdruck (*Regex*) auf eine aus der Zwischenablage eingefügte Zeichenkette anwenden, um den Ausdruck zu testen (Schaltfläche **match**) oder um die Zeichenkette gemäß dem Muster in Feld **Pattern** direkt zu ersetzen (Schaltfläche **replace**). Dabei können Sie verschiedene Regex-Syntaxoptionen aktivieren (z.B. CASE_INSENSITIVE oder UNICODE_CASE).

URLC odec Hier können Sie eine aus der Zwischenablage eingefügte Zeichenkette URL-kodieren oder dekodieren



3.11 URL-Protokolle

Erfahren Sie, welche X4-spezifische URL-Protokolle verfügbar sind, mit denen Sie in XSL-Mappings u. a. auf Prozess-Variablen und Repository-Dateien zur Prozess-Laufzeit zugreifen können.

3.11.1 context:-URL-Protokoll

Mit dem `context:-`URL-Protokoll können Sie zur Laufzeit eines Prozesses in XSL-Mappings auf Variablen zugreifen, die im Prozess über den Variable-Baustein definiert und vor dem XSL-Mapping-Baustein im Prozess angeordnet wurden. Das in der Variable gespeicherte Dokument kann im XSL-Mapping über die `document()`-Funktion eingebunden werden.

Aufbau der URL

Die URL sollte nach folgendem Muster aufgebaut sein (Schrägstriche nach dem URL-Handler können ggf. problematisch sein):

```
context:Variablenname
```

i Ein XSL-Mapping, das dieses URL-Protokoll verwendet, kann nicht in der Transformations-Vorschau des Mapping Editors ausgeführt werden.

Beispiel

In diesem Beispiel wurde in einem Prozess, der auch das folgende XSL-Mapping enthält, eine Variable `Var` über einen Variable-Baustein angelegt. Diese Variable speichert ein XML-Dokument, das im XSL-Mapping über die `document()`-Funktion gelesen und in einer XSL-Variable `DocVar` zwischengespeichert wird. Deren Knoten sind per XPath beliebig selektierbar.

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output encoding="UTF-8" method="xml" version="1.0"/>
  <xsl:template match="/">
    <result>
      <xsl:variable select="document('context:Var')" name="DocVar"/>
      <xsl:value-of select="$DocVar/child::ElemName"/>
    </result>
  </xsl:template>
</xsl:stylesheet>
```

3.11.2 xstore://-URL-Protokoll

Mit dem `xstore://`-URL-Protokoll können Sie zur Laufzeit eines Prozesses auf Dokumente im X4 Repository des aktuellen Benutzers zugreifen. In XSL-Mappings und in Adapter-Parametern können Sie damit beliebige Repository-Dateien referenzieren. XML-Dokumente können in XSL-Mappings über die `document()`-Funktion eingebunden werden.

Aufbau der URL

Eine `xstore://`-URL muss nach folgendem Muster aufgebaut sein:

```
xstore://<Dateipfad>
```

Bestandteile einer `xstore://`-URL:

- `xstore://`: Protokoll-Angabe; in der X4 Suite zur Laufzeit des jeweils ausgeführten Prozesses gültig.
- `Dateipfad`: Repository-Pfad zur Datei im Benutzerordner; Ordner werden durch Schrägstriche (/) getrennt.

Beachten Sie:

- Um Dokumente aus einem anderen *X4 Repository*-Benutzerordner zu lesen, das `x4db:/`-Protokoll verwenden; siehe [x4db:/-URL-Protokoll](#).
- Ein XSL-Mapping, das dieses URL-Protokoll verwendet, kann nicht in der Transformations-Vorschau des Mapping Editors ausgeführt werden.

Beispiel

Im folgenden Beispiel wird in einem XSL-Mapping ein XML-Dokument im Repository über die `document()`-Funktion aufgerufen und in einer XSL-Variable `externalFile` zwischengespeichert. Die Knoten des Variableninhalts sind per XPath beliebig selektierbar.

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output encoding="UTF-8" method="xml" version="1.0"/>
  <xsl:template match="/">
    <result>
      <xsl:variable select="document('xstore://Project/Folder/Data.xml')"
name="externalFile"/>
      <xsl:value-of select="$externalFile/child::ElemName"/>
    </result>
  </xsl:template>
</xsl:stylesheet>
```

3.11.3 x4db:/-URL-Protokoll

Mit dem `x4db:/`-URL-Protokoll können Sie zur Laufzeit eines Prozesses auf Dokumente in einem beliebigen *X4 Repository*-Benutzerordner zugreifen. In XSL-Mappings und in Adapter-Parametern können Sie damit beliebige Repository-Dateien referenzieren. XML-Dokumente können in XSL-Mappings über die `document()`-Funktion eingebunden werden.

Aufbau der URL

Eine `x4db:/`-URL muss nach folgendem Muster aufgebaut sein:

```
x4db:/<Benutzer-Repository>/<Dateipfad>
```

Bestandteile einer `x4db:/`-URL:

- `x4db:/`: Protokoll-Angabe; in der X4 Suite zur Laufzeit des jeweils ausgeführten Prozesses gültig.
- `Benutzer-Repository`: Repository-Benutzerordner ("*User collection*"), z.B. *1* oder *X4P*
- `Dateipfad`: Repository-Pfad zur Datei im Benutzerordner; Ordner werden durch Schrägstriche (/) getrennt.

Beachten Sie:

- Verwenden Sie das `x4db:/`-URL-Protokoll aus Sicherheitsgründen sparsam und ausschließlich zum Lesezugriff auf ein Dokument in einem anderen *X4 Repository*-Benutzerordner. Verwenden Sie ansonsten das `xstore://`-Protokoll; siehe „`xstore://`-URL-Protokoll“.
- Ein XSL-Mapping, das dieses URL-Protokoll verwendet, kann nicht in der Transformations-Vorschau des Mapping Editors ausgeführt werden.

Beispiel

Im folgenden Beispiel wird in einem XSL-Mapping ein XML-Dokument aus dem *X4 Repository*-Benutzerordner *101* über die `document()`-Funktion aufgerufen und in einer XSL-Variable `externalFile` zwischengespeichert. Die Knoten des Variableninhalts sind per XPath beliebig selektierbar.


```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output encoding="UTF-8" method="xml" version="1.0"/>
  <xsl:template match="/">
    <result>
      <xsl:variable select="document('x4db:/101/Project/Folder/Data.xml')"
name="externalFile"/>
      <xsl:value-of select="$externalFile/child::ElemName"/>
    </result>
  </xsl:template>
</xsl:stylesheet>
```

3.11.4 cryptstore://-URL-Protokoll

Mit dem `cryptstore://`-URL-Protokoll können Sie zur Laufzeit eines Prozesses in XSL-Mappings auf verschlüsselte Repository-Dokumente zugreifen und diese in einem Schritt entschlüsseln. Das Dokument kann im XSL-Mapping über die `document()`-Funktion eingebunden werden, die den direkten Zugriff auf das dekodierte Dokument ermöglicht.

Verwenden einer Cryptstore-Implementierung

Der *X4 Server* ermöglicht über das `cryptstore://`-URL-Protokoll die Ver- und Entschlüsselung von Repository-Elementen. Dabei wird der Bytestrom durch eine kundenspezifische Java-Klasse geleitet, um gelesen, entschlüsselt und weitergegeben zu werden. Diese kundenspezifische Klasse muss von `InputStream` ableiten und einen Konstruktor besitzen, der einen `InputStream` als einzigen Parameter entgegennimmt.

Empfohlen wird, die Klasse von `javax.crypto.CipherInputStream` abzuleiten und im Konstruktor das Objekt zu parametrieren. Die fertige Klasse muss als Java-Archiv (`.jar`) im Ordner `<WildFly>\modules\system\layers\base\de\softproject\x4\client\main` angelegt werden. Zusätzlich muss in der Datei `module.xml` innerhalb von `<WildFly>\modules\system\layers\base\de\softproject\x4\client\main` im Element `<resources>` ein Kindelement `<resource-root path="<Name des Java-Archivs>.jar"/>` hinzugefügt werden. Darüber hinaus muss der Klassenname in der Konfigurationsdatei `<X4>/X4config.xml` direkt hinter dem Element `<xstore/>` in einem Element `<cryptstoreStreamHandlerClass/>` als vollqualifizierter Klassenname angegeben werden.

Aufbau der URL

Die URL muss nach folgendem Muster aufgebaut sein:

```
cryptstore://Repository-Pfad/VerschlüsselteDatei
```

❗ Ein XSL-Mapping, das dieses URL-Protokoll verwendet, kann nicht in der Transformations-Vorschau des Mapping Editors ausgeführt werden.

Beispiel

Im folgenden Beispiel wird ein verschlüsseltes XML-Dokument im Repository innerhalb eines XSL-Mappings über die `document()`-Funktion aufgerufen und in einer XSL-Variable `externalFile` zwischengespeichert. Die Knoten des Variableninhalts sind per XPath beliebig selektierbar.

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output encoding="UTF-8" method="xml" version="1.0"/>
  <xsl:template match="/">
    <result>
      <xsl:variable select="document('cryptstore://Project/Folder/Data.xml')"
name="externalFile"/>
      <xsl:value-of select="$externalFile/child::ElemName"/>
    </result>
  </xsl:template>
</xsl:stylesheet>
```

Beispiel für eine mögliche Implementierung

```

import java.io.InputStream;
import java.security.InvalidKeyException;
import java.security.Key;
import java.security.NoSuchAlgorithmException;

import javax.crypto.Cipher;
import javax.crypto.CipherInputStream;
import javax.crypto.NoSuchPaddingException;
import javax.crypto.spec.SecretKeySpec;

public class TestCrypt extends CipherInputStream {

    public TestCrypt(InputStream stream) throws InvalidKeyException,
        NoSuchAlgorithmException, NoSuchPaddingException {
        super(stream, createCipher());
    }

    private static Cipher createCipher() throws InvalidKeyException,
        NoSuchAlgorithmException, NoSuchPaddingException {
        // Ciphers must NOT be cached!!
        Cipher cipher = Cipher.getInstance( "AES" );
        Key key = new SecretKeySpec( readKeyBytes(), "AES" );
        cipher.init( Cipher.DECRYPT_MODE, key );

        return cipher;
    }

    private static byte[] readKeyBytes() {
        // read your Key from where ever it is needed
        // the keyData may be cached
        byte[] keyData;
        return keyData;
    }
}

```

3.11.5 globalcontext:-URL-Protokoll

Mit dem `globalcontext:-URL`-Protokoll können Sie zur Laufzeit eines Prozesses in XSL-Mappings auf Kontextschlüssel zugreifen, die bereits über einen *Global Context Manipulation* Adapter definiert wurden. Das im Kontext gespeicherte Dokument kann im XSL-Mapping über die `document()`-Funktion eingebunden werden.

Globaler Kontext

Mit einem global definierten Kontext bietet die X4 Suite die Möglichkeit, ähnlich wie bei Prozess-Variablen beliebige Dokumente aus einem technischen Prozess in einer globalen Hash-Tabelle zwischenspeichern, sodass auch von anderen Prozessen auf diesen Kontext zugegriffen werden kann; siehe *Handbuch X4 Designer – Variablen-Baustein*. Die im globalen Kontext abgelegten Dokumente werden unter einem Schlüssel gespeichert und sind über die gesamte Laufzeit des X4 Servers verfügbar.

Aufbau der URL

Die URL muss nach folgendem Muster aufgebaut sein:

```
globalcontext:Schlüsselname
```

i Ein XSL-Mapping, das dieses URL-Protokoll verwendet, kann nicht in der Transformations-Vorschau des Mapping Editors ausgeführt werden.

Beispiel

In diesem Beispiel wurde in einem Prozess, der auch das folgende XSL-Mapping enthält, ein Kontextschlüssel *X4Experience.GlobalContext.Sample* über einen *Global Context Manipulation* Adapter angelegt. Dieser Kontexteintrag speichert ein XML-Dokument, das im XSL-Mapping über die `document()`-Funktion gelesen und in einer XSL-Variable *DocVar* zwischengespeichert wird. Deren Knoten sind per XPath beliebig selektierbar.


```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output encoding="UTF-8" method="xml" version="1.0"/>
  <xsl:template match="/">
    <result>
      <xsl:variable
        select="document('globalcontext:X4Experience.GlobalContext.Sample')"
        name="DocVar"/>
      <xsl:value-of select="$DocVar/child::Data"/>
    </result>
  </xsl:template>
</xsl:stylesheet>
```

3.12 Logging von technischen Prozessen

In technischen Prozessen können Sie technische und fachliche Informationen protokollieren und Log-Nachrichten an die Monitoring-Komponente oder über Log4j z. B. in das Server-Log ausgeben.


Es bestehen hierfür zwei Möglichkeiten:

- **Über Funktions-Adapter:** Die X4 Suite stellt die Adapter [X4 BAM Logging](#) und den [Log4J Logging](#) bereit, die Sie in Ihre technischen Prozesse einbinden und in verschiedener Hinsicht konfigurieren können. So kann z.B. das Ergebnis des vorangegangenen Prozess-Schritts im Content Log der Monitoring-Oberfläche gespeichert oder ein Status-Text im Server-Log ausgegeben werden.
- **Als Logging-Aktion auf Transitions:** Im X4 Designer können Sie eine Protokollierung einrichten, indem Sie zu Übergangslinien eine Logging-Aktion definieren. In der **Properties**-Sicht können Sie für markierte Übergangslinien die Eigenschaften **BAM Log** und **Log4j Log** setzen und so das Logging-Verhalten konfigurieren.

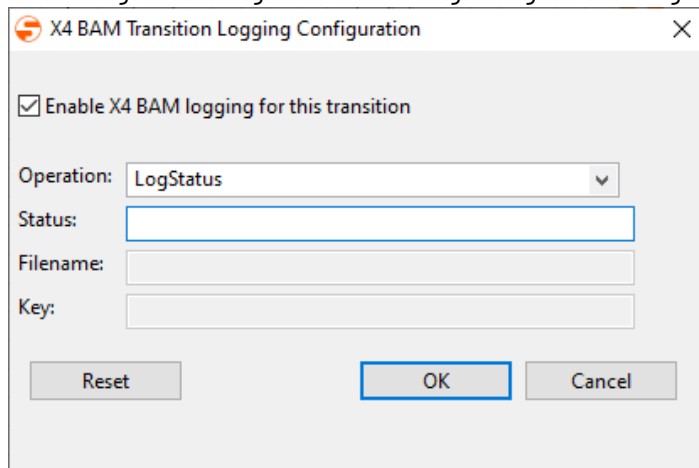
 Dynamische Parametrierung von dahinterliegenden Prozessbausteinen ist hierbei nicht möglich! Denn wenn Sie auf einer Übergangslinie eine Logging-Aktion definieren, wird intern ein versteckter Logging-Adapter im Prozess abgelegt.

Log-Nachrichten in der Monitoring-Oberfläche ausgeben


Um eine Logging-Operation für die Übergangslinien (*Transitions*) über BAM Log zu definieren, gehen Sie wie folgt vor:

1. Übergangslinie (*Transition*) hinter dem entsprechenden Prozessbaustein anklicken.
2. In der **Properties**-Sicht auf **BAM Logging** klicken, dann auf  klicken, um eine Logging-Aktion einzurichten.

Ein Dialog zur Konfiguration von Log-Ausgaben wird geöffnet.




The dialog box is titled "X4 BAM Transition Logging Configuration". It contains a checked checkbox labeled "Enable X4 BAM logging for this transition". Below this are four input fields: "Operation:" with a dropdown menu showing "LogStatus", "Status:" with an empty text box, "Filename:" with an empty text box, and "Key:" with an empty text box. At the bottom are three buttons: "Reset", "OK", and "Cancel".

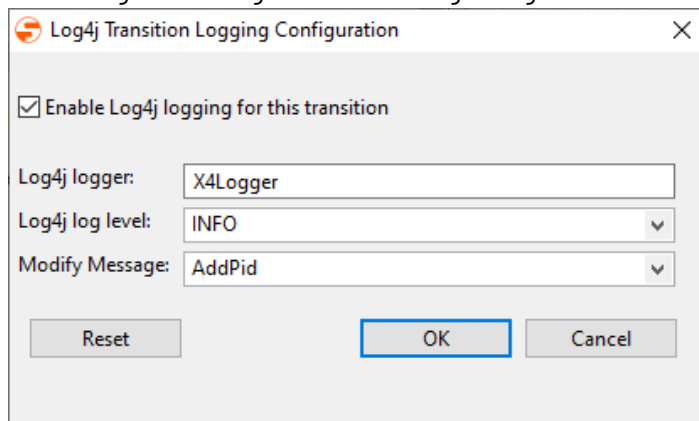
3. Enable **X4 BAM logging for this transition** aktivieren.
4. In **Operation** eine Logging-Operation wählen (standardmäßig *LogStatus*).
Die Operationen des X4 BAM Logging Adapters stehen hier zur Verfügung.
5. Je nach gewählter Operation weitere Einstellungen bearbeiten:
 - In **Status** einen benutzerdefinierten Status-Text eingeben, der ausgegeben werden soll (wenn **Operation** auf *LogStatus* gesetzt ist).
 - In **Filename** einen benutzerdefinierten Dateinamen eingeben, unter dem die protokollierten Input-Daten im Data Log bereitgestellt werden (wenn **Operation** auf *LogData* gesetzt ist).
 - In **Key** einen benutzerdefinierten Schlüsselnamen eingeben, mit dem die Log-Nachricht in der X4 BAM-Datenbank versehen werden soll.
6. **OK** klicken.
Die Statusinformationen bzw. Daten des vorherigen Prozessbausteins werden nun in der Monitoring-Oberfläche für die jeweilige Prozess-Instanz protokolliert. Das Symbol  auf der Übergangslinie zeigt nun an, dass eine Logging-Aktion definiert wurde.

Log-Nachrichten über Log4j ausgeben

Um eine Logging-Operation für die Übergangslinien (*Transitions*) über Log4j zu definieren, gehen Sie wie folgt vor:


1. Übergangslinie (*Transition*) hinter dem entsprechenden Prozessbaustein anklicken.
2. In der **Properties**-Sicht auf **Log4J Log** klicken, dann auf  klicken, um eine Logging-Aktion einzurichten.

Ein Dialog zur Konfiguration von Log-Ausgaben über Log4j wird geöffnet.



The dialog box is titled "Log4j Transition Logging Configuration". It contains a checked checkbox "Enable Log4j logging for this transition". Below this are three input fields: "Log4j logger:" with the value "X4Logger", "Log4j log level:" with a dropdown menu showing "INFO", and "Modify Message:" with a dropdown menu showing "AddPid". At the bottom are three buttons: "Reset", "OK", and "Cancel".

3. **Enable Log4j logging for this transition** aktivieren.
4. In **Log4j logger** den Namen des Loggers eingeben (Standard ist *X4Logger*).
5. In **Log4j log level** die Log-Ebene wählen, unter der die Nachricht in ausgegeben werden soll (standardmäßig *INFO*).
6. Wenn Sie die Log-Nachricht um die aktuelle Prozessinstanz-ID erweitern möchten: In **Modify Message** den Wert *AddPid* wählen.
7. **OK** klicken.


Die Statusinformationen bzw. Daten des vorherigen Prozessbausteins werden nun über Log4j für die jeweilige Prozess-Instanz protokolliert. Dies wird im Prozessdiagramm durch das Symbol  auf der Übergangslinie dargestellt.



3.13 Save Points für wiederaufsetzbare Prozesse verwenden

Mit den sogenannten *Save Points* können Sie in der X4 Suite den kompletten Zustand der Prozess-Engine mit den verarbeiteten Daten und Statusinformationen bei der Ausführung einer technischen Prozess-Instanz im Dateisystem persistieren.

Wird der technische Prozess von einem Save Point fortgesetzt, werden die persistierten Daten verwendet. Werden in der weiteren Ausführung des technischen Prozesses Daten, Adapter und Subprozesse erstmals aufgerufen, dann werden diese in ihrer aktuellen Version eingebunden.

Save Points können Sie im X4 Designer definieren, indem Sie im Prozessdiagramm eine **Übergangslinie** (*Transition*) markieren und in der **Properties**-Sicht für die Eigenschaft **Save Point** eine der folgenden Operationen festlegen:

 Save and Continue	<p>Save Point setzen und die Prozess-Ausführung direkt fortsetzen; Bei komplexeren Prozessen können Sie damit Zwischenstände speichern.</p> <p>Falls Fehler oder Ausfälle auftreten, kann der Prozess direkt ab dem letzten Save Point wiederaufgesetzt werden und muss nicht komplett neu ausgeführt werden.</p>
---	---

	Save and Stop	Save Point setzen, dann die Ausführung der Prozess-Instanz anhalten; Damit können Sie langlaufende Prozesse implementieren, die während der Ausführung "schlafen gelegt" werden können, bis beispielsweise Datenmanipulationen oder Wartungsarbeiten abgeschlossen sind. Anschließend können Sie die Prozess-Instanz fortsetzen.
	Delete	aktuellen Save Point löschen

**Einschränkungen**

- Datenbank-Transaktionen oder Cursors lassen sich nicht persistieren.
- Zu jeder Prozess-Instanz lässt sich ein Save Point speichern. Werden innerhalb eines Prozesses mehrere Save Points gesetzt, so wird der jeweils letzte Stand des Save Points verwendet.
- Im Debug-Modus können Save Points nicht verwendet werden.

Ausführung einer Prozess-Instanz über die Monitoring-Oberfläche fortsetzen

1. Die Monitoring-Oberfläche im Browser aufrufen.
2. Den entsprechenden angehaltenen Prozess in der Instanzsicht wählen.
Der Prozess wird nun in der Detailansicht angezeigt.
3. Unter **Savepoints** den entsprechenden Save Point wählen.
4. **Neustart/Fortsetzen** klicken, um die entsprechende Prozess-Instanz fortzusetzen.
Die Ausführung der Prozess-Instanz wird nun auf Basis des aktuellsten Save Points mit allen verarbeiteten Daten und dem persistierten Zustand der Prozess-Engine fortgesetzt.

3.14 Webservices via HTTP(S) bereitstellen

Mit der zentralen Webservice-Schnittstelle lassen sich technische Prozesse als Webservices via HTTP(S) bereitstellen, z.B. für ReSTful Webservices oder SOAP-Webservices. Die einzelnen Ressourcen eines Webservices lassen sich dabei von außen über einen HTTP-URL (Uniform Resource Locator) ansprechen.

Basis-URL und Funktionsweise

Die Basis-URL der zentralen Webservice-Schnittstelle lautet `http://<server>/X4/httpstarter/ReST` (z.B. `http://localhost:8080/X4/httpstarter/ReST`).

HTTP-Anfragen werden von der zentralen Webservice-Schnittstelle unmittelbar verarbeitet und als XML-Dokument an einen Prozess übergeben. Das Ergebnis des Prozesses, z. B. ein erzeugtes Dokument, kann direkt zurückgegeben werden.

Konfiguration

Webservices werden im X4 Designer über den grafischen *Webservice Configuration Editor* konfiguriert.

Rückgabewerte

Als Rückgabewert liefert die Webservice-Schnittstelle den aktuellen Statuscode (z. B. 201 Created) und je nach Anfrage die identifizierte Ressource.

Unterstützte Webservice-Typen

- **ReSTful-Webservices:** *Representational State Transfer* (ReST) ist ein Architekturstil, bei dem sich jede Ressource eines Informationssystems unter einem eigenen URI z. B. über HTTP ansprechen lässt. Ziel von ReSTful Services im X4 Server ist dementsprechend, Pfadstrukturen eines URI nachvollziehbar und transparent auf Prozesse abzubilden. Für ReSTful-Webservices lassen sich pro Ressource mehrere Operationen bzw. HTTP-Methoden zuordnen und mit jeweils einem Prozess verknüpfen, z.B. für Zugriffe per GET, POST oder DELETE. Hierbei werden folgende HTTP-Methoden unterstützt:

- GET
- PUT
- POST
- DELETE
- Sowie beliebige weitere Methoden, die aus gültigen XML-Zeichen bestehen.

Beim Webservice-Aufruf wird standardmäßig eine bestimmte XML-Datenstruktur an den X4-Prozess übergeben, siehe [ReSTful-Webservices: Input-XML-Struktur](#). Zudem erwartet die Webservice-Schnittstelle ein XML-Dokument als Ergebnis, das gemäß einer definierten Struktur aufgebaut ist und eine Reihe von Optionen bietet.

Die übertragenen Daten müssen in Base64 oder Base16 (hexadezimal) kodiert werden, da über HTTP nur Bytes übertragen werden. Um Probleme bei der Interpretation des Inhalts im Element <Body> zu vermeiden, wird dort direkter Text oder XML unterstützt. Der Text bzw. das XML-Dokument muss also zunächst in Bytes umgewandelt und Base64/Base16-kodiert werden, damit Sie die volle Kontrolle über die Bytes besitzen, die als Antwort zurückgegeben werden.

✓ Für weiterführende Informationen zum ReST-Architekturstil siehe die englischsprachige Dissertation von Roy T. Fielding unter <http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>.

- **SOAP-Webservices:** SOAP ist ein vom W3C standardisiertes Nachrichtenprotokoll (<http://www.w3.org/TR/soap>), welches eine standardisierte Kommunikation zwischen verteilten Anwendungen ermöglicht. SOAP-Webservices können in verschiedenen Varianten bereitgestellt werden, u.a. mit externer WSDL-Definition oder ohne WSDL-Definition, im RPC-Stil oder im Document-Modus. Als Nachrichtenformat wird bei SOAP-Webservices typischerweise XML für Anfrage (Request) und Ergebnis (Response) genutzt und Binärdaten werden ggf. per MIME anhängt (MTOM). SOAP-Webservices lassen sich auf die gleiche Weise wie ReSTful-Services via HTTP(S) bereitstellen, siehe [SOAP-Webservices bereitstellen](#). Der Service-Typ SOAP unterstützt die Standards SOAP 1.1 und SOAP 1.2. SOAP-Service-Definitionen können beliebig mit HTTP-ReST-Webservice-Definitionen und mit File-Webservice-Definitionen kombiniert werden.
- **File-Webservices:** Mit der zentralen Webservice-Schnittstelle lassen sich auch statische Datei-Ressourcen aus dem Repository bzw. dynamisch per X4-Prozess erzeugte Dokumente als Service via HTTP-GET bereitstellen, siehe [File-Webservices einrichten](#).

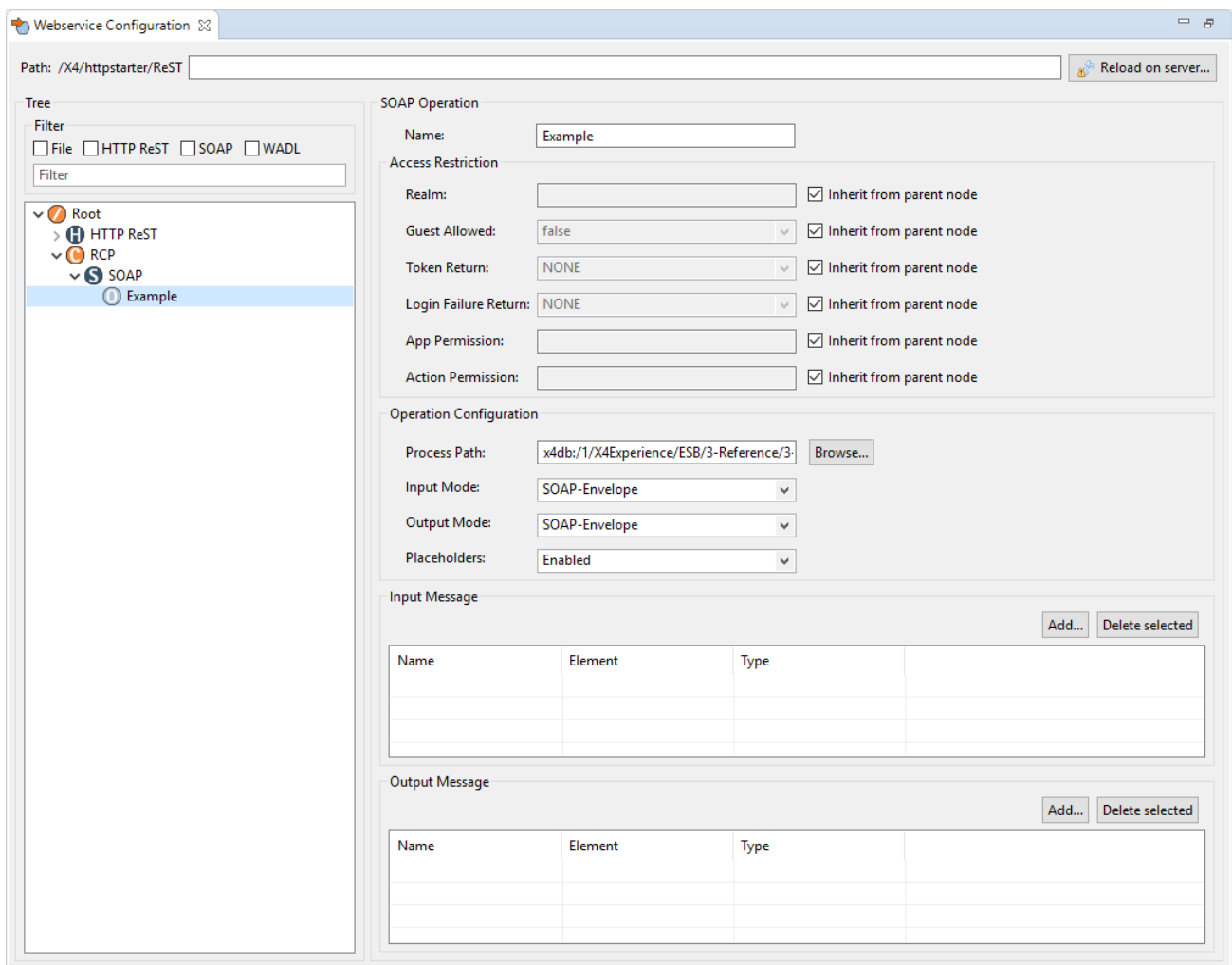
3.14.1 Webservice-Konfiguration grafisch bearbeiten

Die Webservice-Konfiguration lässt sich im X4 Designer über den grafischen *Webservice Configuration Editor* bearbeiten und im X4 Server aktualisieren. Technische Prozesse können so mit wenigen Mausklicks als HTTP-basierte Webservices verschiedener Ausprägungen (ReST, SOAP etc.) bereitgestellt werden.

 Um den Webservice Configuration Editor aufzurufen, Menü **Tools > Manage Webservices** wählen.

Die Oberfläche des Webservice Configuration Editors

Der *Webservice Configuration Editor* ist in folgende Bereiche unterteilt



Path: /X4/httpstarter/ReST Reload on server...

Tree

Filter
☐ File ☐ HTTP ReST ☐ SOAP ☐ WADL
 Filter

- Root
 - HTTP ReST
 - RCP
 - SOAP
 - Example

SOAP Operation

Name: Example

Access Restriction

Realm: ☒ Inherit from parent node

Guest Allowed: false ☒ Inherit from parent node

Token Return: NONE ☒ Inherit from parent node

Login Failure Return: NONE ☒ Inherit from parent node

App Permission: ☒ Inherit from parent node

Action Permission: ☒ Inherit from parent node

Operation Configuration

Process Path: x4db:/1/X4Experience/ESB/3-Reference/3 Browse...

Input Mode: SOAP-Envelope

Output Mode: SOAP-Envelope

Placeholders: Enabled

Input Message Add... Delete selected

Name	Element	Type

Output Message Add... Delete selected

Name	Element	Type

3.14.1.1 Path: Ressourcen-Pfad

Alle Webservices, die auf diesem Weg bereitgestellt werden, besitzen eine einheitliche Basis-URL. Die Basis-URL wird in Feld **Path** angezeigt und ist standardmäßig <http://localhost:8080/X4/httpstarter/ReST>.






In Feld **Path** können Sie beliebige URL-Parts anhängen, um zu prüfen, ob die gewünschten technischen Prozesse angesprochen werden. Dies ist insbesondere bei variablen und verschachtelten URL-Parts nützlich.

3.14.1.2 Tree: URL-Baumstruktur

In Bereich **Tree** wird die hierarchische Webservice-URL-Struktur bzw. werden die URL-Parts in einer auf- und zuklappbaren Baumstruktur angezeigt. Diese Baumstruktur können Sie per Rechtsklick über das Kontextmenü bearbeiten. Zudem können Sie via Drag&Drop Elemente innerhalb des Baums verschieben.

Über das **Filter**-Feld können Sie die angezeigte Baumstruktur anhand des Knotennamens filtern, sodass Strukturen, die das Suchwort nicht enthalten, ausgeblendet werden. Über ✖ oder durch Leeren des Feldes können Sie den Filter zurücksetzen. Zudem kann hier mit File, HTTP ReST, SOAP und WADL auch anhand des Service-Typs gefiltert werden.


Baumelemente

 Root	<p>Wurzeleintrag für die Basis-URL <server>/X4/httpstarter/ReST</p> <p>Für den Wurzeleintrag können Sie in Bereich Scope einen Sicherheitsbereich (Eigenschaft Realm), einen Gastzugang (Eigenschaft Guest Allowed) sowie ein zurückgegebenes Login-Token (Eigenschaft Token Return) global definieren, die standardmäßig an alle Elemente der Baumstruktur vererbt werden.</p>
 Constant	<p>Konstanter URL-Part</p> <p>Sämtliche Eigenschaften zur ausgewählten Konstante können Sie in Bereich Scope definieren oder Sicherheitseigenschaften vom übergeordneten Element erben lassen.</p>
 Variable	<p>Variabler URL-Part</p> <p>Sämtliche Eigenschaften zur ausgewählten Variable können Sie in Bereich Scope definieren oder Sicherheitseigenschaften vom übergeordneten Element erben lassen.</p>
 Remainder	<p>Komplett beliebige URL-Part-Definition</p> <p>Sämtliche Eigenschaften zum ausgewählten Remainder können Sie in Bereich Scope definieren oder Sicherheitseigenschaften vom übergeordneten Element erben lassen.</p>
 Include	<p>Referenzierte Webservice-Konfigurationsdatei als x4db:/-URL</p> <p>Über diesen Knotentyp lassen sich externe Webservice-Konfigurationsdateien mit der Endung .wsinc oder .restconfig referenzieren bzw. einbinden. Sämtliche (Sicherheits-)Eigenschaften können Sie in Bereich Scope definieren oder vom übergeordneten Element erben lassen.</p>

3.14.1.3 Konstanten, Variablen oder Remainder erstellen und verwalten


In der Baumstruktur den gewünschten Knoten (entspricht einem URL-Part des Webservices) markieren und per Rechtsklick im Kontextmenü eine Aktion wählen:

Add Constant	Konstante als Kindknoten anlegen, um einen konstanten URL-Part für Ihren Webservice zu definieren
Add Variable	Variable als Kindknoten anlegen, um mit Platzhaltern variable Bestandteile innerhalb der URL Ihres Webservices zu definieren
Add Remainder	Remainder als Kindknoten anlegen, um alle restlichen URL-Parts abzufangen
Add Include	Weitere Webservice-Konfigurationsdateien aus dem Repository referenzieren
Copy Service URL to Clipboard	Zugehörige Webservice-Ressourcen-URL in die Zwischenablage kopieren
Rename	Markierten Knoten umbenennen
Delete	Markierten Knoten löschen



 Bei den Knotennamen wird zwischen Groß- und Kleinschreibung unterschieden!

3.14.1.4 Sicherheitseigenschaften des Knotens definieren


Für den ausgewählten Knoten können Sie in **Access Restrictions** bei Bedarf (Sicherheits-)Eigenschaften definieren.

 Mit **Inherit from parent node** wird der Sicherheitsbereich vom übergeordneten Element übernommen. Diese Eigenschaft deaktivieren, um einen benutzerdefinierten Wert zu setzen.


Realm	Sicherheitsbereich für den aktuellen URL-Part
Guest Allowed	Wenn ein Sicherheitsbereich in Eigenschaft Realm definiert wurde, kann mit <i>true</i> ein Gastzugriff erlaubt werden

Token Return	<p>Zurückgegebenes Login-Token auswählen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>NONE</i> (Standard) • <i>CUSTOM_HEADER</i> • <i>COOKIE_SESSION</i> • <i>COOKIE_PERM</i> • <i>COOKIE_REMOVE</i> • <i>SEC_COOKIE_SESSION</i> • <i>SEC_COOKIE_PERM</i> • <i>SEC_COOKIE_REMOVE</i>
Login Failure Return	<p>In Kombination mit dem Login-Modul für <i>X4 Activities</i> können Sie mit Login Failure Return steuern, ob nach einem gescheiterten Anmeldeversuch auch der Grund zurückgegeben werden soll.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>NONE</i>(Standard): keine Information über den Grund des Scheiterns zurückmelden (wird im Server-Log protokolliert) • <i>HEADER</i>: den technischen oder fachlichen Grund im HTTP-Header X-X4-Login-Failed ausgeben.
App Permission	<p>Gibt an, für welche Anwendung die Berechtigungen geprüft werden sollen, z.B. <i>Custom Placeholders</i>. Siehe auch PEP Adapter.</p> <p> Wird nur die App Permission ohne Action Permission angegeben, wird lediglich geprüft, ob eine Berechtigung für die angegebene Anwendung vorhanden ist.</p>
Action Permission	<p>Gibt an, welche Aktion geprüft werden soll, z.B. <i>group:read</i> / <i>group:write</i>. Siehe auch PEP Adapter.</p> <p> Wird nur die Action Permission ohne App Permission angegeben, wird geprüft, ob eine Berechtigung für die angegebene Aktion vorhanden ist, unabhängig von einer Anwendung. Sollten mehrere Anwendungen über die selbe Aktion verfügen, gilt die Berechtigung für alle Anwendungen mit dieser Aktion.</p>
WADL Allowed	<p>Hier lässt sich festlegen, ob der Service bzw. die Operation im generierten WADL mit aufgeführt werden soll.</p>

3.14.1.5 Webservice-Konfiguration auf dem X4 Server aktualisieren

Sobald Sie im *Webservice Configuration Editor* die Konfiguration speichern, wird die Webservice-Konfiguration auf dem X4 Server überschrieben. Die neue Webservice-Konfiguration wird jedoch nicht automatisch angewendet. Über die Schaltfläche **Reload on Server**  **Reload on server...** können Sie die Konfiguration auf dem X4 Server aktualisieren, ohne diesen neu zu starten.

Beim Aktualisieren stehen folgende Optionen zur Verfügung geöffnet:

Save and Reload	Konfiguration auf dem X4 Server mit den Änderungen überschreiben und die aktualisierte Webservice-Konfiguration sofort bekannt machen.  Dieser Vorgang lässt sich nicht rückgängig machen!
Reload without saving	Webservice-Konfiguration auf dem X4 Server neu einlesen, ohne die Änderungen aus dem <i>Webservice Configuration Editor</i> anzuwenden. Dies ist ggf. sinnvoll, wenn zwischenzeitlich die Datei auf dem X4 Server anderweitig aktualisiert wurde und neu eingelesen werden soll.
Cancel	Abbrechen ohne die Webservice-Konfiguration auf dem X4 Server neu einzulesen.

3.14.2 ReSTful-Webservices bereitstellen

Für jeden URL-Part Ihres ReSTful-Webservices können Sie pro HTTP-Methode jeweils einen verarbeitenden Prozess zuordnen, z. B. für die *GET*-Methode einen Eintrag *GET*.

Verfügbare HTTP-Methoden

Innerhalb einer Operations-Zuordnung können folgende HTTP-Methoden jeweils einmal definiert werden:

- GET
- PUT
- POST
- DELETE
- Sowie beliebige weitere Methoden, die einen gültigen XML-Namen besitzen

3.14.2.1 Prozess als ReSTful HTTP-Webservice bereitstellen

Um im Webservice Configuration Editor einen technischen Prozess als Operation eines ReSTful Webservices bereitzustellen, gehen Sie wie folgt vor:

1. In der Baumstruktur bei Bedarf einen neuen Knoten (entspricht einem URL-Part des Webservices) erstellen, siehe [Webservice-Konfiguration grafisch bearbeiten](#).
2. Für den gewünschten Knoten (vom Typ Constant, Variable oder Remainder) über **Add HTTP ReST Service** einen neuen Service vom Typ *HTTP ReST* erstellen und ggf. (Sicherheits-)Eigenschaften anpassen.

3. Für den HTTP-ReST-Service über das Kontextmenü **Add HTTP ReST Operation** eine neue Operation anlegen oder einen bestehenden Eintrag in der **Operations**-Tabelle auswählen, der überschrieben werden soll.

i Sie können jeweils einmal pro Service eine Operation GET, POST, PUT, DELETE vergeben oder einen frei wählbaren Operationsnamen eingeben; dabei wird zwischen Groß-/Kleinschreibung unterschieden.

4. Einen Prozess aus dem Repository Navigator per Drag&Drop auf die gewünschte Operation ziehen, um den Prozesspfad zuzuordnen.
5. Input-Modus und Output-Modus für jede Operation definieren, um festzulegen welche XML-Struktur erwartet und erzeugt wird.
6. Ggf. die Nutzung von Platzhaltern aktivieren, um Parameter für die Webservice-Operation zu verarbeiten und Informationen aus dem HTTP-Request als Platzhalter bereitzustellen.

i Folgende Platzhalter mit Informationen aus dem HTTP-Request stehen bei Option Enabled innerhalb des X4-Prozesskontexts zur Verfügung, z.B. innerhalb von XSL-Mappings:

- REQUEST_PARAMETER_<Name>: Gewünschter Parameter (Groß-/Kleinschreibung wird beachtet!)
- REQUEST_HEADER_<NAME>: Gewünschter HTTP-Header (nur Großschreibung!)
- REQUEST_METHOD
- REQUEST_SCHEME
- REQUEST_PATH
- REQUEST_FULL_PATH
- REQUEST_REMOTE_ADDRESS
- REQUEST_REMOTE_PORT
- REQUEST_REMOTE_USER
- REQUEST_LOCAL_ADDRESS
- REQUEST_LOCAL_PORT
- REQUEST_SERVER_NAME
- REQUEST_SERVER_PORT

7. Sicherheits-Optionen wie Realm oder Guest Allowed für die Operation definieren.
8. Webservice-Konfiguration speichern und mit **Reload on Server** anwenden.

Beispiel: ReSTful Webservice-Konfiguration (Auszug XML-Speicherformat)

```
<Constant value="example">
  <Service type="Http">
    <Operation name="GET" process="x4db:/1/RestExamples/http/DirectOutput.wrf">
      <InputHandler type="HTTP-MetaXml" />
      <OutputHandler type="HTTP-Content" />
    </Operation>
  </Service>
</Constant>
```

3.14.2.2 ReSTful-Webservices: Input-XML-Struktur

Beim Aufruf eines Webservices wird ein Dokument an den entsprechenden technischen Prozess übergeben. Wenn Sie in der Webservice-Konfiguration als Input-Modus HTTP-Content definiert haben, handelt sich dabei ausschließlich um den Inhalt der HTTP-Entität. Media Type und Charset aus dem Header Content-Type werden auf das Input-Dokument übertragen.

Wenn als Input-Modus HTTP-MetaXml definiert wurde, wird beim Webservice-Aufruf ein XML-Dokument u.a. mit HTTP-Header-Informationen an den technischen Prozess übergeben, das nach folgendem Muster aufgebaut ist:

```
<Request>
  <Path>
    <Method>ReST-Methode (z. B. GET)</Method>
    <FullPath>Komplette URL inkl. konstantem und variablem URL-Part</FullPath>
    <Server>URL des Servers</Server>
    <RequestPath>Pfad, der im Aufruf verwendet wurde</RequestPath>
    <ContextPrefix>Basis-URL von X4 ReST (immer gleich)</ContextPrefix>
    <PathInfo>Konstanter und variabler URL-Part</PathInfo>
    <Parts>
      <Constant name="Konstanter URL-Part" />
      <Variable name="Name des variablen URL-Parts">tatsächl. URL-Part</Variable>
    </Parts>
  </Path>
  <Headers>
    <Header>
      <Name>Header-Name</Name>
      <Value>Header-Wert</Value>
    </Header>
    <!-- Oder bei Datumswerten: -->
    <Header>
      <Name>HTTP-Header-Name</Name>
      <Date timezone="Zeitzone">Header-Datumswert</Date>
    </Header>
  </Headers>
  <Query>
    <QueryString>Über die URL übergebene Parameter</QueryString>
    <Parameter name="Parameter-Name">Parameter-Wert</Parameter>
  </Query>
  <Security realm="Sicherheitsbereich">
    <Principal class="Prinzipalklasse" name="Prinzipal-Name" />
  </Security>
  <Body encoding="Datenkodierung">Übermittelte Daten</Body>
</Request>
```

Das Wurzel-Element <Request> enthält folgende Elemente:

- <Path>: Enthält u. a. die aufrufende URL und die einzelnen aufgeschlüsselten URL-Parts
- <Headers>: Enthält in jeweils einem Element <Header> die einzelnen HTTP-Header mit Name und Wert.
- <Query>: Enthält die Elemente <QueryString> und ggf. für jeden Parameter ein Element <Parameter>

- `<Security>`: Enthält die zugeordneten JAAS-Subjects (einfache Prinzipale oder Prinzipal-Gruppen), falls es sich um keinen Gastzugriff handelt
- `<Body>`: Enthält in Attribut `encoding` den Kodierungstyp *base64* und als Elementinhalt die entsprechend kodierten Nutzdaten

i Wenn ein Webservice für die Bereitstellung einer X4 Activities Classic Web-Anwendung definiert wird, wird die Anmeldung eines Benutzers über ein Token-Cookie (CUSTOM_HEADER) geregelt. Das Cookie bleibt erhalten, wenn der Token-Header in der HTTP-Response fehlt. Die Token-Generierung kann verhindert werden, indem in der Webservice-Konfiguration für CUSTOM_HEADER NONE vergeben wird.

3.14.2.3 ReSTful Webservices: Erwartete Output-XML-Struktur

Wenn für die jeweilige URI-Operation in der Webservice-Konfigurationsdatei der Output-Handler HTTP-MetaXml definiert wurde, erwartet die Webservice-Schnittstelle als Ergebnis des verarbeitenden Prozesses ein XML-Dokument, das nach folgendem Muster aufgebaut ist:

```
<Response>
  <Status code="HTTP-Status-Code"/>
  <Headers>
    <Header>
      <Name>HTTP-Header-Name</Name>
      <Value>Header-Wert</Value>
    </Header>
    <!-- Oder bei Datumswerten: -->
    <Header>
      <Name>HTTP-Header-Name</Name>
      <Date timezone="Zeitzone">Header-Datumswert</Date>
    </Header>
  </Headers>
  <Content encoding="Kodierung">Übermittelte Daten</Content>
</Response>
```

Das Wurzel-Element `<Response>` enthält folgende Elemente:

- `<Status>`: Enthält in Attribut `code` eine dreistellige Zahl mit dem HTTP-Statuscode (z. B. *202*).
- `<Headers>`: Enthält beliebig viele `<Header>`-Elemente mit HTTP-Header-Informationen. Jedes `<Header>`-Element enthält einen HTTP-Header, der über die Kindelemente `<Name>` und `<Value>` oder `<Date>` (bei Datumsangaben im Format *yyyy-MM-dd HH:mm:ss* gemäß *SimpleDateFormat*) definiert wird. Für Element `<Date>` kann im optionalen Attribut `timezone` die Zeitzone angegeben werden.
- `<Content>` (optional): Enthält in Attribut `encoding` den Kodierungstyp (*base64* oder *base16* bzw. *hex*) und als Elementinhalt die entsprechend kodierten Nutzdaten und/oder die Stream-Elemente.


```

<Content encoding="base64">
  ... Base64...
  <Stream url="http://remote.host/Some/Document?id=foo" transcoding="base64-
us-ascii"/>
  ... mehr Base64...
</Content>

```

3.14.3 SOAP-Webservices bereitstellen

Jeder SOAP Webservice wird über einen Knoten `<Services/>` innerhalb der Webservices-Konfiguration definiert. Dessen Attribute und Kindelemente variieren z.T. je nach Webservice-Typ.

Folgende Arten von SOAP Webservices stehen im Kindelement `<WsdL>` in Attribut mode zur Auswahl:

- **RPC:** Es wird vom X4 Server eine WSDL-Definition nach RPC-Stil aus den Konfigurationsdaten erzeugt, siehe [SOAP-Webservice im RPC-Stil bereitstellen](#)
- **DOCUMENT:** Es wird vom X4 Server eine WSDL-Definition nach Document-Stil aus den Konfigurationsdaten erzeugt, [SOAP-Webservice im Document-Modus bereitstellen](#)
- **EXTERNAL:** Externe WSDL-Definition für den SOAP Webservice verwenden, siehe [SOAP-Webservice mit externer WSDL-Definition bereitstellen](#)
- **NONE:** Es wird keine WSDL-Definition verwendet/bereitgestellt, siehe [SOAP-Webservice ohne WSDL-Definition bereitstellen](#)
- **PROCESS:** Prozess wählen, der die WSDL erzeugt und als HTTP-Content an den Aufrufer zurückgibt. Das Ergebnis des Prozesses wird dabei 1:1 zurückgegeben.

Schematischer Aufbau einer SOAP Webservice-Konfiguration (XML-Speicherformat)

```

<Service type="Soap">
  <SoapVersion>
    <SOAP11/> <!-- für SOAP 1.1-Unterstützung -->
    <SOAP12/> <!-- für SOAP 1.2-Unterstützung -->
  </SoapVersion>
  <WsdL mode="WSDL-Modus"/>
  <Namespace prefix="" namespace="" schemaLocation="URL zu einem XSD-Dokument"/>
  <Operation name="Operationsname" process="Pfad zum X4-Prozess als x4db:/-URL">
    <InputHandler type="Input-Modus"/>
    <OutputHandler type="Output-Modus"/>
    <Input>
      <Part name="Bezeichnung für einen Teil" type="Typ (optional)" element="Elementname (optional)" />
    </Input>
    <Output>
      <Part name="Bezeichnung für einen Teil" type="Typ (optional)" element="Elementname (optional)" />
    </Output>
  </Operation>
</Service>

```

Erläuterungen:

- Knoten `<Services type="Soap">` definiert einen HTTP-Webservice vom Typ SOAP.
- Knoten `<wsdl>` definiert über dessen Attribut `mode` den gewünschten WSDL-Modus (z.B. EXTERNAL, siehe oben).
- Knoten `<SOAP11/>` und `<SOAP12/>` ermöglichen zu definieren, welche SOAP-Versionen der Webservice nutzt (entweder eine oder beide parallel).
- In beliebig vielen Knoten `<Namespace>` lassen sich jeweils ein Namespace für die Definition von Webservices verwenden.
- In beliebig vielen Knoten `<Operation>` lassen sich jeweils eine Operation für den Webservice definieren (siehe unten).

Folgende Namespaces werden vom System definiert:

- `xmlns:xsd=" http://www.w3.org/2001/XMLSchema "`
- `xmlns:soap11=" http://schemas.xmlsoap.org/wsdl/soap/ "`
- `xmlns:soap12=" http://schemas.xmlsoap.org/wsdl/soap12/ "`
- `xmlns:wsdl=" http://schemas.xmlsoap.org/wsdl/ "`


Request Routing für SOAP-Services

- **GET-Requests:** Ein GET-Request wird beim WSDL-Modus NONE mit dem HTTP-Status 405 (Method not allowed) beantwortet. Im WSDL-Modus EXTERNAL wird mit einem HTTP redirect auf die konfigurierte WSDL-URL geantwortet. Bei den WSDL-Modi RPC und DOCUMENT generiert der X4 Server aus der Service-Konfiguration eine WSDL als Antwort.
- **POST-Requests:** Bei einem POST-Request wird der komplette Request-Body in den Speicher geladen, damit das Routing auf die Operation erfolgen kann.
 - a. Zunächst wird nach dem SOAP-Dokument gesucht.
 - Handelt es sich nicht um ein Multipart-Request, wird angenommen, dass der Request-Body das SOAP-Dokument enthält.
 - Handelt es sich um ein Multipart-Request (MTOM), wird nach dem Start-Part des MTOM-Pakets gesucht.
 - b. Das gefundene Dokument wird als XML gelesen und die SOAP-Version aus dem Namespace des Wurzelements ermittelt. Falls es sich dabei nicht um eine akzeptierte SOAP-Versionen handelt, wird mit einem entsprechenden SOAP-Fault geantwortet.
 - c. Aus dem XML-Dokument wird der Namespace und Localname des ersten Kindelements des SOAP-Body herangezogen, um die Operation im Service zu ermitteln.



- Bei den WSDL-Modi NONE, RPC und EXTERNAL wird mit dem konfigurierten Namespace und Localname der Operation verglichen.
- Für den WSDL-Modus DOCUMENT wird die Konfiguration des jeweils ersten Input-Parts verwendet. Wenn es sich um einen *Element*-Part handelt (d.h. das Attribut `element` enthält einen Wert) so wird der *Qualifiedname* des konfigurierten Elementes verwendet. Für *Type*-Parts (d.h. das Attribut `type` enthält einen Wert) wird der Name des Parts (Attribut `name`) verwendet.

- d. Anschließend wird gemäß Input-Modus der Request aufbereitet und an den konfigurierten Prozess für die Operation übergeben.

 Es können keine SOAP-Requests mit leerem Body-Element verarbeitet werden.



3.14.3.1 SOAP-Webservice im RPC-Stil bereitstellen

Wie Sie mit der X4 Suite einen technischen Prozess als SOAP-Webservice im RPC-Stil bereitstellen, der zudem eine automatisch aus der Service-Konfiguration erzeugte WSDL-Definition bereitstellt, wird im Folgenden beschrieben.

Voraussetzungen:

- Grundlegendes Wissen über den Aufbau von SOAP-Webservices im RPC-Stil
- Ein technischer Prozess pro Webservice-Operation, der die Verarbeitung der eingehenden Anfrage übernimmt. Der Prozess wird dabei über seine x4db: /-URL referenziert.
- Ein URL-Part, über den der SOAP-Webservice via HTTP erreichbar sein soll, z.B. über `http://localhost:8080/X4/httpstarter/ReST/MyRPCWebservice`. Diesen URL-Part haben Sie bereits typischerweise als Konstante innerhalb der Webservices-Konfiguration definiert.

Um einen technischen Prozess als SOAP-Webservice im RPC-Stil bereitzustellen, gehen Sie wie folgt vor:

1. Unter der gewünschten URL-Ressource (definiert z.B. durch einen konstanten URL-Part) über das Kontextmenü **Add SOAP Service** einen neuen Service-Eintrag vom Typ *SOAPAnlegen*.
2. SOAP-Version festlegen (entweder *SOAP 1.1*, *SOAP 1.2* oder Unterstützung für beide Standards).
3. WSDL-Modus in **WSDL Mode** und Zielpräfix in **Service Prefix** wählen.
4. Für jede Operation folgende Konfiguration hinterlegen:
 - Einen gültigen Namen für die Operation hinterlegen.
 SOAP-Webservices im RPC-Stil erfordern eindeutige *Localnames* für die Operationen.
 - Technischen Prozess, der die Webservice-Anfrage über den X4 Server verarbeitet, über dessen x4db: /-URL verknüpfen.
 Im *Webservice Configuration Editor* ist dies per Drag & Drop aus dem *Repository Navigator* möglich.
 - Input Handler und Output Handler für den SOAP-Webservice definieren.
Beispiel: SOAP-Content definieren, um das erste Element im `soap:Body` als Input zu verwenden sowie beim Output ein `soap:Body`- und `soap:Envelope`-Element zu erzeugen.
 - Erwartete Input-XML-Datenstruktur innerhalb von Element `<Input>` angeben, optional mit XML-Schemadefinition.
 - Zu erzeugende Ausgabe-Datenstruktur innerhalb von Element `<Output>` angeben.
5. Speichern und Webservice-Konfiguration mit **Reload on Server** anwenden.
 Der SOAP-Webservice im RPC-Stil ist ab sofort aktiv.

Die automatisch generierte zugehörige WSDL-Definition verwendet den RPC-Stil und wird bei einem HTTP-GET-Aufruf des SOAP-Webservice-URLs ausgeliefert.

Für den WSDL-Modus RPC erzeugt der X4 Server eine WSDL-Definition aus der Service-Konfiguration.

Die WSDL-Definition wird wie folgt erstellt:

- Für jedes Namespace-Element mit `schemaLocation`-Attribut wird unter `wsdl:types` ein `xsd:schema` mit `xsd:include` für dieschemaLocation erzeugt.
- Für jede Operation wird eine `wsdl:message` erzeugt, deren Name dem *Localname* der konfigurierten Operation entspricht. Innerhalb dieser Message wird für jedes Input-Part ein entsprechendes `wsdl:part` erstellt.
- Für jede Operation wird eine `wsdl:message` erzeugt, deren Name dem *Localname* der konfigurierten Operation, ergänzt um Response, entspricht. Innerhalb dieser Nachricht wird für jedes *Output*-Part ein entsprechendes `wsdl:part` erstellt.
- Für jede Operation wird eine `wsdl:operation` und damit ein `wsdl:portType` erzeugt. Der Name der `wsdl:operation` ist der *Localname* der konfigurierten Operation. Als Input-/Output-Nachricht werden die oben erzeugten Nachrichten verwendet.
- Je nach SOAP-Version werden entsprechende `wsdl:bindings` erzeugt (ausschließlich `literal` Bindings). Es wird immer eine leere SOAP-Action definiert. Bei SOAP-Webservices mit RPC-Stil wird für jede Operation ein `targetNamespace` deklariert, falls der Operationsname ein *Qualifiedname* ist (d.h. ein Name mit Doppelpunkt).

Beispiel: Konfiguration eines SOAP-Webservices im RPC-Modus

```

<Service type="Soap">
  <SoapVersion>
    <SOAP11/>
    <SOAP12/>
  </SoapVersion>
  <Wsd1 name="NameOfTheRpcService" targetPrefix="service" mode="RPC"/>
  <Namespace prefix="service" namespace="http://example.org/serviceSpace"/>
  <Namespace prefix="exampleSpace" namespace="http://example.org/namespace"
    schemaLocation="http://localhost:8080/X4/httpstarter/ReST/soapService/xsd/
namespace.xsd"/>
  <Namespace prefix="anotherSpace" namespace="http://example.org/different/
namespace"
    schemaLocation="http://localhost:8080/X4/httpstarter/ReST/soapService/xsd/
differentNamespace.xsd"/>
  <!-- RPC requires unique localnames for the operations -->
  <Operation name="exampleSpace:RpcOp1"
    process="x4db:/1/RestExamples/soap/DirectOutput.wrf">
    <InputHandler type="SOAP-Envelope"/>
    <OutputHandler type="SOAP-Envelope"/>
    <Input>
      <Part name="firstPart" element="exampleSpace:Operation1"/>
    </Input>
    <Output>
      <Part name="firstPart" element="exampleSpace:Operation1"/>
    </Output>
  </Operation>
  <Operation name="anotherSpace:RpcOp2"
    process="x4db:/1/RestExamples/soap/DirectOutput.wrf">
    <InputHandler type="SOAP-Envelope"/>
    <OutputHandler type="SOAP-Envelope"/>
    <Input>
      <Part name="firstPart" type="anotherSpace:SomeType"/>
    </Input>
    <Output>
      <Part name="Output" type="anotherSpace:SomeType"/>
    </Output>
  </Operation>
  <Operation name="RpcOp3"
    process="x4db:/1/RestExamples/soap/DirectOutput.wrf">
    <InputHandler type="SOAP-Envelope"/>
    <OutputHandler type="SOAP-Envelope"/>
    <Input>
      <Part name="Value" type="xsd:integer"/>
    </Input>
    <Output>
      <Part name="Value" type="xsd:String"/>
    </Output>
  </Operation>
</Service>

```

3.14.3.2 SOAP-Webservice im Document-Modus bereitstellen

Wie Sie mit der X4 Suite einen SOAP-Webservice im Document-Stil bereitstellen, wird im Folgenden beschrieben.



Voraussetzungen:

- Grundlegendes Wissen über den Aufbau von SOAP-Webservices im Document-Stil, bei welchen der Request/Response Body per XML-Schema (XSD) vollständig beschrieben sind.
- Ein technischer Prozess pro Webservice-Operation, der die Verarbeitung der eingehenden Anfrage übernimmt. Der Prozess wird dabei über seine `x4db:/`-URL referenziert.
- Ein URL-Part, über den der SOAP-Webservice via HTTP erreichbar sein soll, z.B. über `http://localhost:8080/X4/httpstarter/ReST/MyDocumentWebservice`. Diesen URL-Part haben Sie bereits typischerweise als Konstante innerhalb der Webservices-Konfiguration definiert, siehe [Webservice-Konfiguration grafisch bearbeiten](#).

Um einen technischen Prozess als SOAP-Webservice im Document-Stil bereitzustellen, gehen Sie wie folgt vor:

1. Unter der gewünschten URL-Ressource (definiert z.B. durch einen konstanten URL-Part) über das Kontextmenü **Add SOAP Service** einen neuen Service-Eintrag vom Typ *SOAP* anlegen.
2. SOAP-Version festlegen (entweder *SOAP 1.1*, *SOAP 1.2* oder Unterstützung für beide Standards).
3. WSDL-Modus in **WSDL Mode** und Zielpräfix in **Service Prefix** wählen.
4. Für jede Operation folgende Informationen hinterlegen:
 - Einen gültigen Namen für die Operation hinterlegen.
 - SOAP-Webservices im Document-Modus erfordern eindeutige *Localnames* für die Operationen.
 - Namespace-Deklaration für die Daten im Request/Response Body hinterlegen.
 - Technischen Prozess, der die Webservice-Anfrage über den X4 Server verarbeitet, über dessen `x4db:/`-URL verknüpfen.
 - Im *Webservice Configuration Editor* ist dies per Drag & Drop aus dem Repository Navigator möglich.
 - Input Handler und Output Handler für den SOAP-Webservice definieren.

Beispiel: SOAP-Content definieren, um das erste Element im `soap:Body` als Input zu verwenden sowie beim Output ein `soap:Body`- und `soap:Envelope`-Element zu erzeugen.
 - Erwartete Input-XML-Datenstruktur innerhalb von Element `<Input>` inklusive Namespace angeben.
 - Zu erzeugende Ausgabe-Datenstruktur innerhalb von Element `<Output>` inklusive Namespace angeben.
5. Speichern und Webservice-Konfiguration mit **Reload on Server** anwenden.
Der SOAP-Webservice im Document-Stil ist ab sofort aktiv.

Generierte WSDL-Definition im Document-Modus

Die automatisch generierte zugehörige WSDL-Definition verwendet den Document-Stil und wird bei einem HTTP GET-Aufruf der SOAP-Webservice-URLs ausgeliefert.

Für den WSDL-Modus DOCUMENT erzeugt der *X4 Server* eine WSDL-Definition aus der Service-Konfiguration. Die WSDL-Definition wird wie folgt erstellt:

- Für jedes Namespace-Element mit `schemaLocation`-Attribut wird unter `wsdl:types` ein `xsd:schema` mit `xsd:include` für die `schemaLocation` erzeugt.
- Für jede Operation wird eine `wsdl:message` erzeugt, deren Name dem *Localname* der konfigurierten Operation entspricht. Innerhalb dieser Message wird für jedes Input-Part ein entsprechendes `wsdl:part` erstellt.
- Für jede Operation wird eine `wsdl:message` erzeugt, deren Name dem *Localname* der konfigurierten Operation, ergänzt um Response, entspricht. Innerhalb dieser Nachricht wird für jedes *Output*-Part ein entsprechendes `wsdl:part` erstellt.
- Für jede Operation wird eine `wsdl:operation` und damit ein `wsdl:portType` erzeugt. Der Name der `wsdl:operation` ist der *Localname* der konfigurierten Operation. Als Input-/Output-Nachricht werden die oben erzeugten Nachrichten verwendet.
- Je nach SOAP-Version werden entsprechende `wsdl:bindings` erzeugt (ausschließlich `literal` Bindings). Es wird immer eine leere SOAP-Action definiert.

Beispiel: Konfiguration eines SOAP-Webservices im Document-Modus

```

<Service type="Soap">
  <SoapVersion>
    <SOAP11/>
    <SOAP12/>
  </SoapVersion>
  <Wsdli name="NameOfTheDocumentService" targetPrefix="service" mode="DOCUMENT"/>
  <Namespace prefix="service" namespace="http://example.org/serviceSpace"/>
  <Namespace prefix="exampleSpace" namespace="http://example.org/namespace"
    schemaLocation="http://localhost:8080/X4/httpstarter/ReST/soapService/xsd/
namespace.xsd"/>
  <Namespace prefix="anotherSpace" namespace="http://example.org/different/
namespace"
    schemaLocation="http://localhost:8080/X4/httpstarter/ReST/soapService/xsd/
differentNamespace.xsd"/>
  <!-- DOCUMENT requires unique localnames for the operations -->
  <Operation name="irrelevantName1"
    process="x4db:/1/RestExamples/soap/DirectOutput.wrf">
    <InputHandler type="SOAP-Envelope"/>
    <OutputHandler type="SOAP-Envelope"/>
    <Input>
      <Part name="firstPart" element="exampleSpace:SpacedOperation"/>
    </Input>
    <Output>
      <Part name="firstPart" element="exampleSpace:Operation1"/>
    </Output>
  </Operation>
  <Operation name="irrelevantName2"
    process="x4db:/1/RestExamples/soap/DirectOutput.wrf">
    <InputHandler type="SOAP-Envelope"/>
    <OutputHandler type="SOAP-Envelope"/>
    <Input>
      <Part name="firstPart" element="anotherSpace:SpacedOperation"/>
    </Input>
    <Output>
      <Part name="Output" element="anotherSpace:SpacedOperation"/>
    </Output>
  </Operation>
  <Operation name="irrelevantName3"
    process="x4db:/1/RestExamples/soap/DirectOutput.wrf">
    <InputHandler type="SOAP-Envelope"/>
    <OutputHandler type="SOAP-Envelope"/>
    <Input>
      <Part name="UnSpacedOperation" type="xsd:integer"/>
    </Input>
    <Output>
      <Part name="Value" type="xsd:String"/>
    </Output>
  </Operation>
</Service>

```


3.14.3.3 SOAP-Webservice mit externer WSDL-Definition bereitstellen

Wie Sie mit der X4 Suite SOAP-Webservices z.B. im RPC-Stil auf Basis einer bestehenden WSDL-Definition bereitstellen, wird im Folgenden beschrieben.



Voraussetzungen

- Grundlegendes Wissen über den Aufbau von SOAP-Webservices
- Die WSDL-Definition muss gültig und über eine für den Konsumenten gültige URL erreichbar sein, z.B. via HTTP.
- Ein technischer Prozess pro Webservice-Operation, der die Verarbeitung der eingehenden Anfrage übernimmt. Der Prozess wird dabei über seine x4db: /-URL referenziert.
- Ein URL-Part, über den der SOAP-Webservice via HTTP erreichbar sein soll, z.B. über `http://localhost:8080/X4/httpstarter/ReST/MyExternalWSDLWebservice`. Diesen URL-Part haben Sie bereits typischerweise als Konstante innerhalb der Webservice-Konfiguration definiert, siehe [Webservice-Konfiguration grafisch bearbeiten](#).

Um einen technischen Prozess als SOAP-Webservice mit externer WSDL-Definition bereitzustellen, gehen Sie wie folgt vor:

1. Unter der gewünschten URL-Ressource (definiert z.B. durch einen konstanten URL-Part) über das Kontextmenü **Add SOAP Service** einen neuen Service-Eintrag vom Typ *SOAP* anlegen und ggf. (Sicherheits-)Eigenschaften anpassen.
2. SOAP-Version festlegen (entweder *SOAP 1.1*, *SOAP 1.2* oder Unterstützung für beide Standards).
3. WSDL-Modus EXTERNAL wählen und in **WSDL URL** (Attribut `url="..."`) eine gültige URL zur externen WSDL-Definition angeben.

- ✓ Sie können auch eine selbsterzeugte WSDL-Definition und eine XSD aus dem Repository bereitstellen. Stellen Sie die WSDL-Definitionsdatei dazu als File-Service via HTTP bereit, siehe [File-Webservice bereitstellen](#).

Beispiel (Auszug aus XML-Speicherformat)

```
<Constant value="externalWsdL">
  <Service type="File" path="x4db:/1/RestExamples/soap/
externalWsdL.xml"/>
</Constant>
<Constant value="xsd">
  <Remainder name="FILE">
    <Service type="File" path="x4db:/1/RestExamples/soap/xsd/
&lt;FILE>"/>
  </Remainder>
</Constant>
```

Anschließend können Sie die File-Webservice-URL in der Definition des SOAP-Webservices z.B. wie folgt referenzieren:

Beispiel (Auszug aus XML-Speicherformat)

```
<WsdL mode="EXTERNAL" url="http://localhost:8080/X4/httpstarter/Rest/
soapService/externalWsdL"/>
```

4. Auf **Load Configuration from WSDL** klicken, um die WSDL-Definition mit den entsprechenden Webservice-Informationen (SOAP-Version, Service-Präfix, verknüpfte XSD-Schemas und verfügbare Webservice-Operationen) zu laden.
In der Baumstruktur des *X4 Webservice Configuration Editors* wird für jede in der WSDL-Definition definierte Webservice-Operation ein entsprechend benannter Kindknoten angelegt.
5. Für jede Webservice-Operation im entsprechenden Kindknoten die Konfiguration anpassen:
 - Technischen Prozess, der die Webservice-Anfrage über den X4 Server verarbeitet, über dessen x4db:/-URL verknüpfen.
Im *Webservice Configuration Editor* ist dies per Drag & Drop aus dem *Repository Navigator* möglich.
 - Input Handler und Output Handler für den SOAP-Webservice definieren.
 - Bei Bedarf Sicherheitseinstellungen (*Guest Allowed*, *Token Return* etc.) anpassen.
6. Speichern und Webservice-Konfiguration mit **Reload on Server** anwenden.
Der SOAP-Webservice ist ab sofort aktiv und kann über dessen Service-URL via HTTP POST aufgerufen werden.

i Bei einem Aufruf der Service-URL via HTTP GET wird die in **WSDL URL** (Attribut `url`) angegebene URL weitergeleitet (HTTP-Redirect).

Beispiel: Konfiguration des Webservices mit externer WSDL-Definition

```
<Service type="Soap">
  <SoapVersion>
    <SOAP11/>
    <SOAP12/>
  </SoapVersion>
  <Wsdll mode="EXTERNAL" url="http://www.webservice.com/CurrencyConvertor.asmx?
wsdl"/>
  <Namespace prefix="exampleSpace" namespace="http://example.org/namespace"/>
  <Namespace prefix="anotherSpace" namespace="http://example.org/different/
namespace"/>
  <!-- requires unique qualified names for the operations -->
  <Operation name="exampleSpace:SpacedOperation"
    process="x4db:/1/RestExamples/soap/DirectOutput.wrf">
    <InputHandler type="SOAP-Envelope"/>
    <OutputHandler type="SOAP-Envelope"/>
  </Operation>
  <Operation name="anotherSpace:SpacedOperation"
    process="x4db:/1/RestExamples/soap/DirectOutput.wrf">
    <InputHandler type="SOAP-Envelope"/>
    <OutputHandler type="SOAP-Envelope"/>
  </Operation>
  <Operation name="UnSpacedOperation"
    process="x4db:/1/RestExamples/soap/DirectOutput.wrf">
    <InputHandler type="SOAP-Envelope"/>
    <OutputHandler type="SOAP-Envelope"/>
  </Operation>
</Service>
```

3.14.3.4 SOAP-Webservice ohne WSDL-Definition bereitstellen

Wie Sie mit der X4 Suite SOAP-Webservices z.B. im RPC-Stil ohne WSDL-Definition via HTTP-POST bereitstellen, wird im Folgenden beschrieben.

Voraussetzungen:

- Grundlegendes Wissen über den Aufbau von SOAP-Webservices
- Ein Prozess pro Webservice-Operation, der die Verarbeitung der eingehenden Anfrage übernimmt. Der Prozess wird dabei über seine x4db:/-URL referenziert.
- Ein URL-Part, über den der SOAP-Webservice via HTTP erreichbar sein soll, z.B. über `http://localhost:8080/X4/httpstarter/ReST/MyWebservice`. Diesen URL-Part haben Sie bereits typischerweise als Konstante innerhalb der Webservice-Konfiguration definiert, siehe [Webservice-Konfiguration grafisch bearbeiten](#).

Um einen technischen Prozess als SOAP-Webservice ohne WSDL-Definition bereitzustellen, gehen Sie wie folgt vor:

1. Unter der gewünschten URL-Ressource (definiert z.B. durch einen konstanten URL-Part) über das Kontextmenü **Add SOAP Service** einen neuen Service-Eintrag vom Typ *SOAP* anlegen.
2. SOAP-Version festlegen (entweder *SOAP 1.1*, *SOAP 1.2* oder Unterstützung für beide Standards).

3. WSDL-Modus NONE wählen.
4. Für jede Operation folgende Konfiguration hinterlegen:
 - Als Namen den *Qualified Name* des ersten Elements im Body des SOAP Requests für die Operation hinterlegen.
 - Technischen Prozess, der die Webservice-Anfrage über den X4 Server verarbeitet, über dessen x4db: /-URL verknüpfen.
Im *Webservice Configuration Editor* ist dies per Drag & Drop aus dem Repository Navigator möglich.
 - Input Handler und Output Handler für den SOAP-Webservice definieren.
5. Speichern und Webservice-Konfiguration mit **Reload on Server** anwenden.
Der SOAP-Webservice ist ab sofort aktiv und kann über dessen Service-URL ausschließlich via HTTP POST aufgerufen werden. Ein Aufruf der Service-URL via HTTP GET gibt keine WSDL-Definition zurück.

Beispiel: Konfiguration des Webservices ohne WSDL-Definition

```
<Service type="Soap">
  <SoapVersion>
    <SOAP11/>
    <SOAP12/>
  </SoapVersion>
  <WsdL mode="NONE"/>
  <Namespace prefix="exampleSpace" namespace="http://example.org/namespace"/>
  <Namespace prefix="anotherSpace" namespace="http://example.org/different/
namespace"/>
  <!-- requires unique qualified names for the operations -->
  <Operation name="exampleSpace:SpacedOperation"
    process="x4db:/1/RestExamples/soap/DirectOutput.wrf">
    <InputHandler type="SOAP-Envelope"/>
    <OutputHandler type="SOAP-Envelope"/>
  </Operation>
  <Operation name="anotherSpace:SpacedOperation"
    process="x4db:/1/RestExamples/soap/DirectOutput.wrf">
    <InputHandler type="SOAP-Envelope"/>
    <OutputHandler type="SOAP-Envelope"/>
  </Operation>
  <Operation name="UnSpacedOperation"
    process="x4db:/1/RestExamples/soap/DirectOutput.wrf">
    <InputHandler type="SOAP-Envelope"/>
    <OutputHandler type="SOAP-Envelope"/>
  </Operation>
</Service>
```

3.14.3.5 SOAP-Webservices: Input- und Output-XML

Zu jedem SOAP-Webservice lässt sich in der Webservice-Konfiguration ein Input-/Output-Handler (in den Knoten `<InputHandler/>` und `<OutputHandler/>`) definieren. Diese steuern, welche XML-Datenstruktur erwartet bzw. bei der Übergabe an Webservices erzeugt wird.

3.14.3.5.1 Modus SOAP-Envelope: Input-Handler


Wenn als Input-Handler der Modus *SOAP-Envelope* gewählt wurde (XML-Speicherformat: `<InputHandler type="SOAP-Envelope"/>`), so wird das komplette XML des SOAP-Requests inklusive Header-Informationen und Inhalt im Body übergeben, beispielsweise:

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <GetEntry>
      <id>?</id>
    </GetEntry>
  </soapenv:Body>
</soapenv:Envelope>
```

3.14.3.5.2 Modus SOAP-Envelope-Validated: Input-Handler

Wenn als Input-Handler der Modus *SOAP-Envelope-Validated* gewählt wurde, so wird das komplette XML-Dokument des SOAP-Requests inklusive Header-Informationen und Inhalt im Body übergeben. Zusätzlich wird jedoch der Input gegen die verknüpfte WSDL-Definition und die darin enthaltenen XML-Schemata validiert.

Bei Validierungsfehlern wird im Ergebnisdokument ein SOAP Fault (z.B. in einem Element `<SOAP-ENV:Fault>`) zurückgegeben, das die entsprechenden Fehlermeldungen enthält.

 Dieser Modus funktioniert nur, wenn bei der Service-Konfiguration das Feld **WSDL URL** ausgefüllt und die WSDL mit **Load Configuration from WSDL** verfügbar gemacht wurde. Siehe auch [SOAP-Webservice mit externer WSDL-Definition bereitstellen](#).

3.14.3.5.3 Modus SOAP-Envelope: Output-Handler

Analog zum Input-Handler gilt, wenn als Output-Handler der Modus *SOAP-Envelope* gewählt wurde (XML-Speicherformat: `<OutputHandler type="SOAP-Envelope"/>`), so ist der Prozess für die Erstellung des vollständigen SOAP-Dokuments verantwortlich, d.h. der Prozess muss sich selber um die Erstellung von SOAP-Envelope, SOAP-Header und SOAP-Body kümmern.

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <SomeHeader />
  </soap:Header>
  <soap:Body>
    <SomeData />
  </soap:Body>
</soap:Envelope/>
```

3.14.3.5.4 Modus SOAP-Content: Input-Handler

Wenn als Input-Handler der Modus SOAP-Content gewählt wurde (XML-Speicherformat: `<InputHandler type="SOAP-Envelope"/>`), so wird der erste Knoten des Bodys innerhalb des SOAP-Requests übergeben, beispielsweise:

Beispiel: SOAP-Request

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <SomeHeader />
  </soap:Header>
  <soap:Body>
    <GetEntry>
      <id>101</id>
    </GetEntry>
  </soap:Body>
</soap:Envelope/>
```


Beispiel: Prozess-Input

```
<?xml version="1.0" encoding="UTF-8"?>
<GetEntry>
  <id>101</id>
</GetEntry>
```

3.14.3.5.5 Modus SOAP-Content-Validated: Input-Handler

Wenn als Input-Handler der Modus SOAP-Content-Validated gewählt wurde, so wird der erste Knoten des Bodys innerhalb des SOAP-Requests übergeben. Zusätzlich wird jedoch der Input gegen die verknüpfte WSDL-Definition und die darin enthaltenen XML-Schemata validiert.

Bei Validierungsfehlern wird im Ergebnisdokument ein SOAP Fault (z.B. in einem Element `<SOAP-ENV:Fault>`) zurückgegeben, das die entsprechenden Fehlermeldungen enthält.

 Dieser Modus funktioniert nur, wenn bei der Service-Konfiguration das Feld **WSDL URL** ausgefüllt und die WSDL mit **Load Configuration from WSDL** verfügbar gemacht wurde. Siehe auch [SOAP-Webservice mit externer WSDL-Definition bereitstellen](#).

3.14.3.5.6 Modus SOAP-Content: Output-Handler

Analog zum Input-Handler gilt, wenn als Output-Handler der Modus SOAP-Content gewählt wurde (XML-Speicherformat: `<OutputHandler type="SOAP-Content"/>`), so wird das Ergebnis des X4-Prozesses als Inhalt des SOAP-Bodys für die SOAP-Response genommen.

```
<?xml version="1.0" encoding="UTF-8"?>
<GetEntryResponse>
  <Data>
    <customer>
      <customerid>101</customerid>
      <companyname>SoftProject GmbH</companyname>
      <address>Am Erlengraben 3</address>
      <city>Ettlingen</city>
      <postalcode>76275</postalcode>
      <country>Germany</country>
      <phone>+49 (0)7243 | 561 75-0</phone>
    </customer>
  </Data>
</GetEntryResponse>
```

3.14.3.5.7 Transformation des Prozessergebnisses in den Response-Body

Ist das Prozessergebnis `<SomeData/>` und handelt es sich um ein SOAP 1.1 Request, wird folgende Antwort gesendet:

```
<SOAP11:Envelope xmlns:SOAP11="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP11:Body>
    <SomeData/>
  </SOAP11:Body>
</SOAP11:Envelope/>
```

Ist das Prozessergebnis `<SomeData/>` und handelt es sich um ein SOAP 1.2 Request, wird folgende Antwort gesendet:

```
<SOAP12:Envelope xmlns:SOAP12="http://www.w3.org/2003/05/soap-envelope">
  <SOAP12:Body>
    <SomeData/>
  </SOAP12:Body>
</SOAP12:Envelope/>
```

3.14.4 File-Webservice bereitstellen

Zusätzlich zur Möglichkeit, X4-Prozesse als ReSTful-Services via HTTP(S) bereitzustellen, lassen sich auch statische Datei-Ressourcen aus dem Repository als Service via HTTP GET bereitstellen.

Voraussetzungen:

- Eine Datei im Repository, die Sie als File-Service via HTTP zum Abruf bereitstellen möchten. Die Datei wird dabei über seine x4db: /-URL referenziert.
- Ein URL-Part, über den die Datei via HTTP(S) erreichbar sein soll, z.B. über `http://localhost:8080/X4/httpstarter/ReST/MyFile`. Diesen URL-Part haben Sie bereits (als Konstante, Variable oder Remainder) innerhalb der Webservice-Konfiguration definiert, siehe [Webservice-Konfiguration grafisch bearbeiten](#).

3.14.4.1 Datei-Ressource via HTTP bereitstellen

1. Unter der gewünschten URL-Ressource (definiert z.B. durch einen konstanten oder variablen URL-Part) über das Kontextmenü **Add File Service** einen neuen Service-Eintrag vom Typ *File* anlegen.
2. Unter **Path** den `x4db:/`-Pfad zur gewünschten Datei im Repository hinterlegen, deren Inhalt als Datei-Ressource via HTTP unverändert bereitgestellt werden soll.

✓ Im *X4 Webservice Configuration Editor* können Sie eine Datei aus dem Repository auch per Drag & Drop auf das Feld **Path** zuweisen.

3. Bei Bedarf die Sicherheitseinstellungen (*Guest Allowed*, *Token Return* etc.) anpassen.
4. Speichern und mit **Reload on Server** die Webservice-Konfiguration anwenden.
Die Änderungen sind sofort aktiv und Ihr File-Webservice wurde bereitgestellt.

3.14.4.2 File-Webservices mit Variablen nutzen

Die Pfadangabe zur Repository-Datei in Attribut `path` können Sie dynamisch aufbauen, indem Sie den File-Service in einem Variablen-Block definieren, etwa für wechselnde Dateinamen oder IDs innerhalb eines variablen URL-Parts. Die Zuordnung erfolgt anhand des Variablennamens, z.B. `<myVar>`.

Beispiel: Definition eines FILE-Services als Konstante (Auszug aus `restconfig.xml`)

```
<Variable name="myVar">
  <Service type="File" path="x4db:/1/Project/Folder/&lt;myVar"/>
</Variable>
```

Erläuterungen:

- Element `<Variable/>` definiert in Attribut `name` den Namen einer Variable, z.B. `myVar`.
- In Attribut `path` wird diese Variable an beliebiger Stelle innerhalb der `x4db:/`-URL als Platzhalter referenziert, auch Mehrfachnennungen des Platzhalters sind möglich.
- Der Platzhalter beginnt immer mit einer spitzen öffnenden Klammer `<(<` im XML-Speicherformat), gefolgt vom Variablennamen und einer spitzen schließenden Klammer `>`.

Beispiel: Variablen-Platzhalter an beliebiger Stelle platziert (Auszug aus `restconfig.xml`)

```
<Constant value="varied_1">
  <Variable name="VAR">
    <Service type="File" path="x4db:/1/RestExamples/file/varied_1/&lt;VAR"/>
  </Variable>
</Constant>
<Constant value="varied_2">
  <Variable name="VAR">
    <Service type="File" path="x4db:/1/RestExamples/file/varied_1/&lt;VAR>.png"/>
  </Variable>
</Constant>
```


Beispiel: Kombinierte Variablen (Auszug aus XML-Speicherformat)

```
<Constant value="varied_3">
  <Variable name="TYPE">
    <Variable name="PART">
      <Service type="File"
        path="x4db:/1/RestExamples/file/deep/&lt;TYPE>/&lt;PART>.&lt;TYPE>"/>
    </Variable>
  </Variable>
</Constant>
<Constant value="varied_4">
  <Variable name="VAR">
    <Constant value="image">
      <Service type="File"
        path="x4db:/1/RestExamples/file/varied_1/&lt;VAR>.png"/>
    </Constant>
  </Variable>
</Constant>
```

3.14.4.2.1 File-Webservices mit Remainder nutzen

Wenn Sie einen File-Webservice mit beliebigen URL-Parts definieren möchten, können Sie auf das *Remainder*-Konstrukt zurückgreifen. Als tolerantere Variante einer Variablen bietet dieses die Möglichkeit, beliebig tiefe URL-Parts abzufangen. Analog zu Variablen definieren Sie hier einen Remainder mit einem Namen und referenzieren diesen in Attribut path.

Beispiel: Definition eines FILE-Services mit Remainder (Auszug aus XML-Speicherformat)

```
<Remainder name="myRemainderVar">
  <Service type="File"
    path="x4db:/1/Project/Folder/&lt;myRemainderVar>"/>
</Remainder>
```

Erläuterungen:

- Element `<Remainder/>` definiert in Attribut `name` den Namen einer Remainder-Variable, z.B. `myRemainderVar`.
- In Attribut `path` wird diese Variable an beliebiger Stelle innerhalb der `x4db:/`-URL als Platzhalter referenziert. Auch Mehrfachnennungen des Platzhalters sind möglich.
- Der Platzhalter beginnt immer mit einer spitzen öffnenden Klammer `<` (bzw. `<` im XML-Speicherformat), gefolgt vom Variablennamen und einer spitzen schließenden Klammer `>`.

Beispiel: Weitere Beispiele für Remainder (Auszug aus XML-Speicherformat)

```
<Constant value="deep_1">
  <Remainder name="VAR">
    <Service type="File"
      path="x4db:/1/RestExamples/file/deep/&lt;VAR>"/>
    </Remainder>
  </Constant>
<Constant value="deep_2">
  <Remainder name="VAR">
    <Service type="File"
      path="x4db:/1/RestExamples/file/deep/&lt;VAR>.png"/>
    </Remainder>
  </Constant>
```

3.14.4.2.2 Caching von bereitgestellten Datei-Ressourcen

Jede Datei-Ressource, die über einen File-Webservice per HTTP GET bereitgestellt wird, ist 10 Stunden gültig. Der Client (z.B. ein Web-Browser) ruft die Ressource also erst nach Ablauf dieser Frist erneut auf.

Um für Entwicklungs- und Testzwecke die aktuelle Ressource zu laden, setzen Sie im Client den Cache manuell zurück oder deaktivieren diesen.

3.14.5 WADL-Services bereitstellen

Mit einem WADL-Service ist es beispielsweise einfach möglich, für ReSTful-Webservices eine standardisierte Beschreibung in der *Web Application Description Language* (WADL) automatisch zu erzeugen, um diese in einem OpenAPI-konformen Service Repository zu hinterlegen. Das dynamisch erzeugte WADL-Dokument enthält eine Auflistung der Webservice-Methoden bzw. Parameter.

Voraussetzungen:


- Mindestens ein ReSTful-Webservice wurde via HTTP(S) bereitgestellt.
- Für den ReSTful-Service wurde im der Webservice-Definition im aktuell gültigen Knotenkontext die Eigenschaft `WADL Allowed` (analog zu `Guest Allowed`) aktiviert.

Um im *Webservice Configuration Editor* einen WADL-Service für einen ReSTful-Webservice bereitzustellen, gehen Sie wie folgt vor:

1. *Webservice Configuration Editor* über das Menü **Tools > Manage Webservices** öffnen.
2. In der Baumstruktur bei Bedarf einen neuen Knoten (entspricht einem URL-Part des Services) erstellen, siehe [Webservice-Konfiguration grafisch bearbeiten](#).
3. Für den gewünschten Knoten (vom Typ `Constant`, `Variable` oder `Remainder`) über **Add WADL Service** einen neuen Service vom Typ `HTTP WADL` erstellen und ggf. (Sicherheits-)Eigenschaften anpassen.
4. Unter **Service Configuration** die Service-Konfiguration hinterlegen:
 - **WADL Path** (`wadlPath`): Hier den Pfad des Knotens angeben, ab welchem das WADL-Dokument erzeugt werden soll – falls für einen bestimmten ReSTful Service eine WADL


erzeugt werden soll.

Beispiel: /X4WebResource

 Wenn Sie dieses Feld leer lassen, wird ein WADL-Dokument für alle ReSTful Webservices in der *X4 Webservices*-Konfiguration erzeugt.


- **WADL Base** (wadlBase): Bei Bedarf die Basis-Referenz (Hostname) des ReSTful Webservices überschreiben.

Beispiel: `http://localhost:9876/`

 Wenn Sie dieses Feld leer lassen, werden im vom WADL-Service erzeugten WADL-Dokument die Werte aus dem HTTP-Request ermittelt.

WADL Base Path (wadlBasePath): Bei Bedarf den Basis-URL-Pfad des ReSTful Webservices überschreiben.

Beispiel: `/application/api`.

 Wenn Sie dieses Feld leer lassen, werden im vom WADL-Service erzeugten WADL-Dokument die Werte aus dem HTTP-Request ermittelt.


5. Speichern und mit **Reload on Server** die Webservice-Konfiguration anwenden. Die Änderungen sind sofort aktiv und Ihr WADL-Service wurde bereitgestellt.

3.15 Prozesse debuggen und ausführen


Erfahren Sie hier, wie Sie technische Prozesse und Geschäftsprozesse auf Fehler überprüfen (debuggen) und auf dem X4 Server ausführen.


3.15.1 Prozess debuggen


Während der Prozess-Entwicklung können Sie direkt im Process Editor Ihre technischen Prozesse (und unter bestimmten Voraussetzung auch Geschäftsprozesse) auf syntaktische Fehler der einzelnen Prozessbausteine überprüfen.

 Voraussetzungen:

- Mit allen Prozessbausteine ist falls erforderlich eine Repository-Datei verknüpft.
- Bei allen Prozessbausteine sind sämtliche Pflicht-Parameter und Eigenschaften vollständig parametrisiert.
- Für Geschäftsprozesse: Der Geschäftsprozess wurde nicht im Free-flow-Modus angelegt.

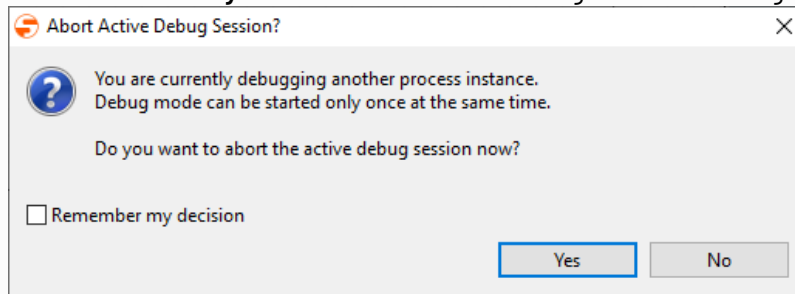
1. Falls gewünscht, Debug-Haltepunkte (Breakpoints) setzen, um bei kritischen Prozess-Schritten das Debugging automatisch anzuhalten:
 - Im Prozessdiagramm den gewünschten Prozessbaustein anklicken.
 - In der Symbolleiste  **Toggle Breakpoint** klicken.

2.  **Start Debugging** klicken, um den Debug-Modus zu starten.


 Beachten Sie:

- Wenn die Konfiguration unvollständig ist (meist eine fehlende Angabe in Eigenschaft `Operation`), kommt sofort eine Fehlermeldung und der betroffene Prozessbaustein erhält eine Warnung.
- Wenn aktuell ein anderer Debug-Vorgang ausgeführt wird, müssen Sie diesen zunächst abbrechen.

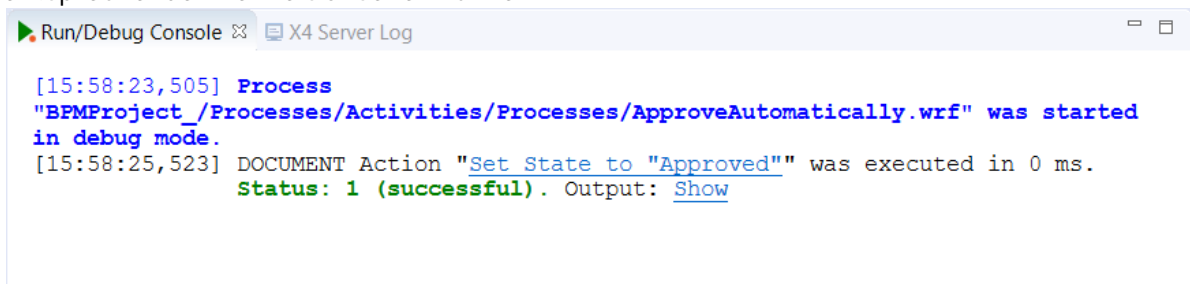
Im folgenden Dialogfenster können Sie das gewünschte Verhalten und optional mit **Remember my decision** in den Einstellungen des X4 Designers speichern.




Tipp: Mit der F4-Taste können Sie den Debugging-Modus neu starten.

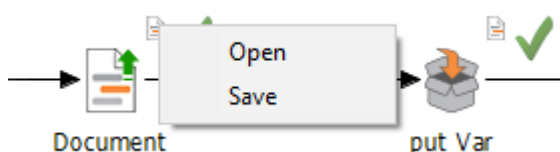
3.  **Step Into** klicken, um den nächsten Einzelschritt auszuführen und anzuzeigen (auch in Subprozesse springen).

Zu jedem ausgeführten Prozess-Schritt wird ein Eintrag in der **Run/Debug Console**-Sicht angezeigt. Fehlermeldungen werden rot hervorgehoben, erfolgreiche Statusmeldungen sind grün. Außerdem wird jeweils eine temporäre Ausgabe-Datei angelegt, die Sie über den entsprechenden Verweis ansehen können.






Zusätzlich wird die jeweilige Ausgabedatei der entsprechenden Prozesskomponente im Prozessmodell angehängt. Über das Kontextmenü lässt sich die Ausgabedatei zudem abspeichern.

 Enthält das Ergebnis Inhalte, die größer als 1 MB sind, wird eine Meldung ausgegeben und das Ergebnis lässt sich über den Dialog herunterladen.



- ✓ Die Output-Sicht und die Quick-Watch-Sicht bieten zusätzliche Möglichkeit beim Debuggen:
 - **Output:** Die Ergebnisse jedes einzelnen Schrittes lassen sich während des Debuggens einsehen; Große Dateien können über einen Download-Button heruntergeladen werden.
 - **Quick:** Platzhalter, dynamischen Parameter und Kontextvariablen lassen sich während des Debuggens betrachten; Große Dateien können über einen Download-Button heruntergeladen werden.



Beide Sichten lassen sich über das Menü **Debug > Output** bzw. **Quick Watch** oder über die jeweiligen Icons  (Output) und  (Quick Watch) in der Symbolleiste aufrufen.

4. Mit  (**Terminate**) können Sie den Debug-Modus bei Bedarf manuell beenden.
Wenn ein technischer Fehler auftritt, dann wird der Debug-Modus automatisch abgebrochen.

3.15.2 Prozess ausführen

Sie können technische Prozesse oder Geschäftsprozesse direkt aus dem X4 Designer heraus starten. Dabei wird der technische Prozess auf dem X4 Server ausgeführt und es werden keine Debug-Informationen im Server-Protokoll angezeigt.

Um technische Prozesse oder Geschäftsprozesse auszuführen, gibt es mehrere Möglichkeiten:

Geöffneten Prozess starten	In der Symbolleiste auf  Run Without Debugging klicken, um den Prozess zu starten.
Im Repository Navigator	Den Prozess rechtsklicken, im Kontextmenü  Run Without Debugging wählen.
Als Subprozess	Dieser wird automatisch vom übergeordneten technischen Prozess aufgerufen.
Zeitgesteuert ausführen	Mithilfe des Schedule -Eigenschaft des Prozesses zu festgelegten Zeiten starten

3.16 COBOL Copybook Converter Wizard

Mit dem *COBOL Copybook Converter Wizard* können Sie eine Textdatei im *COBOL Copybook*-Format in ein XML-Dokument konvertieren und im selben Schritt eine XSLT-Transformation der XML-Daten durchführen. Wenn alle Namen denselben Präfix besitzen, können Sie außerdem im Ergebnis-XML-Dokument alle Namen mit demselben Präfix reduzieren.

- ❗ Um den *COBOL Copybook Converter Wizard* aufzurufen, im Menü **Project > Cobol Copybook Converter...** wählen.

Cobol Copybook Converter Wizard

Welcome to Cobol Copybook Converter Wizard

Input file (Cobol Copybook)

File name: **Browse...**

Preamble:

Output file (XML)

☐ Save output file

Choose...

Options

☐ Use XSLT to transform XML document result

Browse...

☐ Reduce the names (if all the names have the same prefixes)

Finish **Cancel**

3.16.1 Voraussetzungen

Damit der Wizard korrekt ausgeführt werden kann, muss der Ordner <X4>\X4DB\999\CCB2XML auf dem *X4 Server* vollständig vorhanden sein.

3.16.2 Funktionen im COBOL Copybook Wizard

File name	Pfad und Dateiname der <i>COBOL Copybook</i> -Datei im Repository
Preamble	Präambel der <i>COBOL Copybook</i> -Datei (Standard ist 6)
Display result of conversion	Vorschau des erzeugten XML-Dokuments im internen Browser anzeigen
Save output file	Ergebnis in einer Datei speichern, deren Repository-Pfad Sie über Browse wählen können
Use XSLT to transform ...	Eine XSL-Transformation vor dem Speichern auf das erzeugte XML-Dokument anwenden
Reduce the names ...	Alle Namen mit demselben Präfix reduzieren

4 Prozess-Dokumentation in der X4 Suite

Mit der X4 Suite können Sie für jeden Prozess im Repository zu jedem Zeitpunkt im Prozesslebenszyklus eine Dokumentation im HTML-Format erzeugen. Diese Prozess-Dokumentation enthält zu jedem Prozessbaustein sämtliche Parameter und stellt das Prozessdiagramm als interaktive Grafik dar, siehe Abbildung.

ACHTUNG: Möglicherweise sind vertrauliche Daten einsehbar!

- Das Erzeugen der Prozess-Dokumentation ist standardmäßig über den Benutzer **admin** mit dem Passwort **demo** abgesichert.
- Stellen Sie sicher, dass die Prozess-Dokumentation nur von Befugten eingesehen werden kann.
- Setzen Sie alle Adapter-Parameter, die vertrauliche Daten enthalten, immer dynamisch per XSL-Mapping, z. B. über die Processing Instruction `DynamicParameter`, siehe [Parameter dynamisch setzen](#). So werden vertrauliche Daten erst zur Laufzeit des Prozesses im Adapter gesetzt und die entsprechenden Parameter müssen nicht im Adapter angegeben werden.

TECHNICAL PROCESS SendRequests.wrf

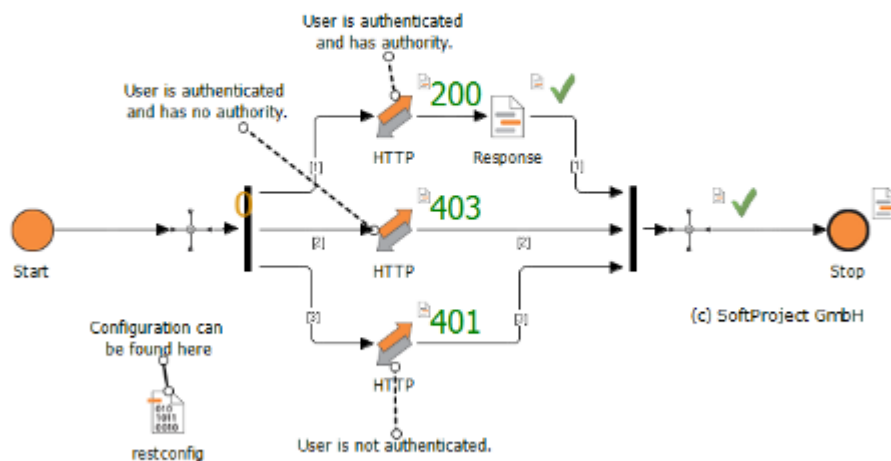
Technical Process Definition

Comment:

Attribute	Value(s)
Process Path	1/X4Experience/ESB/3-Reference/3-Features/SecuredWebservice/Processes
Process Name	SendRequests.wrf
Process Title	Call Secured HTTP-Service
Process Type	ESB
Version	1488969595008
Project Name	
Author	
Created	8. 3. 2017 10:56:09
Modified	8. 3. 2017 11:39:55
Modified by	
Can Stop	true
Schedule	
In Service	true
Instance Limit	0
Stop on Error	true

Technical Process Diagram

Call Secured HTTP-Service



Prozess-Dokumentation erzeugen

Möglichkeiten, um eine Prozess-Dokumentation zu erzeugen, sind:


Im Repository Navigator	Über die Kontextmenü-Option Show Process Documentation
Im Prozess-Designer	Zu einem im Prozessdiagramm markierten Subprozess über die Kontextmenü-Option Show Process Documentation

Im X4 Control Center

In der Monitoring-Oberfläche des X4 Control Centers für einen ausgewählten Prozess mit Klick auf **Documentation**

5 Mit Adapter-Projekten arbeiten

Innerhalb von Adapter-Projekten lassen sich neue Adapter für die X4 Suite mit Hilfe der verschiedenen Werkzeuge der X4 Suite erstellen, siehe Eigene Adapter mit der X4 Suite entwickeln.

 Um diese Funktion der X4 Suite zu nutzen, wird eine ADK-Lizenz vorausgesetzt.

6 Übersicht der X4 Adapter

6.1 Über X4 Adapter

Die X4 Suite verfügt über eine umfangreiche Sammlung an Standard-Adaptoren zur Integration von Anwendungen und Daten in beliebige Unternehmensprozesse. Mit ihnen verfolgt die X4 Suite einen modularen Ansatz. Dabei werden Adapter als Bausteine in technische Prozesse integriert und stellen Datenanbindungen her, beispielsweise mit Oracle-Datenbanken, übertragen Daten über verschiedene Protokolle wie HTTP und JMS, oder transformieren und konvertieren verschiedene Daten und Datenformate.

Dieser Lösungsansatz ermöglicht es, heterogene IT-Landschaften in eine Service-orientierte Architektur zu überführen.

Eigenschaften der X4 Adapter:


- Plug-and-play-Fähigkeit
- Stetig wachsende Anzahl an Adaptern, derzeit mehr als 200
- Adapter für alle gängigen Standardprotokolle, Formate und Systeme
- Einfache Erstellung eigener Adapter durch das X4 Adapter Development Kit (ADK)
- Einfache Nutzung im X4 Designer per Drag&Drop
- Konfigurieren statt programmieren

6.2 Big Data

6.2.1 Amazon S3 DynamoDB Connector

Amazon S3 DynamoDB Connector: Amazon S3 DynamoDB ist ein vollständig verwalteter NoSQL-Datenbankdienst, der eine schnelle und vorhersehbare Leistung bei nahtloser Skalierbarkeit bietet. Der Adapter speichert Daten im Amazon S3 DynamoDB Storage und ermöglicht die Verwendung der X4 Suite, um Daten in Cloud-Umgebungen zu persistieren.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • CreateTable: Erstellt eine Tabelle mit Tabellen-Name und Primärschlüssel Erforderliche Parameter: tableName, partitionKey, partitionType. Für sortierbare Schlüssel zusätzlich sortKey, sortType. Für AutoScaling zusätzlich scalingMinCapacity, scalingMaxCapacity, scalingTargetValue, scalingInCooldown, scalingOutCooldown, scalableType, arnRole • UpdateTable: Aktualisiert vorhandene Tabellen und ändert die Read/Write Capacity Units. Erforderliche Parameter: tableName, partitionKey, readCapacityUnits, writeCapacityUnits • ListTable: Listet alle Tabellen auf. • EnableAutoScaling: Aktiviert AutoScaling für eine bestehende Tabelle mit dem entsprechenden scalableType (Read/Write). Erforderliche Parameter: tableName, scalingMinCapacity, scalingMaxCapacity, scalingTargetValue, scalingInCooldown, scalingOutCooldown, scalableType, arnRole • DisableAutoScaling: Deaktiviert AutoScaling für eine bestehende Tabelle mit dem entsprechenden scalableType (Read/Write). Erforderliche Parameter: tableName, scalableType • PutItem: Empfängt ein XML-Input mit der Struktur des zu platzierenden Elements Erforderliche Parameter: tableName • BatchPutMultipleItems: Fügt mehrere Elemente in einem einzigen Aufruf ein. Erforderliche Parameter: tableName <div data-bbox="579 1686 1430 1865" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> Erforderliche Dateien</p> <p>Für diese Operation wird eine Input-XML-Datei benötigt, die die Elemente enthält.</p> </div> <ul style="list-style-type: none"> • GetItem: Ruft das spezifische Element ab. Erforderliche Parameter: tableName, partitionKey, partitionType, partitionValue (muss mit dem Element, das abgerufen werden soll übereinstimmen). Bei sortierbare Schlüssel zusätzlich sortKey, sortType, sortKeyValue.
-----------	--

- **BatchGetMultipleItems:** Ruft mehrere Elemente abhängig von den Werten von `partitionKey`, die in einer XML-Datei spezifiziert sind. Alle Elemente werden mit allen Elementattributen abgeholt.
Erforderliche Parameter: `tableName`, `partitionKey`, `partitionType`, `partitionValue` (muss mit dem Element, das abgerufen werden soll übereinstimmen). Bei sortierbare Schlüssel zusätzlich `sortKey`, `sortType`, `sortKeyValue`.

- **Delete:** Löscht Element oder Tabelle gemäß den konfigurierten Parametern.

i Wenn `tableName` und die **Primärschlüssel**-Elemente angegeben sind, dann löscht die Operation `Delete` das spezifische Element. Wenn nur `tableName` angegeben ist, dann löscht die Operation `Delete` die Tabelle.

- **BatchDeleteMultipleItems:** Löscht mehrere Elemente in einem einzigen Aufruf.

Erforderliche Parameter: `tableName`

i Erforderliche Dateien

Für diese Operation wird eine Input-XML-Datei benötigt, die die Elemente enthält.

- **Query:** Ruft Teilmenge von Elementen innerhalb einer Tabelle ab. Für die verschiedenen Attribute werden Bedingungen angegeben. Weitere Informationen zur Input-Datei, die die Query aufruft, siehe Input-Abschnitt.



Erforderliche Parameter: `tableName`, `partitionKey`

i `partitionKey` ist obligatorisch und der Komparator ist immer `"="`. Wenn `sortKey` vorhanden ist, ist der Komparator zwischen `partitionKey` und `sortKey` `"AND"`.

Erlaubte Operatoren: `=`, `>`, `<`, `>=`, `<=`, `!=`, `BETWEEN`, `begins_with`

Parameter

Adapter	<p>Adapter-Hauptklasse (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.amazondynamodbdb.AmazonDynamoDBAdapter</code>: Hauptklasse (Standard)</p>
keyID	<p>Zugangsschlüssel-ID für Anfragen an AWS (Amazon Web Services)</p> <p>i Die Zugangsdaten werden über Parameter an den Adapter gesendet. Dabei gibt es verschiedene Konfigurationsmöglichkeiten, siehe Zugangsdaten einrichten.</p> <p>Mögliche Werte: Beliebige Zeichenkette mit der Zugangsschlüssel-ID, z. B. <code>AKIAIOSFODNN7EXAMPLE</code></p>

secretKey	<p>Geheimer Zugangsschlüssel für Anfragen an AWS</p> <p> Die Zugangsdaten werden über Parameter an den Adapter gesendet. Dabei gibt es verschiedene Konfigurationsmöglichkeiten, siehe Zugangsdaten einrichten.</p> <p>Mögliche Werte: Beliebige Zeichenkette mit der geheimen Zugangsschlüssel, z. B. <code>wJa1rXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY</code></p>
region	<p>Region der Domains</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Namen der Region, z. B. <code>us-east-2</code></p> <p> Buckets, die in einer bestimmten Region erstellt wurden, können nicht von einer anderen Region aus aufgerufen werden, siehe Regionen und Endpoints.</p>
tableName	<p>Tabellen-Name</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Tabellen-Namen</p> <p>Folgende Namensregeln gelten für Tabellen-Namen:</p> <ul style="list-style-type: none"> • Alle Namen müssen UTF-8-kodiert sein • Die Groß- und Kleinschreibung muss beachtet werden • Tabellen-Namen und Index-Namen müssen zwischen 3 und 255 Zeichen lang sein und dürfen nur folgende Zeichen enthalten: <ul style="list-style-type: none"> • a-z • A-Z • 0-9 • _ (Unterstrich) • - (Bindestrich) • . (Punkt) • Attribut-Namen müssen zwischen 1 und 255 Zeichen lang sein
partitionKey, partitionType und partitionKeyValue	<p>Elemente des Tabellen-Primärschlüssels</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • PartitionKey: Name des Partitionsschlüssels • PartitionType: Partitionstyp <ul style="list-style-type: none"> • S: Das Attribut ist eine Zeichenkette • N: Das Attribut ist eine Zahl • B: Das Attribut ist vom Typ Binär • PartitionKeyValue: Wert des Elements; Wird für die Operation <code>GetItem</code> benötigt
sortKey, sortType and sortKeyValue	<p>Elemente des Tabellen-Primärschlüssels</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • SortKey: Name des Sortierschlüssels • SortType: Sortiertyp <ul style="list-style-type: none"> • S: Das Attribut ist eine Zeichenkette • N: Das Attribut ist eine Zahl • B: Das Attribut ist vom Typ Binär • SortKeyValue: Wert des Elements; Wird für die Operation <code>GetItem</code> benötigt

readCapacityUnits, writeCapacityUnits	Read/Write Capacity Units Mögliche Werte: <ul style="list-style-type: none"> • Integer Capacity Units
scalableType	Definiert den Skalierbarkeitstyp für AutoScaling. Mögliche Werte: <ul style="list-style-type: none"> • ReadCapacity • WriteCapacity
autoScaling	Legt fest, dass die Tabelle mithilfe von AutoScaling erstellt wird. Im Fall einer Überprüfung müssen alle Scaling-Parameter angegeben werden. Mögliche Werte: <ul style="list-style-type: none"> • <i>true</i> • <i>false</i>
scalingMinCapacity, scalingMaxCapacity, scalingTargetValue, scalingInCooldown, scalingOutCooldown	Definiert die Scaling-Parameter für AutoScaling. Mögliche Werte: <ul style="list-style-type: none"> • Weitere Informationen siehe AutoScaling.
arnRole	ARN der Rolle, die autorisiert ist, AutoScaling für die Tabellenressource anzuwenden. Mögliche Werte: <ul style="list-style-type: none"> • Weitere Informationen siehe Amazon Resource Names (ARN)
consistentRead	DynamoDB verwendet Eventually Consistent Reads, sofern nichts anderes angegeben ist. Leseoperationen wie GetItem und Query liefern einen ConsistentRead-Parameter. Wenn dieser Parameter auf true gesetzt ist, verwendet DynamoDB während des Betriebs Strongly Consistent Reads.

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt
---	--

0	Operation Delete: <ul style="list-style-type: none"> • Das Element existiert nicht • Die Tabelle existiert nicht
-1	Bei der Ausführung des Adapters kam es zu einem Fehler


Input

Der Adapter erwartet eine vordefinierte XML-Struktur als Input. Die Struktur ist abhängig von der verwendeten Operation.

- PutItem-Operation:

Erwartete Struktur für die Operation PutItem


```
<?xml version="1.0" encoding="UTF-8"
<RootElement>
  <Item>
    <Attribute name="..Name of PartionKey.." type="..Type of Partition.."
value="..Value of PartitionKey.."/>
    <Attribute name="..Name of SortKey.." type="..Type of Sort.." value="..
Value of SortKey.."/>
    <Attribute name="..Attribute Name.." type="..Attribute Type.." value="..
.Attribute Value.."/>
  </Item>
</RootElement>
```

 Für die Operation PutItem gibt es keine Einschränkungen für Elementnamen. Die Attribute name, value und type sind jedoch erforderlich.

- BatchGetMultipleItems-Operation:

Erwartete Struktur für die Operation BatchGetMultipleItems

```
<?xml version="1.0" encoding="UTF-8" ?>
<PrimaryKeyValues>
  <PrimaryKey partitionKey="2" SortKey="Carlos" />
  <PrimaryKey partitionKey="5" SortKey="Luis" />
</PrimaryKeyValues>
```

 Primary Keys können für verschiedene Objekte spezifiziert werden. Wenn das Objekt mit einer Tabelle übereinstimmt, in der nur partitionKey definiert ist, darf nur partitionKey angegeben werden, andernfalls partitionKey und SortKey.

- BatchPutMultipleItems-Operation:

Erwartete Struktur für die Operation BatchPutMultipleItems

```
<?xml version="1.0" encoding="UTF-8" ?>
<RootElement>
  <Item>
    <Attribute name="ID" type="N" value="5" />
    <Attribute name="Name" type="S" value="Luis" />
    <Attribute name="Phone" type="N" value="0034666777888" />
    <Attribute name="Bonus" type="BOOL" value="true" />
    <Attribute name="Colours" type="LIST">
      <Attribute type="S" value="Red" />
      <Attribute type="S" value="Green" />
      <Attribute type="S" value="Blue" />
    <Attribute name="Cars" type="MAP">
      <Attribute5 name="Phone" type="N" value="637559681" />
      <Attribute6 name="Bonus" type="BOOL" value="true" />
      <Attribute7 name="Profession" type="S" value="Programmer" />
      <Attribute7 name="DocumentCode" type="B" value="qwrqfgdas122345
5sdfffr" />
    </Attribute>
  </Attribute>
  <Attribute name="Cars" type="MAP">
    <Attribute5 name="Phone" type="N" value="637559681" />
    <Attribute6 name="Bonus" type="BOOL" value="true" />
    <Attribute7 name="Profession" type="S" value="Programmer" />
    <Attribute7 name="DocumentCode" type="B" value="qwrqfgdas1223455sdfffr" />
  </Attribute>
</Item>
<Item>
  <Attribute1 name="ID" type="N" value="5" />
  <Attribute3 name="Name" type="S" value="Luis" />
  <Attribute4 name="Surname" type="S" value="Delgado" />
  <Attribute2 name="Address" type="S" value="Pirandello" />
  <Attribute5 name="Phone" type="N" value="637559681" />
  <Attribute6 name="Bonus" type="BOOL" value="true" />
  <Attribute7 name="Profession" type="S" value="Programmer" />
  <Attribute7 name="DocumentCode" type="B" value="qwrqfgdas1223455sdfffr" />
</Item>
</RootElement>
```

- ❗ Es können mehrere Items spezifiziert werden. In den Items können Attribute vom Typ Boolean (BOOL), String (S), Number (N), Map (MAP), List (LIST) und Binary (B) spezifiziert werden. Der Output dieser Operation ist dieselbe XML-Datei. Das kann für weitere Operationen nützlich sein.

- BatchDeleteMultipleItems-Operation:

Erwartete Struktur für die Operation BatchDeleteMultipleItems

```
<?xml version="1.0" encoding="UTF-8" ?>
<RootElement>
  <Item>
    <Attribute1 name="ID" type="N" value="7" />
    <Attribute7 name="Name" type="S" value="Luis" />
  </Item>
  <Item>
    <Attribute1 name="ID" type="N" value="6" />
    <Attribute7 name="Name" type="S" value="Luis" />
  </Item>
</RootElement>
```

- ❗ In diesem Input werden die Primärschlüssel der Elemente angegeben, die gelöscht werden sollen. Wenn der Primärschlüssel nur aus dem `partitionKey` besteht, wird nur der `partitionKey` benötigt. Wenn der Primärschlüssel aus `partitionKey` und `SortKey` besteht, werden beide Schlüssel benötigt, damit das zu löschende Element eindeutig ist. Der Output dieser Operation ist auch der Input.

- Query-Operation:

Erwartete Struktur für die Operation Query

```
<?xml version="1.0" encoding="UTF-8" ?>
<Query>
  <KeyCondition>ID=:param1</KeyCondition>
  <OtherFilters>Colors[3].Profession=:param3 or DocumentCode=:param2</OtherFilters>
  <QueryParameters>
    <Parameter type="N" alias=":param1" value="5" />
    <Parameter type="B" alias=":param2" value="cXdycWZnZGFzMtIyMzQ1NXNkZmZm
cg==" />
    <Parameter type="S" alias=":param3" value="Programmer" />
  </QueryParameters>
</Query>
```

 Das Elternelement Query enthält drei Elemente:

- KeyCondition: enthält nur den spezifischen partitionKey und falls vorhanden SortKey, die folgende Bedingungen erfüllen:
 - Der Komparator zwischen partitionKey und SortKey ist immer "AND".
 - partitionKey ist obligatorisch und wird immer mit "=" verwendet.
 - Wenn SortKey verwendet wird, können folgende Operatoren verwendet werden: =, >, <, >=, <=, !=, BETWEEN, begins_with
 - partitionKey und SortKey dürfen nicht mit für [AWS DynamoDB reservierten Wörtern](#) übereinstimmen.
- OtherFilters: spezifiziert Bedingungen für die restlichen Attribute, die nicht Teil des Primärschlüssels sind. Die betroffenen Attribute dürfen nicht mit für [AWS Dynamo DB reservierten Wörtern](#) übereinstimmen.
- QueryParameters: spezifiziert das Mapping zwischen den verwendeten Parametern und Ihren entsprechenden Werten. Der Typ jedes Parameters muss spezifiziert werden.

Output

Je nach Operation, gibt der Adapter Daten in verschiedenen XML-Strukturen aus:

- GetItem-Operation

Beispiel-Output für die Operation GetItem

```
<?xml version="1.0" encoding="UTF-8"
<Result>
  <Item partitionKey="ID" partitionKeyValue="2" SortKey="Name" SortKeyValue="
Carlos">
    <Attribute name="Address" type="S" value="Pirandello"/>
    <Attribute name="Phone" type="N" value="637559681"/>
    <Attribute name="Bonus" type="BOOL" value="true"/>
    <Attribute name="DocumentCode" type="B" value="qwrqfgdas1223455sdfffr"/
  >
    <Attribute name="Surname" type="S" value="Delgado"/>
    <Attribute name="Profession" type="S" value="Programmer"/>
  </Item>
</Result>
```

- BatchGetMultipleItems-Operation:

Beispiel-Output für die Operation BatchGetMultipleItems

```

<?xml version="1.0" encoding="UTF-8" ?>
<Result>
  <Item partitionKey="ID" partitionKeyValue="5" SortKey="Name" SortKeyValue="
Luis">
    <Attribute type="MAP" name="Cars">
      <Attribute type="N" name="Phone" value="637559681" />
      <Attribute type="BOOL" name="Bonus" value="true" />
      <Attribute type="B" name="DocumentCode" value="qwrqfgdas1223455sdff
fr" />
    <Attribute type="S" name="Profession" value="Programmer" />
  </Attribute>
    <Attribute type="N" name="Phone" value="34666777888" />
    <Attribute type="LIST" name="Colours">
      <Attribute type="S" value="Red" />
      <Attribute type="S" value="Green" />
    <Attribute type="S" value="Blue" />
    <Attribute type="MAP" name="Colours">
      <Attribute type="N" name="Phone" value="637559681" />
      <Attribute type="BOOL" name="Bonus" value="true" />
      <Attribute type="B" name="DocumentCode" value="qwrqfgdas1223455
sdfffr" />
    <Attribute type="S" name="Profession" value="Programmer" />
  </Attribute>
</Attribute>
    <Attribute type="BOOL" name="Bonus" value="true" />
  </Item>
  <Item partitionKey="ID" partitionKeyValue="2" SortKey="Name" SortKeyValue="
Carlos">
    <Attribute type="S" name="Address" value="Pirandello" />
    <Attribute type="N" name="Phone" value="637559681" />
    <Attribute type="BOOL" name="Bonus" value="true" />
    <Attribute type="B" name="DocumentCode" value="qwrqfgdas1223455sdfffr" />
    <Attribute type="S" name="Surname" value="Delgado" />
    <Attribute type="S" name="Profession" value="Programmer" />
  </Item>
</Result>

```

- ❗ Für jedes Element entsteht ein Item-Element mit Feldern für den Primärschlüssel und die dazugehörigen Werte. Innerhalb des Item-Elements findet sich eine Liste der dazugehörigen Attribute.

- ListTables-Operation

Beispiel-Output für die Operation ListTables

```
<?xml version="1.0" encoding="UTF-8" ?>
<Result>
  <sampleItems />
</Result>
```

- ❗ Es wird ein Elternelement `Result` geliefert, das ein Element für jede Tabelle enthält, die im Account deklariert ist.

- Query-Operation

Beispiel-Output für die Operation Query

```
<?xml version="1.0" encoding="UTF-8" ?>
<Result hasMorePages="false">
  <Item>
    <Attribute value="{Phone=637559681, Bonus=true,
DocumentCode=[B@231dee4d, Profession=Programmer]}" name="Cars" />
    <Attribute value="34666777888" name="Phone" />
    <Attribute value="[Red, Green, Blue, {Phone=637559681, Bonus=true,
DocumentCode=[B@5857d888, Profession=Programmer]}" name="Colours" />
    <Attribute value="true" name="Bonus" />
    <Attribute value="5" name="ID" />
    <Attribute value="Luis" name="Name" />
  </Item>
</Result>
```

- ❗ Es wird ein Elternelement `Result` geliefert. Das Attribut `hasMorePages` gibt an, ob das Ergebnis noch mehr Seiten hat, die angezeigt werden sollen oder nicht. Das ist hilfreich bei umfangreichen Antworten. Für jedes Element, das die Abfragebedingungen erfüllt, wird ein `Item`-Element erzeugt, das alle Attribute enthält.


6.2.2 Apache Cassandra Connector

Apache Cassandra Connector: Verbindet sich mit einer Apache Cassandra Datenbank.

Eigenschaften (Properties)

Operation	Vom Adapter ausgeführte Operation Mögliche Werte: <ul style="list-style-type: none"> • Execute: Führt das im Input-Dokument angegebene Statement aus • Terminate: Schließt und entfernt das Connection Pooling mit allen offenen Verbindungen
-----------	---

Parameter

Adapter	<p>Adapter-Hauptklasse (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.apache.cassandra.ApacheCassandraConnectorAdapter</code>: Hauptklasse (Standard)</p>
host	<p>Host innerhalb des Clusters</p> <pre>Cluster cluster1 = Cluster.builder().addContactPoints("172.30.210.11", "172.30.210.12") .build();</pre> <p> Sie können mehrere IPs oder Host-Namen hinzufügen, die verschiedenen Servern entsprechen, die als ausführbare Datei von Apache Cassandra innerhalb des Moduls ausgeführt werden.</p>
port	Native-Protocol-Port
login	<p>Login, wenn das Cluster Zugangsdaten erfordert</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Login, z. B. <i>myLogin</i></p>
password	<p>Passwort, wenn das Cluster Zugangsdaten erfordert</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Passwort, z. B. <i>myPassword</i></p>

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt
-1	Bei der Ausführung des Adapters kam es zu einem Fehler

Input

Der Adapter erwartet ein spezifisches XML-Dokument als Input:

Beispiel-Input

```




<RootElement>
  <Preparedstatement resulttag="result element name" timeZone="timezone">
    <Cql>
      select * from keyspace.table;
    </Cql>
    <Parameters>
      <parameter index="n" type="Cassandra_Type">Value</parameter>
    </Parameters>
  </Preparedstatement>
  <Preparedstatement resulttag="result 2 element name" timeZone="timezone 2">
    <Cql>
      INSERT INTO keyspace.table (id, lastname, firstname) VALUES (uuid(),
myLastname,'myFirstName');
    </Cql>
  </Preparedstatement>
  ....
</RootElement>

```

Das Input-XML-Dokument muss folgenden Regeln entsprechen:

Element / Attribut	Beschreibung
resulttag	<i>Erforderliches</i> Attribut des Elements Preparedstatement.
timeZone	<i>Optionales</i> Attribut des Elements Preparedstatement. Bestimmt die Zeitzone entsprechend zu <code>java.util.TimeZone</code> (z. B. <code>GMT+2</code>). Standardmäßig wird die UTC-Zeitzone für Datumsangaben verwendet.
Cql	Element innerhalb des Elements Preparedstatement. Enthält das CQL-Statement. Kann beliebig oft innerhalb von Preparedstatement verwendet werden. <div> <p>i Wenn innerhalb des CQL-Statements Zeichen verwendet werden, die in XML vorbelegt sind, dann muss das SQL-Statement mit einem CDATA-Block escaped werden.</p> </div>
Parameters	Element innerhalb des Elements Preparedstatement. Darf nur genau einmal innerhalb von Preparedstatement verwendet werden. Enthält beliebig viele Elemente parameter mit Parameterwerten.

Element / Attribut	Beschreibung
parameter	<p>Element innerhalb des Elements Parameters. Kann beliebig oft innerhalb von Parameters verwendet werden.</p> <p>Attribute:</p> <ul style="list-style-type: none"> • index: Definiert die Position, an der der Parameterwert das ? innerhalb des CQL-Statements ersetzt. • type: Definiert den Parametertyp. Folgende Werte sind mit Apache Cassandra möglich: <ul style="list-style-type: none"> • TEXT: UTF-8-kodierter Text • VARCHAR: UTF-8-kodierter Text • ASCII: US-ASCII-kodierter Text • BLOB: Hexadezimal kodierte Binärdateien ohne Validierung • BOOLEAN: Bool'sche Werte • BIGINT: 64-Bit-Ganzzahl • COUNTER: 64-Bit-Zählerwert • INT: 32-Bit-Ganzzahl • SMALLINT: 2-Byte-Ganzzahl • TINYINT: 1-Byte-Ganzzahl • DATE: Datum im Format yyyy-mm-dd <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p>i Apache Cassandra kodiert Datumsangaben als 32-Bit-Ganzzahl, die die Tage seit dem 1. Januar 1970 repräsentiert.</p> </div> <ul style="list-style-type: none"> • DECIMAL: Dezimalzahl • DOUBLE: 64-Bit-IEEE-Fließkommazahl • FLOAT: 32-Bit-IEEE-Fließkommazahl • INET: IP-Adresse im IPv4- oder IPv6-Format. • TIME: Uhrzeit <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p>i Apache Cassandra kodiert Zeitangaben als 64-Bit-Ganzzahl, die die Nanosekunden seit Mitternacht repräsentiert.</p> </div> <ul style="list-style-type: none"> • TIMESTAMP: Datum und Uhrzeit. Das Format muss über das Attribut format angegeben werden, z. B.: <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <pre><parameter index="n" type="TIMESTAMP" format="yyyy-MM-dd HH:mm:ss">2012-06-01 08:00:00</parameter></pre> </div> <ul style="list-style-type: none"> • TIMEUUID: Version 1 UUID • UUID: UUID im Standard UUID-Format • VARINT: Ganzzahl • MAP: Array von Zeichen im JSON-Format • LIST: Sammlung eines oder mehrerer geordneter Elemente, z. B. <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <pre><parameter index="n" type="LIST">["tcodd@relational.com","ted.codd@relational.com"]</parameter></pre> </div>

Element / Attribut	Beschreibung
	<div> <p> Die Verwendung der Anführungszeichen ist erforderlich.</p> <ul style="list-style-type: none"> • SET: Sammlung eines oder mehrerer Elemente, z. B. <pre><parameter index="n" type="SET">["Element 1","Element 1","Element 3"]</parameter></pre> <p> Die Verwendung der Anführungszeichen ist erforderlich.</p> <ul style="list-style-type: none"> • TUPLE: Gruppe von Komma-getrennten Feldern. Kann verschiedene Simple Types enthalten: <ul style="list-style-type: none"> • keySpace: Name des Key Space der Tabelle • table: Name der Tabelle • column: Name der Spalte <div> <p> Wenn das Tuple Date-, Time- oder Timestamp-Felder enthält sind folgende Attribute erforderlich:</p> <ul style="list-style-type: none"> • formatWithoutTime: Erforderlich, falls das Tuple Date-Felder enthält. • format: Erforderlich, falls das Tuple Timestamp-Felder enthält. • formatTime: Erforderlich, falls das Tuple Time-Felder enthält. </div> <div> <p>Beispiel Tuple</p> <pre><parameter index="n" type="TUPLE" keySpace="videodb" table="videos" column="tupleColl" formatWithoutTime="yyyy-MM-dd" format="yyyy-MM-dd HH:mm:ss" formatTime="HH:mm:ss:SSS">1, fisrt tuple element, V1, 1.01, 14:25:16:555, 2012-06-01 08:00:00, 2018-04-03, 127.0.0.1, 5.678, 9e816244-f90a-4394-86a5-ddcf4fb9f7aa, true, 97, blobValue</parameter></pre> <p>Im obigen Beispiel wird das Tuple-Feld während der Tabellenerstellung wie folgt definiert:</p> <pre>tupleColl tuple<int, text, varchar, float, time, timestamp, date, inet, decimal, uuid, boolean, ascii, blob></pre> </div> </div>

Die Ausgabe ist ein XML-Dokument mit den Fehlern und dem Ergebnis der Adapter-Ausführung. Je nach Operation gibt der Adapter Daten in verschiedenen XML-Strukturen aus:

- Batch-Operationen (INSERT, UPDATE, DROP, CREATE, ...)


Beispiel Batch-Operationen

```
<Result>
  <resulttag_name> Operation executed successfully</resulttag_name>
</Result>
```

- SELECT

Beispiel SELECT

```
<Result>
  <resulttag_name>
    <Object>
      <Value name="field_name" type="String|Number|
Boolean...">field_value</Value>
      ...
      <Object name="Map_field_name">
        <Value name="key_name" type="String|Number|
Boolean...">object_value</Value>
      </Object>
      <Array name="Colleciton_field_name">
        <Value type="String|Number|
Boolean...">value_of_the_array_at_this_position</Value>
        ...
      </Array>
      ...
    </Object>
  </resulttag_name>
</Result>
```

 Wenn das Feld den Datentyp MAP hat, wird sein Wert durch einen Objektknoten repräsentiert. Wenn das Feld den Datentyp LIST, SET oder TUPLE hat, wird sein Wert durch einen Array-Knoten repräsentiert. Wenn das Feld ein Simple Type ist, wird sein Wert innerhalb des Value-Knotens repräsentiert.

Beispiel

- CREATE

Beispiel CREATE

```
<RootElement>
  <Preparedstatement resulttag="createTableVideos">
    <Cql>
      <![CDATA[CREATE TABLE videodb.videos (
        videoid uuid,
        videoname varchar,
        username varchar,
        description varchar,
        location map<varchar,varchar>,
        tags set<varchar>,
        tupleColl tuple<int, text, varchar, float, time, timestamp,
date, inet, decimal, uuid, boolean, ascii, blob>,
        upload_date timestamp,
        upload_date_noTime date,
        PRIMARY KEY (videoid));]]>
    </Cql>
  </Preparedstatement>
</RootElement>
```

- DROP

Beispiel DROP

```
<RootElement>
  <Preparedstatement resulttag="drop_Users">
    <Cql>
      <![CDATA[DROP TABLE IF EXISTS videodb.users;]]>
    </Cql>
  </Preparedstatement>
</RootElement>
```

- INSERT

Beispiel INSERT

```

<RootElement>
  <Preparedstatement resulttag="insertVideos" timeZone="UTC">
    <Cql>
      INSERT INTO videodb.videos (videoid, videoname, username,
        description, location, tags, tupleColl, upload_date, upload_date_noTime)
        VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?);
    </Cql>
    <Parameters>
      <parameter index="0" type="UUID">99051fe9-6a9c-46c2-
b949-38ef78858dd0</parameter>
      <parameter index="1" type="VARCHAR">My funny cat</parameter>
      <parameter index="2" type="VARCHAR">tcodd</parameter>
      <parameter index="3" type="VARCHAR">My cat likes to play the piano!
So funny.</parameter>
      <parameter index="4" type="MAP">{"US":"/us/vid/
99/99051fe9-6a9c-46c2-b949-38ef78858dd0"}</parameter>
      <parameter index="5" type="SET" >["cats","piano","lol"]</parameter>
      <parameter index="6" type="TUPLE" keySpace="videodb" table="videos"
column="tupleColl" formatWithoutTime="yyyy-MM-dd" format="yyyy-MM-dd HH:mm:ss"
formatTime="HH:mm:ss:SSS">1, first tuple element, V1, 1.01, 14:25:16:555,
2012-06-01 08:00:00, 2018-04-03, 127.0.0.1, 5.678, 9e816244-f90a-4394-86a5-
ddcf4fb9f7aa, true, 97, blobValue</parameter>
      <parameter index="7" type="TIMESTAMP" format="yyyy-MM-dd
HH:mm:ss">2012-06-01 08:45:32</parameter>
      <parameter index="8" type="DATE" formatWithoutTime="yyyy-MM-
dd">2012-06-01</parameter>
    </Parameters>
  </Preparedstatement>
</RootElement>

```

- UPDATE

Beispiel UPDATE

```

<RootElement>
  <Preparedstatement resulttag="add_email_in_list">
    <Cql>
      <![CDATA[ UPDATE videodb.users set email = ? + email where username
= ?]]>
    </Cql>
    <Parameters>
      <parameter type="LIST" index="0">["test1@mail.com","test0@mail.com"
]</parameter>
      <parameter type="TEXT" index="1">pmcfadin</parameter>
    </Parameters>
  </Preparedstatement>
</RootElement>

```

- SELECT

Beispiel SELECT

```

<mainNode>
  <Preparedstatement resulttag="select_NY_rides" timezone="UTC">
    <Cql>
      <![CDATA[SELECT * from NY.rides
        where vendor_id = ?
        and pickup_datetime = ?
        and rate_code = ?
        and total_amount = ?
        limit 10 ALLOW FILTERING;
      ]]>
    </Cql>
    <Parameters>
      <parameter index="0" type="TEXT">2</parameter>
      <parameter index="1" type="TIMESTAMP" format="yyyy-MM-dd
hh:mm:ss">2016-01-02 17:02:28</parameter>
      <parameter index="2" type="INT">1</parameter>
      <parameter index="3" type="DECIMAL">64.56</parameter>
    </Parameters>
  </Preparedstatement>
</mainNode>

```

- DELETE

Beispiel DELETE

```

<mainNode>
  <Preparedstatement resulttag="delete_video_one_record" timezone="UTC">
    <Cql>
      DELETE FROM videodb.videos WHERE videoid = ?;
    </Cql>
    <Parameters>
      <parameter index="0" type="UUID">7dacb4da-9155-4613-
b04e-8c5305a20342</parameter>
    </Parameters>
  </Preparedstatement>
</mainNode>

```

6.2.3 ElasticSearch

6.2.3.1 ElasticSearch Connector




ElasticSearch Connector: Stellt eine Verbindung zu einer ElasticSearch-Datenbank (bis einschließlich ElasticSearch-Version 5.x) über deren ReST-API her und führt ein Statement aus, das als Input-XML- oder JSON-Dokument mit adapterspezifischen Strukturen übergeben wurde.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>GetXMLDocument</i>: JSON-Dokument mit der angegebenen ID aus dem Index und Type laden und als XML-Dokument ausgeben • <i>PutXMLDocument</i>: Erwartet ein spezielles XML-Dokument, wandelt dieses in ein JSON-Dokument um und überträgt dieses • <i>DeleteXMLDocument</i>: Daten unter dem angegebenen Index, Type und ID löschen und als Antwort ein XML-Dokument ausgeben • <i>URISearchXML</i>: Erwartet einen Query-String oder ein XML-Dokument für die Suche und liefert ein XML-Dokument als Ergebnis • <i>GetJSONDocument</i>: JSON-Dokument mit der angegebenen ID aus dem Index und Type laden und dieses ausgeben • <i>PutJSONDocument</i>: Übergebenes JSON-Dokument übertragen • <i>DeleteJSONDocument</i>: Daten unter dem angegebenen Index, Type und ID löschen und ein JSON-Dokument als Antwort ausgeben • <i>URISearchJSON</i>: Erwartet einen Query-String oder ein XML-Dokument für die Suche und liefert ein JSON-Dokument als Ergebnis
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.elastic.ElasticsearchConnector</code>: Hauptklasse (Standard)</p>
<i>url</i>	<p>URL der ElasticSearch-Instanz (Pflichtparameter)</p> <p>Mögliche Werte: Beliebige gültige URL, z. B. <code>http://localhost:9200/</code></p>
<i>index</i>	<p>Definiert in welchen Index das Dokument geschrieben werden soll (Pflichtparameter, bei Suche optional)</p> <p>Mögliche Werte: Beliebiger gültiger Index-Name, z. B. <code>x4</code></p>
<i>type</i>	<p>Definiert den Typ innerhalb des Index (Pflichtparameter, bei Suche optional)</p> <p>Mögliche Werte: Beliebiger gültiger Type-Name, z. B. <code>document</code></p>

<i>id</i>	<p>Definiert die ID des Dokumentes. Wenn kein Wert angegeben ist, wird dieser durch ElasticSearch generiert.</p> <p>Mögliche Werte: Beliebige gültige ID, z. B. <i>id42</i></p>
<i>username</i>	<p>Benutzername für eine Basic-, Digest- oder NTLM-Authentifizierung</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>HTTP-Basic-Authentifizierung</i>: Benutzername • <i>HTTP-Basic-Authentifizierung an einer Windows-Domäne</i>: Zeichenkette nach dem Muster <Domäne>\<Benutzername>, z. B. <i>MyDomain\MyUserName</i> • <i>NTLM-Authentifizierung an einer Windows-Domäne</i>: nur der Benutzername; In Parameter <i>ntlmDomain</i> die Domäne angeben, z. B. <i>MyUsername</i>
<i>password</i>	<p>Passwort für eine Basic-, Digest- oder NTLM-Authentifizierung</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>ntlmWorkstation</i>	<p>Workstation (Computer-Name) des Clients</p> <p> Nur für NTLM-Authentifizierung, siehe Parameter <i>allowNtlm</i></p> <p>Mögliche Werte: Beliebiger gültiger Workstation-Name</p>
<i>ntlmDomain</i>	<p>Domänen-Name des Servers</p> <p> Nur für NTLM-Authentifizierung, siehe Parameter <i>allowNtlm</i></p> <p>Mögliche Werte: Beliebiger gültiger Domänen-Name</p>
<i>usePreemptiveBasicAuth</i>	<p>Präemptive Authentifizierung mit den Zugangsdaten aus <i>username</i> und <i>password</i> für HTTP(S)-Aufrufe aktivieren</p> <p> Nur für Basic-Authentifizierung, siehe Parameter <i>allowBasic</i></p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Zugangsdaten direkt (präemptiv) bei jedem HTTP(S)-Aufruf mitschicken • <i>false</i>: Keine präemptive Authentifizierung (Standard)

<i>allowBasic</i>	<p>HTTP(S)-Basic-Authentifizierung verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>true</i>: Basic-Authentifizierung verwenden• <i>false</i>: Keine Basic-Authentifizierung verwenden (Standard)
<i>allowDigest</i>	<p>Digest-Authentifizierung verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>true</i>: Per Digest authentifizieren• <i>false</i>: Nicht per Digest authentifizieren (Standard)
<i>allowNtlm</i>	<p>Proprietäre NTLM-Server-Authentifizierung von Microsoft verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>true</i>: Per NTLM authentifizieren• <i>false</i>: Nicht per NTLM authentifizieren (Standard)
<i>useProxy</i>	<p>Proxy-Server für den Verbindungsaufbau verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>true</i>: Verbindung über den in <i>proxyHost</i> angegeben Proxy-Server aufbauen• <i>false</i>: Kein Proxy-Server (Standard)
<i>proxyHost</i>	<p>IP-Adresse oder Host-Name eines Proxy-Servers</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette, z. B. <i>192.168.1.1</i></p>
<i>proxyPort</i>	<p>Port-Nummer des Proxy-Servers</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• Beliebige ganze Zahl• <i>-1</i>: Standard-Port verwenden (Standard)

<i>proxyUsername</i>	<p>Benutzername für eine Basic-, Digest- oder NTLM-Authentifizierung am Proxy-Server</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>HTTP-Basic-Authentifizierung</i>: Benutzername • <i>HTTP-Basic-Authentifizierung an einer Windows-Domäne</i>: Zeichenkette nach dem Muster <Domäne>\<Benutzername>, z. B. <i>MyDomain\MyUserName</i> • <i>NTLM-Authentifizierung an einer Windows-Domäne</i>: nur der Benutzername; In Parameter <i>ntlmDomain</i> die Domäne angeben, z. B. <i>MyUsername</i>
<i>proxyPassword</i>	<p>Passwort für den Proxy-Server</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>proxyNtlmWorkstation</i>	<p>Workstation, von der die Authentifizierungs-Anfrage ausgeht</p> <p> Für eine NTLM-Authentifizierung via NTLM-Proxy</p> <p>Mögliche Werte: Beliebiger Workstation-Name</p>
<i>proxyNtlmDomain</i>	<p>Windows-Domäne für eine NTLM-Authentifizierung via NTLM-Proxy</p> <p>Mögliche Werte: Beliebiger Domänen-Name</p>
<i>proxyUsePreemptiveBasicAuth</i>	<p>Präemptive Authentifizierung mit den Zugangsdaten aus <i>proxyUsername</i> und <i>proxyPassword</i> für den Proxy-Server aktivieren</p> <p> Nur für Basic-Authentifizierung, siehe Parameter <i>proxyAllowBasic</i></p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Zugangsdaten direkt (präemptiv) bei jedem Proxy-Aufruf mitschicken • <i>false</i>: Keine präemptive Authentifizierung (Standard)
<i>proxyAllowBasic</i>	<p>Basic-Authentifizierung für den Proxy verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Basic-Authentifizierung für den Proxy verwenden • <i>false</i>: Keine Basic-Authentifizierung für den Proxy verwenden (Standard)

<i>proxyAllowDigest</i>	Digest-Authentifizierung für den Proxy verwenden Mögliche Werte: <ul style="list-style-type: none"> • <i>true</i>: Am Proxy per Digest authentifizieren • <i>false</i>: Nicht per Digest am Proxy authentifizieren (Standard)
<i>proxyAllowNtlm</i>	Proprietäre NTLM-Server-Authentifizierung von Microsoft für den Proxy verwenden Mögliche Werte: <ul style="list-style-type: none"> • <i>true</i>: Am Proxy per NTLM authentifizieren • <i>false</i>: Nicht per NTLM am Proxy authentifizieren (Standard)

Statuswerte

<i>1(successful)</i>	Die Anfrage war erfolgreich; Entspricht dem HTTP-Status-Code 200
<i>0(empty)</i>	Die Anfrage war zwar erfolgreich, es wurde jedoch kein Ergebnis zurückgegeben
<i>-1(failed)</i>	Die Anfrage schlug aufgrund eines technischen Fehlers fehl
<i>-HTTP Status Code</i>	Die Anfrage wurde ausgeführt, jedoch mit einem Status-Code ungleich 200 beendet. Fehler-Interpretation entsprechend dem HTTP-Status-Code.

Input***XML-Dokument für die Operationen `putXMLDocument` und `putJSONDocument`***

Für das Hinzufügen von Dokumenten können beliebige JSON-Dokumente oder XML-Dokumente im Format des [JSON Converter 2.0](#) übergeben werden.

Beispiel: XML-Dokument im JSON Converter 2.0 Format

```
<Object>
  <Value key="stringKey" type="String">Hallo Welt!</Value>
  <Value key="integerKey" type="Number">0</Value>
</Object>
```

Im Falle der Operation `putXMLDocument` wird dieses XML-Dokument in ein JSON-Dokument umgewandelt und an ElasticSearch übertragen. Alternativ kann dieses JSON-Dokument direkt mit der Operation `putJSONDocument` übertragen werden:

Beispiel: JSON-Dokument

```
{
  "stringKey": "Hallo Welt!",
  "integerKey": 0
}
```

XML-Dokument für die Operationen *URISearchXML* und *URISearchJSON***Beispiel: XML-Dokument für *URISearchXML* und *URISearchJSON***

```
<Query>
  <q>stringKey:Hallo Welt!</q>
  <q>integerKey:0</q>
</Query>
```

Erläuterungen:

Gültige Elemente unterhalb des Query-Elements sind folgende Parameter:

- **q:** Suchanfrage im Format `property:value`
- **sort:** Sortierung des Ergebnisses im Format `property:asc|desc`
- **from:** Information, ab welcher Position das Ergebnis ausgegeben werden soll
- **size:** Anzahl der auszugebenden Ergebnisse
- **timeout:** Information, nach welcher Zeit die Anfrage abgebrochen werden soll. Zeitangaben werden mit Wert und Einheit angegeben, z.B. *2d* oder *250ms*

i Wird das XML-Dokument an die Operation übergeben, wird damit eine Suche in Elasticsearch ausgelöst: `http://localhost:9200/index/type/_search?q=stringKey:Hallo%20Welt!&q=integerKey:0`. Alternativ kann der Query-String direkt als Text in einer Datei übergeben werden.

6.2.3.2 Elasticsearch Time Series Connector

ElasticSearch Time Series Connector: Ermöglicht das Ablegen und Auslesen von beliebigen Messwerten und Metriken in einem definierten Format in Elasticsearch.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt
	Mögliche Werte: <ul style="list-style-type: none"> • <i>Put</i>: Erwartet ein spezielles XML-Dokument, wandelt dieses in einzelne JSON-Dokumente um und überträgt diese • <i>Get</i>: Erwartet ein spezielles XML-Dokument, wandelt dieses in eine Abfrage um und ruft die Daten ab • <i>GetAggregate</i>: Erwartet ein spezielles XML-Dokument, wandelt dieses in eine Abfrage um und ruft aggregierten Daten ab

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.elastic.ElasticsearchTimeSeries</i>: Hauptklasse (Standard)</p>
<i>url</i>	<p>URL der Elasticsearch-Instanz (Pflichtparameter)</p> <p>Mögliche Werte: Beliebige gültige URL, z. B. http://localhost:9200/</p>
<i>index</i>	<p>Definiert, in welchen Index das Dokument geschrieben werden soll (Pflichtparameter)</p> <p>Mögliche Werte: Beliebiger gültiger Index-Name, z. B. <i>x4</i></p>
<i>indexDateFormat</i>	<p>Definiert den variablen Datumsteil des Index (optional)</p> <p>Mögliche Werte: Beliebiger gültiger Index-Teilname mit Datumsformatierung, z.B. <i>yyyy-MM-dd</i></p>
<i>fetchSize</i>	<p>Durch die Eingabe einer positiven ganzen Zahl lässt sich die Anzahl der Ergebnisse pro Abfrage beschränken.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>-1</i>: gibt die maximal mögliche Anzahl an Daten aus (Standard) • <i>0-10000</i>: Anzahl der ausgegebenen Daten, wobei 10000 die maximal möglich Anzahl darstellt
<i>forceMapping</i>	<p>Legt ein optimiertes Template-Mapping an. Das Mapping muss für jeden Index nur einmalig angelegt werden.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Mapping wird angelegt (Standard) • <i>false</i>: Mapping wird nicht angelegt
<i>username</i>	<p>Benutzername für eine Basic-, Digest- oder NTLM-Authentifizierung</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>HTTP-Basic-Authentifizierung</i>: Benutzername • <i>HTTP-Basic-Authentifizierung an einer Windows-Domäne</i>: Zeichenkette nach dem Muster <i><Domäne>\<Benutzername></i>, z. B. <i>MyDomain\MyUserName</i> • <i>NTLM-Authentifizierung an einer Windows-Domäne</i>: nur der Benutzername; In Parameter <i>ntlmDomain</i> die Domäne angeben, z. B. <i>MyUsername</i>

<i>password</i>	<p>Passwort für eine Basic-, Digest- oder NTLM-Authentifizierung</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>ntlmWorkstation</i>	<p>Workstation (Computer-Name) des Clients Nur für NTLM-Authentifizierung, siehe Parameter <code>allowNtlm</code></p> <p>Mögliche Werte: Beliebiger gültiger Workstation-Name</p>
<i>ntlmDomain</i>	<p>Domänen-Name des Servers Nur für NTLM-Authentifizierung, siehe Parameter <code>allowNtlm</code></p> <p>Mögliche Werte: Beliebiger gültiger Domänen-Name</p>
<i>usePreemptiveBasicAuth</i>	<p>Präemptive Authentifizierung mit den Zugangsdaten aus <code>username</code> und <code>password</code> für HTTP(S)-Aufrufe aktivieren Nur für Basic-Authentifizierung, siehe Parameter <code>allowBasic</code></p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Zugangsdaten direkt (präemptiv) bei jedem HTTP(S)-Aufruf mitschicken • <i>false</i>: Keine präemptive Authentifizierung (Standard)
<i>allowBasic</i>	<p>HTTP(S)-Basic-Authentifizierung verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Basic-Authentifizierung verwenden • <i>false</i>: Keine Basic-Authentifizierung verwenden (Standard)
<i>allowDigest</i>	<p>Digest-Authentifizierung verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Per Digest authentifizieren • <i>false</i>: Nicht per Digest authentifizieren (Standard)
<i>allowNtlm</i>	<p>Proprietäre NTLM-Server-Authentifizierung von Microsoft verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Per NTLM authentifizieren • <i>false</i>: Nicht per NTLM authentifizieren (Standard)

<i>useProxy</i>	<p>Proxy-Server für den Verbindungsaufbau verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Verbindung über den in proxyHost angegebenen Proxy-Server aufbauen • <i>false</i>: Kein Proxy-Server (Standard)
<i>proxyHost</i>	<p>IP-Adresse oder Host-Name eines Proxy-Servers</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette, z. B. <i>192.168.1.1</i></p>
<i>proxyPort</i>	<p>Port-Nummer des Proxy-Servers</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl • <i>-1</i>: Standard-Port verwenden (Standard)
<i>proxyUsername</i>	<p>Benutzername für eine Basic-, Digest- oder NTLM-Authentifizierung am Proxy-Server</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>HTTP-Basic-Authentifizierung</i>: Benutzername • <i>HTTP-Basic-Authentifizierung an einer Windows-Domäne</i>: Zeichenkette nach dem Muster <Domäne>\<Benutzername>, z. B. <i>MyDomain\MyUserName</i> • <i>NTLM-Authentifizierung an einer Windows-Domäne</i>: nur der Benutzername; In Parameter <i>ntlmDomain</i> die Domäne angeben, z. B. <i>MyUsername</i>
<i>proxyPassword</i>	<p>Passwort für den Proxy-Server</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>proxyNtlmWorkstation</i>	<p>Workstation, von der die Authentifizierungs-Anfrage ausgeht Für eine NTLM-Authentifizierung via NTLM-Proxy</p> <p>Mögliche Werte: Beliebiger Workstation-Name</p>
<i>proxyNtlmDomain</i>	<p>Windows-Domäne für eine NTLM-Authentifizierung via NTLM-Proxy</p> <p>Mögliche Werte: Beliebiger Domänen-Name</p>

<i>proxyUsePreemptiveBasicAuth</i>	<p>Präemptive Authentifizierung mit den Zugangsdaten aus <code>proxyUsername</code> und <code>proxyPassword</code> für den Proxy-Server aktivieren Nur für Basic-Authentifizierung, siehe Parameter <code>proxyAllowBasic</code></p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Zugangsdaten direkt (präemptiv) bei jedem Proxy-Aufruf mitschicken • <i>false</i>: Keine präemptive Authentifizierung (Standard)
<i>proxyAllowBasic</i>	<p>Basic-Authentifizierung für den Proxy verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Basic-Authentifizierung für den Proxy verwenden • <i>false</i>: Keine Basic-Authentifizierung für den Proxy verwenden (Standard)
<i>proxyAllowDigest</i>	<p>Digest-Authentifizierung für den Proxy verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Am Proxy per Digest authentifizieren • <i>false</i>: Nicht per Digest am Proxy authentifizieren (Standard)
<i>proxyAllowNtlm</i>	<p>Proprietäre NTLM-Server-Authentifizierung von Microsoft für den Proxy verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Am Proxy per NTLM authentifizieren • <i>false</i>: Nicht per NTLM am Proxy authentifizieren (Standard)

Statuswerte

<i>1(successful)</i>	Die Operation wurde erfolgreich ausgeführt
<i>0(empty)</i>	Die Operation hat kein Ergebnis geliefert
<i>-1(failed)</i>	Die Operation schlug aufgrund eines technischen Fehlers fehl

Input**XML-Dokument für die Operation *Put***

Die Operation erwartet für das Ablegen von Zeitreihen folgendes XML-Dokument.

```

<?xml version="1.0" encoding="UTF-8"?>
<MeasurementPoints>
  <MeasurementPoint id="id1">
    <Name>Weather station 1</Name><!-- Opitonal -->
    <Metric type="temparature">
      <Name>Temperatur</Name><!-- Opitonal -->
      <Unit>C</Unit><!-- Opitonal -->
      <Data>
        <Timestamp>2017-01-01T00:00:00.00Z</Timestamp><!-- Optional: SYSDATE
wird verwendet wenn kein Wert vorhanden ist -->
        <Comment>Night is starless</Comment><!-- Optional -->
        <Float>-1.5</Float>
      </Data>
      <Data>
        <Timestamp>2017-01-02T00:00:00.00Z</Timestamp><!-- Optional -->
        <Float>0.5</Float>
      </Data>
    </Metric>
    <Metric type="rainfall">
      <Name>Rainfall</Name><!-- Opitonal -->
      <Unit>mm</Unit><!-- Opitonal -->
      <Data>
        <!-- SYSDATE as Timestamp -->
        <Float>0</Float>
      </Data>
    </Metric>
  </MeasurementPoint>
  <MeasurementPoint id="id2">
    <Name>Weather station 2</Name><!-- Opitonal -->
    <Metric type="temparature">
      <Name>Temperatur</Name><!-- Opitonal -->
      <Unit>C</Unit><!-- Opitonal -->
      <Data>
        <Timestamp>2017-01-01T00:00:00.00Z</Timestamp><!-- Optional -->
        <Float>-1.5</Float>
      </Data>
    </Metric>
  </MeasurementPoint>
</MeasurementPoints>

```

XML-Dokument für die Operation Get

Die Operation erwartet für das Auslesen von Zeitreihen folgendes XML-Dokument.


```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Über die Attribute @start und @stop kann der Zeitraum der Datenabfrage
eingeschränkt werden. -->
<Request start="2017-02-02T00:00:00" stop="2017-03-31T00:00:00" from="0"><!-- @start,
@stop und @from sind optional.

Ist bei
@start und @stop keine Zeitzone ist UTC+0 der Standardwert.
@form:
Offset der zu lesenden Daten aus der Ergebnismenge. -->
  <Metrics><!-- Metrics- und MeasurementPoints-Filter sind Optional -->
    <Type>temperature</Type>
    <Type>rainfall</Type>
  </Metrics>
  <MeasurementPoints>
    <Id>id1</Id>
    <Id>id2</Id>
  </MeasurementPoints>
</Request>

```

XML-Dokument für die Operation GetAggregate

Die Operation erwartet für das Auslesen von Zeitreihen das folgende XML-Dokument.

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Über die Attribute @start und @stop kann der Zeitraum der Datenabfrage
eingeschränkt werden. -->
<Request start="2017-02-02T00:00:00" stop="2017-03-31T00:00:00" from="0"><!-- @start,
@stop und @from sind optional.

Ist bei
@start und @stop keine Zeitzone ist UTC+0 der Standardwert.
@form:
Offset der zu lesenden Daten aus der Ergebnismenge. -->
  <Metrics><!-- Metrics- und MeasurementPoints-Filter sind Optional -->
    <Type>temperature</Type>
    <Type>rainfall</Type>
  </Metrics>
  <MeasurementPoints>
    <Id>id1</Id>
    <Id>id2</Id>
  </MeasurementPoints>
  <Aggregate>
    <GroupBy><!-- Wenn keine Gruppierung angegeben ist, werden die Berechnungen
für die gefundene Gesamtdatenmenge durchgeführt -->
      <!-- Metric, MeasurementPoint und Timestamp sind hier gültige Elemente --
>
      <Metric />
      <MeasurementPoint />
      <Timestamp interval="15m" timezone="+02:00" /><!-- @timezone ist optional
(default ist UTC+0) -->
    </GroupBy>
    <!-- Avg, Min, Max, Sum, Median und Percentiles sind hier gültige Funktions-
Element. Über @count bzw. @total wird die Anzahl
immer bei den gruppierten Elementen in der Ergebnismenge ausgegeben und
steht nicht als Funktion bereit -->
    <Avg />
    <Min />
    <Max />
    <Sum />
    <Median /><!-- Median entspricht <Percentiles percents="50"/> -->
    <Percentiles percents="50 90 95 99" /><!-- @percents ist optional (default
ist 50). Zahlen-Array durch Leerzeichen getrennt. -->
  </Aggregate>
</Request>

```

Ausgabe

XML-Dokument für die Operation Put

Gibt die von Elasticsearch kommende JSON-Antwort als XML-Dokument im Format des [JSON Converter 2.0](#) aus.

XML-Dokument für die Operation Get

Das XML-Dokument ist identisch mit dem Input der Operation Put.

XML-Dokument für die Operation GetAggregate

Die Ausgabe kann durch die unterschiedlichen Aggregations-Funktionen und Reihenfolgen im GroupBy-Element vom Beispiel auch abweichen.

```
<?xml version="1.0" encoding="UTF-8"?>
<MeasurementPoints total="2137" took="19">
  <Timestamp count="744" key="2015-01-01T00:00:00.000Z">
    <Avg>-3.787634406268837</Avg>
    <Min>-9.399999618530273</Min>
    <Max>6.099999904632568</Max>
    <Percentiles>
      <Percent25.0>-5.5</Percent25.0>
      <Percent50.0>-4.0111111005147295</Percent50.0>
      <Percent75.0>-2.3562500327825546</Percent75.0>
      <Percent90.0>-0.4300000041723206</Percent90.0>
      <Percent99.0>4.642000136375458</Percent99.0>
    </Percentiles>
  </Timestamp>
  <Timestamp count="672" key="2015-02-01T00:00:00.000Z">
    <Avg>-2.7650297661533667</Avg>
    <Min>-10.199999809265137</Min>
    <Max>8.0</Max>
    <Percentiles>
      <Percent25.0>-5.9000000095367432</Percent25.0>
      <Percent50.0>-3.299999952316284</Percent50.0>
      <Percent75.0>-0.699999988079071</Percent75.0>
      <Percent90.0>3.7799999475479082</Percent90.0>
      <Percent99.0>7.599999904632568</Percent99.0>
    </Percentiles>
  </Timestamp>
  <Timestamp count="721" key="2015-03-01T00:00:00.000Z">
    <Avg>-0.5556171970150011</Avg>
    <Min>-8.8000000190734863</Min>
    <Max>9.5</Max>
    <Percentiles>
      <Percent25.0>-3.1875000298023224</Percent25.0>
      <Percent50.0>-1.6000000023841858</Percent50.0>
      <Percent75.0>1.75</Percent75.0>
      <Percent90.0>5.199999809265137</Percent90.0>
      <Percent99.0>8.97999992370605</Percent99.0>
    </Percentiles>
  </Timestamp>
</MeasurementPoints>
```

6.2.4 Google Cloud Storage BigTable Connector

Google Cloud BigTable Connector: Ermöglicht den Anschluss an Google Cloud BigTable, einen hochverfügbaren NoSQL-Datenspeicher, der die Arbeit der Datenbankverwaltung entlastet.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • CreateTable: Erstellt eine Tabelle mit Tabellennamen und Spaltenfamilienname Erforderliche Parameter: tableName, columnFamilyName • ListTable: Listet alle Tabellen für das Projekt und die Instanz auf Erforderliche Parameter: tableName, columnFamilyName • PutRow: Fügt Zeilen mit den Werten aus dem XML-Eingabedokument ein. Verwendet columnName und value für das rowKey-Attribut. Erforderliche Parameter: tableName, columnFamilyName • GetRow: Durchsucht die Tabelle, indem RowKey über Parameter übergeben wird und gibt das Ergebnis in einer Ausgabe-XML zurück. Erforderliche Parameter: tableName, columnFamilyName • Scan: Scannt die Tabelle und gibt eine Ausgabe-XML entsprechend der Parameter zurück. Der Scanvorgang filtert die Suche. Erforderliche Parameter: tableName, columnFamilyName, columnName, startRow, stopRow, scanCached, scanResultSize, scanTimestamp, scanMaxVersions, scanCacheBlocks • EnableTable: Aktiviert die Tabelle (Tabelle muss deaktiviert sein). Erforderliche Parameter: tableName, columnFamilyName • DisableTable: Deaktiviert die Tabelle (Tabelle muss aktiviert sein). Erforderliche Parameter: tableName, columnFamilyName • Delete: Löscht die Tabelle. Wenn die Parameter rowKey und columnName angegeben sind, wird nur die entsprechende Spalte in der gewählten Zeile gelöscht. Erforderliche Parameter: tableName, columnFamilyName, columnName, rowKey
-----------	--

Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.google.cloud.bigtable.storage.GoogleCloudBigtableAdapter</i>: Hauptklasse (Standard)</p>
---------	---

serviceAccountFile	<p>Pfad zur JSON-Datei, die den Service Account Key enthält, um sich gegenüber dem Google Cloud Storage zu authentifizieren.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Systempfad: z. B.: C: \\X4\\V_5.4\\Server\\X4DB\\1\\ExampleProcess\\GoogleCloud\\Config\\serviceAccountFile.json • X4DB-URL: z. B.: x4db://1/ExampleProcess/GoogleCloud/Config/serviceAccountFile.json • Xstore-URL: z. B.: xstore://ExampleProcess/GoogleCloud/Config/serviceAccountFile.json
projectId	<p>ID des Projekts</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ID eines Google Cloud Projekts. Kann in der Google Cloud Konsole eingesehen werden.
instanceID	<p>ID der Instanz</p> <p>Mögliche Werte :</p> <ul style="list-style-type: none"> • ID einer Google Cloud Instanz. Kann in der Google Cloud Konsole eingesehen werden. <p>Weitere Informationen, wie eine Instanz in Google BigTable erstellt werden kann, finden sich hier.</p>
tableName	Name der Tabelle. Notwendiger Parameter für alle Operationen
columnFamilyName	Spaltenfamilienname. Gruppiert Spalten mit eindeutigen Namen.
rowKey	Zeilenschlüssel der Zeile. Identifikator um Zeilen einzufügen oder zu durchsuchen.
columnName	Spaltenname. Wird verwendet um mit der Delete-Operation eine bestimmte Spalte in einer Zeile zu löschen.
startRow	<p>Spezifiziert eine bestimmte Zeile, bei der der Scanvorgang beginnt.</p> <p>Mögliche Werte: Zeilenschlüssel rowKey</p>
stopRow	<p>Spezifiziert eine bestimmte Zeile, bei der der Scanvorgang endet. Die Zeile selbst wird nicht mehr in den Scan-Ergebnissen angezeigt.</p> <p>Mögliche Werte: Zeilenschlüssel rowKey</p>
scanCached	<p>Ändert das Scanner-Caching für diesen Scan. Wenn der Parameter nicht gesetzt ist, wird der Caching-Wert der Hosting-Tabelle verwendet.</p> <p>Mögliche Werte: Ganzzahl</p>

scanResultSize	Legt die maximale Anzahl der Scan-Ergebnisse fest. Mögliche Werte: Ganzzahl <div> <i>i</i> Wenn die beiden Parameter scanCached und scanResultSize gesetzt sind, werden einzelne Server Requests entweder durch die Anzahl der Zeilen oder die maximale Ergebnisgröße begrenzt, je nachdem, welches Limit zuerst eintritt. </div>
scanTimeStamp	Filtert nach Spalten mit einem bestimmten Zeitstempel. Mögliche Werte: Ganzzahl
scanMaxVersions	Begrenzt die Anzahl Versionen jeder Spalte, die zurückgegeben werden sollen. Mögliche Werte: Ganzzahl
scanCacheBlocks	Serverseitiges Caching für diesen Scan explizit deaktivieren. Mögliche Werte: true / false

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt.
0	Die Adapter-Operation wurde nicht vollständig ausgeführt, es ist aber kein Fehler aufgetreten.
-1	Bei der Ausführung des Adapters ist ein Fehler aufgetreten.

Input

- Der Adapter erwartet für die PutRow-Operation XML in folgender Struktur:

```

<ListRows>
  <row rowKey="IT employee">
    <rowAttribute1 columnName="ID" value="00001" />
    <rowAttribute2 columnName="Address" value="Karlsruhe" />
    <rowAttribute3 columnName="Name" value="Paolo" />
    <rowAttribute4 columnName="Surname" value="Green" />
    <rowAttribute5 columnName="Phone" value="552-968-369" />
    <rowAttribute6 columnName="Bonus" value="Yes" />
    <rowAttribute7 columnName="Profession" value="System Administrator" />
  </row>
</ListRows>

```

- i* Elementnamen können frei gewählt werden, allerdings sind die Attribute rowKey, columnName und value erforderlich.

Output

Der Adapter liefert abhängig von der ausgeführten Operation unterschiedliche Arten von Dokumenten zurück.

- GetRow-Operation ausführen:

```
<Result>
  <Row rowKey="DEV employee">
    <RowColumn ColumnName="Address" value="Ettlingen" />
  <RowColumn ColumnName="Frameworks" value="Java-Hibernate" />
    <RowColumn ColumnName="ID" value="00003" />
    <RowColumn ColumnName="Name" value="Lorena" />
  </Row>
</Result>
```

- Scan-Operation ausführen:

```

<Column FamilyName="FamilyNamePrimary">
  <Row rowKey="Clean employee">
    <Attribute ColumnName="Address" value="Ettlingen" timeStamp="1544603282
081" />
    <Attribute ColumnName="ID" value="00005" timeStamp="1544603282081" />
  </Row>
  <Row rowKey="DEV employee">
    <Attribute ColumnName="Address" value="Ettlingen" timeStamp="1544603280
340" />
    <Attribute ColumnName="Frameworks" value="Java-Hibernate" timeStamp="15
44603280340" />
    <Attribute ColumnName="ID" value="00003" timeStamp="1544603280340" />
    <Attribute ColumnName="Name" value="Lorena" timeStamp="1544603280340" /
>
  </Row>
  <Row rowKey="HR employee">
    <Attribute ColumnName="ID" value="00002" timeStamp="1544603278048" />
    <Attribute ColumnName="Name" value="Francis Ernesto" timeStamp="1544603
278048" />
    <Attribute ColumnName="Surname" value="Underwood" timeStamp="1544603278
048" />
  </Row>
  <Row rowKey="IT employee">
    <Attribute ColumnName="Address" value="Karlsruhe" timeStamp="1544603276
625" />
    <Attribute ColumnName="Bonus" value="Yes" timeStamp="1544603276625" />
    <Attribute ColumnName="ID" value="00001" timeStamp="1544603276625" />
    <Attribute ColumnName="Name" value="Paolo" timeStamp="1544603276625" />
    <Attribute ColumnName="Phone" value="552-968-369" timeStamp="1544603276
625" />
    <Attribute ColumnName="Profession" value="System Administrator" timeStamp="
1544603276625" />
    <Attribute ColumnName="Surname" value="Green" timeStamp="1544603276625"
/>
  </Row>
  <Row rowKey="Marketing employee">
    <Attribute ColumnName="ID" value="00004" timeStamp="1544603281175" />
    <Attribute ColumnName="Name" value="Petra" timeStamp="1544603281175" />
  </Row>
  <Row rowKey="Security employee">
    <Attribute ColumnName="Address" value="Ettlingen/Build1" timeStamp="154
4603283155" />
    <Attribute ColumnName="ID" value="00006" timeStamp="1544603283155" />
    <Attribute ColumnName="Name" value="Robert" timeStamp="1544603283155" /
>
    <Attribute ColumnName="Permissions" value="Day-Night" timeStamp="1544603
283155" />
  </Row>
</Column>

```


- ListTable-Operation ausführen:


```
<?xml version="1.0" encoding="UTF-8" ?>
<Result>
  <ListTables>
    <Table name="TestTable" />
  </ListTables>
</Result>
```

6.2.5 Microsoft Azure Data Lake Gen 1 Storage Connector

Microsoft Azure Data Lake Gen 1 Storage Connector: Ermöglicht die Kommunikation mit Microsoft Azure Data Lakes der ersten Generation (basiert auf Apache Hadoop).

Eigenschaften


Operation	<p>Bestimmt, welche Operation der Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • AppendFile: Datei zu einer bereits vorhandenen Datei hinzufügen. • ConcatenateFiles: Eine Liste von Dateien zu einer Zielfeile zusammenführen.  Nach der erfolgreichen Zusammenführung werden die Quelldateien gelöscht. Ist die Zielfeile noch nicht vorhanden, wird diese automatisch angelegt. Andernfalls wird der Inhalt zusammengeführt. • CreateFolder: Ordner erstellen. • DeleteFile: Datei löschen. • DeleteFolder: Ordner und seinen gesamten Inhalt rekursiv löschen. • EnumerateDirectory: Alle Dateien und Ordner aus einem bestimmten Ordnerpfad auflisten. • GetMetadata: Metadaten einer Datei oder eines Ordners auslesen. • ReadFile: Datei lesen. • RenameFile: Datei umbenennen. Wenn die Zielfeile bereits existiert (eine Datei mit demselben Namen), wird die Quelldatei nur dann umbenannt, wenn der Parameter <code>overwrite</code> aktiviert ist. • RenameFolder: Ordner umbenennen. Wenn der Zielordner bereits existiert, wird der Quellordner in den Zielordner verschoben. • SetPermission: Berechtigungen für eine Datei oder einen Ordner erteilen. • UploadFile: Neue Datei anlegen.
-----------	---

Parameters

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.microsoft.azure.datalake.MicrosoftAzureDataLakeStoreConnector</code>: Hauptklasse (Standard)</p>
authEndpoint	<p><i>Pflichtparameter</i>: OAuth 2.0 Token-Endpoint (v1)</p> <p>Mögliche Werte: Zeichenkette mit einem gültigen Endpunkt, z. B. <code>https://login.microsoftonline.com/<DirectoryID>/oauth2/token</code></p>

clientID	<p><i>Pflichtparameter:</i> Client-ID der "Azure Active Directory"-Anwendung.</p> <p>Mögliche Werte: Zeichenkette mit der Client-ID, z. B. <i>10925xxx-1111-4444-bd3d-82165c0d3841</i></p>
clientSecret	<p><i>Pflichtparameter:</i> Geheimer Clientschlüssel der "Azure Active Directory"-Anwendung.</p> <p>Mögliche Werte: Zeichenkette mit dem geheimen Clientschlüssel.</p>
accountName	<p><i>Pflichtparameter:</i> Voll qualifizierter Data Lake Storage Account-Name (FQDN).</p> <p>Mögliche Werte: Zeichenkette mit dem FQDN, z. B. <i><your_account_name>.azuredatalakestore.net</i></p>
fileName	<p>Dateiname</p> <div> <p> • Dieser Parameter ist für die Operationen AppendFile, Concatenate Files, DeleteFile, ReadFile, RenameFile und UploadFile erforderlich.</p> <p>• Dieser Parameter ist optional für die Operationen GetMetadata und SetPermission.</p> </div> <p>Mögliche Werte: Zeichenkette mit dem Dateinamen, z. B. <i>inputFile.txt</i> oder <i>root/subfolder/inputFile.txt</i></p> <div> <p> Die Zeichenkette darf sowohl den vollständigen Dateipfad (<i>root/subfolder/inputFile.txt</i>) als auch den Dateinamen (<i>fileName inputFile.txt</i>) und Ordernamen (<i>folderPath root/subfolder</i>) enthalten.</p> </div>
folderPath	<p>Ordnerpfad; Kann mit einem Schrägstrich beginnen und enden.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>/</i>: Wurzelordner • Beliebiger gültiger Pfad, z. B. <i>root/subFolder/.../lastFolder</i>
overwrite	<p>Angabe, ob vorhandene Dateien überschrieben werden sollen.</p> <div> <p> Dieser Parameter ist nur für die Operationen UploadFile und RenameFile gültig.</p> </div> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Vorhandene Datei (in <i>fileName</i>) angegeben) mit neuem Inhalt überschreiben • <i>false</i>: Für die Operation UploadFile gibt der Adapter einen Fehler aus, wenn die Datei bereits existiert. Für die Operation RenameFile gibt der Adapter den Status 1 aus, benennt die Quelldatei jedoch nicht um.

listFilesToConcatenate	<p>Name der Dateien, die zusammengeführt werden sollen getrennt durch einen Doppelpunkt.</p> <div>  Dieser Parameter ist für die Operation ConcatenateFiles notwendig. </div> <p>Mögliche Werte: Vorhandene Dateien (vollständige Pfadangabe) durch Doppelpunkt getrennt, z.B. <i>files/File1.txt:files/File2.txt:otherPath/File3.txt</i></p>
renamedObject	<p>Neuer Pfadname der umzubenennenden Datei oder des Ordners.</p> <div>  Dieser Parameter ist für die Operationen RenameFile und RenameFolder notwendig. </div> <p>Mögliche Werte: Pfadname, z.B. <i>files/newFileName.txt</i></p>
octalPermissions	<p>Zu setzende Berechtigungen in Unix-Octal-Format. Für mehr Informationen zu Unix-Dateirechten, siehe auch https://www.tutorialspoint.com/unix/unix-file-permission.htm.</p> <p><i>Beispiel:</i> 770 → 7 Eigener Benutzer mit Lese-, Schreib- und Ausführ-Berechtigung; 7 Gruppe mit Lese-, Schreib- und Ausführ-Berechtigung; 0 Anderer Benutzer ohne Berechtigungen.</p> <div>  Dieser Parameter ist für die Operation SetPermission notwendig. </div>
maxEntriesToRetrieve	<p>Maximale Anzahl von Einträgen, die abgerufen werden sollen.</p> <div>  Dieser Parameter ist optional und wird ignoriert, wenn der Parameter entriesRecursively aktiviert ist. Hilfreich für die Operation EnumerateDirectory. </div> <div>  Beachten Sie, dass der Server die Anzahl der abgerufenen Einträge auf eine kleinere Zahl als die angegebene Zahl begrenzen kann. </div> <p>Mögliche Werte: Ganze Zahl größer 0.</p>
startAfterPathName	<p>Datei- oder Ordnername, nach dem die Aufzählung beginnen soll.</p> <div>  Dieser Parameter ist optional und wird ignoriert, wenn der Parameter entriesRecursively aktiviert ist. Hilfreich für die Operation EnumerateDirectory. </div> <p>Mögliche Werte: Zeichenkette mit dem Pfadnamen der Datei oder des Ordners.</p>

entriesRecursively	<p>Ermöglicht es, Einträge rekursiv zu extrahieren. Wenn der Parameter aktiviert ist, werden <code>maxEntriesToRetrieve</code> und <code>startAfterPathName</code> ignoriert.</p> <div>  Dieser Parameter ist optional und für die Operation <code>EnumerateDirectory</code> hilfreich. </div> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>true</code>: Alle Einträge des angegebenen Ordnerpfads werden rekursiv abgerufen. • <code>false</code>: Alle Einträge des angegebenen Ordnerpfads werden abgerufen, sodass nur direkte Kind-Elemente abgerufen werden.
--------------------	---

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt
-1	Bei der Ausführung kam es zu einem Fehler

Input

Für die Operationen `UploadFile` und `AppendFile` erwartet der Adapter ein beliebiges Input-Dokument. Alle anderen Operationen benötigen kein Input.

Ausgabe

Je nach gewählter Operation gibt der Adapter verschiedene Dokumente aus:

AppendFile

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter das Input-Dokument zurück.
- Im Falle eines Fehlers (Status -1), gibt der Adapter ein XML-Dokument mit der Fehlerbeschreibung zurück, z. B.

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>com.microsoft.azure.datalake.store.ADLException: Error appending to file /ivan/subfolder22/input.txt Operation APPEND failed with HTTP404 :
FileNotFoundException Last encountered exception thrown after 1 tries. [HTTP404(FileNotFoundException)] [ServerRequestId:7eacc9bc-a0d4-48c0-
b15a-678156cb10c5]</Error>
```

CreateFolder

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter das Input-Dokument zurück – falls ein Input-Dokument zugewiesen wurde.
- Im Falle eines Fehlers (Status -1), gibt der Adapter ein XML-Dokument mit der Fehlerbeschreibung zurück, z. B.

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>com.microsoft.azure.datalake.store.ADLException: Error creating directory path is null Last encountered exception thrown after 1 tries. [HTTP0
(null)] [ServerRequestId:null]</Error>
```

ConcatenateFiles

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter das Input-Dokument zurück.

- Im Falle eines Fehlers (Status -1), gibt der Adapter ein XML-Dokument mit der Fehlerbeschreibung zurück, z. B.

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>com.microsoft.azure.datalake.store.ADLException: Error concatenating files into /files/File1_File2.txt concat() source list contains a file more
than once: Last encountered exception thrown after 1 tries. [ServerRequestId:null]</Error>
```

DeleteFile

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter das Input-Dokument zurück – falls ein Input-Dokument zugewiesen wurde.
- Im Falle eines Fehlers (Status -1), gibt der Adapter ein XML-Dokument mit der Fehlerbeschreibung zurück, z. B.

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>com.microsoft.azure.datalake.store.ADLException: Error deleting directory path is null Last encountered exception thrown after 1 tries. [HTTP0
(null)] [ServerRequestId:null]</Error>
```

DeleteFolder

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter das Input-Dokument zurück – falls ein Input-Dokument zugewiesen wurde.
- Im Falle eines Fehlers (Status -1), gibt der Adapter ein XML-Dokument mit der Fehlerbeschreibung zurück, z. B.

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>com.microsoft.azure.datalake.store.ADLException: Error deleting directory tree path is null Last encountered exception thrown after 1 tries.
[HTTP0(null)] [ServerRequestId:null]</Error>
```

EnumerateDirectory

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter ein XML-Dokument mit einer Liste aller Dateien und Ordner aus, z. B.:

```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>
<Entries>
  <Entry LastAccessTime="Tue Sep 17 10:43:09 CEST 2019" LastModifiedTime="Wed Sep 18 13:22:16 CEST 2019" Group="00000000-0000-0000-0000-000000000000" User="8029dc42-2162-445a-8463-
9af9f9316ab2" Permission="770" Name="Axxx" FullName="/Axxx" Type="DIRECTORY"/>
  <Entry LastAccessTime="Tue Sep 17 10:48:57 CEST 2019" LastModifiedTime="Tue Sep 17 10:48:57 CEST 2019" Group="00000000-0000-0000-0000-000000000000" User="8029dc42-2162-445a-8463-
9af9f9316ab2" Permission="770" Name="B_2" FullName="/Bxxx2/B_2" Type="DIRECTORY"/>
  <Entry LastAccessTime="Tue Sep 17 10:43:14 CEST 2019" LastModifiedTime="Wed Sep 18 12:41:46 CEST 2019" Group="00000000-0000-0000-0000-000000000000" User="8029dc42-2162-445a-8463-
9af9f9316ab2" Permission="770" Name="Bxxx2" FullName="/Bxxx2" Type="DIRECTORY"/>
  <Entry LastAccessTime="Wed Sep 18 13:18:52 CEST 2019" LastModifiedTime="Wed Sep 18 13:18:52 CEST 2019" Group="00000000-0000-0000-0000-000000000000" User="8029dc42-2162-445a-8463-
9af9f9316ab2" Permission="770" Name="File1.txt" FullName="/Cxxx/File1.txt" Type="FILE"/>
  <Entry LastAccessTime="Tue Sep 17 10:43:18 CEST 2019" LastModifiedTime="Wed Sep 18 13:31:06 CEST 2019" Group="00000000-0000-0000-0000-000000000000" User="8029dc42-2162-445a-8463-
9af9f9316ab2" Permission="770" Name="Cxxx" FullName="/Cxxx" Type="DIRECTORY"/>
  <Entry LastAccessTime="Wed Sep 18 13:48:38 CEST 2019" LastModifiedTime="Wed Sep 18 13:48:38 CEST 2019" Group="00000000-0000-0000-0000-000000000000" User="54538915-bca5-4e66-8d93-
dcf2ba9e8652" Permission="770" Name="File1.txt" FullName="/Dxxx/File1.txt" Type="FILE"/>
  <Entry LastAccessTime="Wed Sep 18 13:46:11 CEST 2019" LastModifiedTime="Wed Sep 18 13:46:11 CEST 2019" Group="00000000-0000-0000-0000-000000000000" User="54538915-bca5-4e66-8d93-
dcf2ba9e8652" Permission="770" Name="File_Renamed.txt" FullName="/Dxxx/File_Renamed.txt" Type="FILE"/>
  <Entry LastAccessTime="Wed Sep 18 13:47:46 CEST 2019" LastModifiedTime="Wed Sep 18 13:47:47 CEST 2019" Group="00000000-0000-0000-0000-000000000000" User="54538915-bca5-4e66-8d93-
dcf2ba9e8652" Permission="770" Name="File_Renamed2.txt" FullName="/Dxxx/File_Renamed2.txt" Type="FILE"/>
  <Entry LastAccessTime="Wed Sep 18 13:39:02 CEST 2019" LastModifiedTime="Wed Sep 18 13:48:38 CEST 2019" Group="00000000-0000-0000-0000-000000000000" User="54538915-bca5-4e66-8d93-
dcf2ba9e8652" Permission="770" Name="Dxxx" FullName="/Dxxx" Type="DIRECTORY"/>
  <Entry LastAccessTime="Tue Sep 17 10:42:43 CEST 2019" LastModifiedTime="Tue Sep 17 10:42:43 CEST 2019" Group="00000000-0000-0000-0000-000000000000" User="8029dc42-2162-445a-8463-
9af9f9316ab2" Permission="770" Name="Usuarios_Roles_Servidores.txt" FullName="/Usuarios_Roles_Servidores.txt" Type="FILE"/>

```



Jedes Entry-Element entspricht einer Datei oder einem Ordner. Folgende Attribute beschreiben das Entry-Element:

- LastAccessTime: Zeitpunkt des letzten Zugriffs
- LastModifiedTime: Zeitpunkt der letzten Änderung
- Group: ID der Gruppe, die die Datei oder den Ordner besitzt
- User: ID des Benutzers, der die Datei oder den Ordner besitzt
- Permission: Berechtigungszeichenfolge für die Datei in UNIX-Syntax
- Name: Dateiname
- FullName: Vollständiger Datei- oder Ordnerpfad

- Im Falle eines Fehlers (Status -1), gibt der Adapter ein XML-Dokument mit der Fehlerbeschreibung zurück, z. B.:


```
<?xml version="1.0" encoding="UTF-8"?>
<Error>The folderPath parameter must be informed</Error>
```

GetMetadata

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter ein XML-Dokument mit den Metadaten der Datei oder des Ordners aus.

```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>
<Metadata>
  <Entry LastAccessTime="Wed Sep 18 09:07:45 CEST 2019" LastModifiedTime="Wed Sep 18 09:07:45 CEST 2019" Group="00000000-0000-0000-0000-000000000000" User="54538915-bca5-4e66-8d93-dcf2ba9e8652" Permission="770" Name="File1_File2.txt" FullName="/files/File1_File2.txt" Type="FILE"/>
</Metadata>
```

```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>
<Metadata>
  <Entry LastAccessTime="Tue Sep 17 10:14:18 CEST 2019" LastModifiedTime="Wed Sep 18 09:07:45 CEST 2019" Group="00000000-0000-0000-0000-000000000000" User="8029dc42-2162-445a-8463-9af9f9316ab2" Permission="770" Name="files" FullName="files" Type="DIRECTORY"/>
</Metadata>
```

 Das Wurzelement Metadata enthält ein Entry-Element mit folgenden Attributen:

- LastAccessTime: Zeitpunkt des letzten Zugriffs
- LastModifiedTime: Zeitpunkt der letzten Änderung
- Group: ID der Gruppe, die die Datei oder den Ordner besitzt
- User: ID des Benutzers, der die Datei oder den Ordner besitzt
- Permission: Berechtigungszeichenfolge für die Datei oder den Ordner in UNIX-Stil
- Name: Datei- oder Ordnername
- FullName: Vollständiger Datei- oder Ordnerpfad

- Im Falle eines Fehlers (Status -1), gibt der Adapter ein XML-Dokument mit der Fehlerbeschreibung zurück, z. B.:

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>com.microsoft.azure.datalake.store.ADLException: Error getting info for file File1_File2.txt Operation GETFILESTATUS failed with HTTP404 : FileNotFoundExpection Last encountered exception thrown after 1 tries. [HTTP404(FileNotFoundExpection)] [ServerRequestId:c547c88f-ed41-40a4-bcc9-5f319fd83da7]</Error>
```

ReadFile

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter die eingelesene Datei aus
- Im Falle eines Fehlers (Status -1), gibt der Adapter ein XML-Dokument mit der Fehlerbeschreibung zurück, z. B.

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>com.microsoft.azure.datalake.store.ADLException: Error getting info for file path is null Last encountered exception thrown after 1 tries. [HTTP404(FileNotFoundExpection)] [ServerRequestId:c547c88f-ed41-40a4-bcc9-5f319fd83da7]</Error>
```

RenameFile

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter das Input-Dokument zurück – falls ein Input-Dokument zugewiesen wurde.

- Im Falle eines Fehlers (Status -1), gibt der Adapter ein XML-Dokument mit der Fehlerbeschreibung zurück, z. B.:

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>The 'renamedObject' parameter must be set for this operation</Error>
```

RenameFolder

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter das Input-Dokument zurück – falls ein Input-Dokument zugewiesen wurde.
- Im Falle eines Fehlers (Status -1), gibt der Adapter ein XML-Dokument mit der Fehlerbeschreibung zurück, z. B.:

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>The 'renamedObject' parameter must be set for this operation</Error>
```

SetPermission


- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter das Input-Dokument zurück – falls ein Input-Dokument zugewiesen wurde.
- Im Falle eines Fehlers (Status -1), gibt der Adapter ein XML-Dokument mit der Fehlerbeschreibung zurück, z. B.:

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>com.microsoft.azure.datalake.store.ADLErrorException: Error setting times for /folderA/folderA_2/folderB2/New Text Document.txt Specified permissions are not valid Octal
Permissions: 66055 Last encountered exception thrown after 1 tries. [ServerRequestId:null]</Error>
```

UploadFile

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter die hochgeladene Datei aus
- Im Falle eines Fehlers (Status -1), gibt der Adapter ein XML-Dokument mit der Fehlerbeschreibung zurück, z. B.

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>com.microsoft.azure.datalake.store.ADLErrorException: Error creating file path is null Last encountered exception thrown after 1 tries. [HTTP0(null)]
[ServerRequestId:null]</Error>
```








 Ist der Parameter `folderPath` leer, wird die Datei im Wurzelordner des Benutzerkontos angelegt. Existiert der Pfad nicht, der im Parameter `folderPath` gesetzt wurde, wird dieser angelegt.

6.3 Cloud

6.3.1 Amazon S3 Simple Storage Connector

Amazon S3 Simple Storage Connector: Speichert Daten in Amazon S3 Blob Storage und ermöglicht die Verwendung der X4 Suite, um Daten in Cloud-Umgebungen zu persistieren.




Eigenschaften (Properties)


Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> <p>CreateBucket: Legt ein Bucket an.</p> <p> Für die Operation CreateBucket können verschiedene Konfigurationen vorgenommen werden. Ist beispielsweise der Parameter bucketVersion auf <i>ENABLED</i> gesetzt, wird die Bucket-Versionierung aktiviert. Andernfalls ist die Versionierung deaktiviert. Ist der Parameter publicAccess aktiviert, muss über den Parameter publicPermission eine Berechtigung definiert werden.</p> <p>Erforderliche Parameter: bucket, bucketVersion, publicAccess, publicPermission</p> <p>ChangeBucketVersion: Ändert die Versionierungs-Konfiguration des Buckets.</p> <p> Neue Buckets sind standardmäßig im OFF-Status. Sobald die Versionierung aktiviert wird, kann der Status nicht mehr auf <i>OFF</i> zurückgesetzt werden.</p> <p>Erforderliche Parameter: bucket, bucketVersion</p> <p>GetAccessControlList: Listet die Rechte und Berechtigungen für ein Bucket oder eine Datei auf.</p> <p> Sind die Parameter bucket und file angegeben, werden die Rechte und Berechtigungen für die Datei ausgegeben. Ist nur bucket angegeben, werden die Rechte und Berechtigungen für den Bucket ausgegeben.</p> <p>Erforderliche Parameter: bucket, file</p> <p>AddGrantsPermission: Fügt einem Bucket oder einer Datei Rechte mit Berechtigungen hinzu.</p> <p> Ist der Parameter publicAccess aktiviert, wird ein öffentlich zugänglicher Bucket oder eine Datei angelegt mit den Berechtigungen, die im Parameter publicPermission definiert wurden. Andernfalls werden Informationen aus dem Input genommen. Sind die Parameter bucket und file angegeben, werden die Rechte und Berechtigungen zur Datei hinzugefügt. Ist nur bucket angegeben, werden die Rechte und Berechtigungen dem Bucket hinzugefügt.</p> <p>Erforderliche Parameter: bucket, file</p> <p>RevokeGrantsPermission: Widerruft alle Berechtigungen.</p> <p> Sind die Parameter bucket und file angegeben, werden die Rechte und Berechtigungen für die Datei widerrufen. Ist nur bucket angegeben, werden die Rechte und Berechtigungen für den Bucket widerrufen.</p> <p>Erforderliche Parameter: bucket, file</p> <p>RevokePublicPermission: Widerruft die Berechtigungen für den öffentlichen Zugang.</p> <p> Sind die Parameter bucket und file angegeben, werden die Berechtigungen für den öffentlichen Zugang auf eine Datei widerrufen. Ist nur bucket angegeben, werden die Berechtigungen für den öffentlichen Zugang auf den Bucket widerrufen.</p> <p>Erforderliche Parameter: bucket, file</p> <p>AddBucketPolicy: Fügt eine Bucket-Richtlinie zu einem angegebenen Bucket hinzu.</p> <p> Unter folgenden Links finden Sie Zusatzinformationen, wie Beispiele für Bucket-Richtlinien, Berechtigungen in einer Richtlinien und Bedingungen in Richtlinien.</p> <p>Erforderliche Parameter: bucket</p>
-----------	---

- `GetBucketPolicy`: Ruft die Bucket-Richtlinie auf.
Erforderliche Parameter: `bucket`
- `RemoveBucketPolicy`: Löscht die Richtlinie für den angegebenen Bucket.
Erforderliche Parameter: `bucket`
- `Search`: Sucht nach Informationen im AmazonS3 Simple Storage Konto und optional auch noch nach Objekt-Metadaten.
 ⓘ Ist der Parameter `includeObjectsMetadata` ausgewählt, werden Metadaten-Informationen zu jedem Objekt der Standard-Antwort hinzugefügt.
Erforderliche Parameter: `file`, `includeObjectsMetadata`
- `Upload`: Lädt Dateien zum Amazon S3 Simple Storage Konto hoch.
 ⓘ Legen Sie Ordnerstrukturen mit "/" an, z. B. `Lib/Document/doc.txt`.
Erforderliche Parameter: `bucket`, `file`, `overwrite`, `path`
- `Download`: Lädt Dateien vom Amazon S3 Simple Storage Konto herunter.
Erforderliche Parameter: `bucket`, `file`
- `Copy`: Kopiert Dateien in den Ziel-Bucket.
 ⓘ Um eine Datei zu kopieren, ist eine Berechtigung für das Ziel-Bucket erforderlich.
Erforderliche Parameter: `bucket`, `file`, `targetBucket`, `targetFile`
- `Move`: Verschiebt Dateien in den Ziel-Bucket.
 ⓘ Um eine Datei zu verschieben, ist eine Berechtigung für das Ziel-Bucket erforderlich.
Erforderliche Parameter: `bucket`, `file`, `targetBucket`, `targetFile`
- `Rename`: Benennt Dateien in den Ziel-Bucket um.
- `Delete`: Löscht Dateien oder Buckets.
 ⓘ Sind `bucket` und `file` angegeben, wird die angegebene Datei gelöscht. Ist nur `bucket` angegeben, werden alle Dateien im Bucket und anschließend der Bucket selbst gelöscht. Ist die Bucket-Versionierung eingeschaltet, werden zuerst die Dateiversionen und anschließend der Bucket gelöscht.
- `DeleteMultiFiles`: Löscht mehrere Dateien.
Erforderliche Parameter: `bucket`, `file`

Parameter

Adapter	Adapter-Hauptklasse (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.amazons3.AmazonS3SimpleStorageAdapter</i> : Hauptklasse (Standard)
keyID	Zugangsschlüssel-ID für Anfragen an AWS (Amazon Web Services) ⓘ Die Zugangsdaten werden über Parameter an den Adapter gesendet. Dabei gibt es verschiedene Konfigurationsmöglichkeiten, siehe Zugangsdaten einrichten . Mögliche Werte: Beliebige Zeichenkette mit der Zugangsschlüssel-ID, z. B. <i>AKIAIOSFODNN7EXAMPLE</i>

secretKey	<p>Geheimer Zugangsschlüssel für Anfragen an AWS</p> <p> Die Zugangsdaten werden über Parameter an den Adapter gesendet. Dabei gibt es verschiedene Konfigurationsmöglichkeiten, siehe Zugangsdaten einrichten.</p> <p>Mögliche Werte: Beliebige Zeichenkette mit der geheimen Zugangsschlüssel, z. B. <code>wJa1rXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY</code></p>
region	<p>Bucket-Region</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Namen der Region, z. B. <code>us-east-2</code></p> <p> Buckets, die in einer bestimmten Region erstellt wurden, können nicht von einer anderen Region aus aufgerufen werden, siehe Regionen und Endpoints.</p>
bucket	<p>Bucket-Name</p> <p> Der Bucket-Name muss innerhalb von Amazon S3 eindeutig sein, siehe Buckets.</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Bucket-Namen</p> <p>Folgende Regeln gelten für die Benennung von S3-Buckets in allen AWS-Regionen. Bucket-Namen ...</p> <ul style="list-style-type: none"> • müssen innerhalb von Amazon S3 eindeutig sein • müssen den DNS-Namenskonventionen entsprechen • müssen mindestens 3 und höchstens 63 Zeichen lang sein • dürfen keine Großbuchstaben oder Unterstriche enthalten • müssen mit einem Kleinbuchstaben oder einer Zahl beginnen • müssen eine Reihe aus einem oder mehreren Zeichen sein Aufeinander folgende Namen werden durch einen einzelnen Punkt (.) getrennt. Bucket-Namen können Kleinbuchstaben, Zahlen und Bindestriche enthalten. Jeder Name muss mit einem Kleinbuchstaben oder einer Zahl beginnen und enden. • dürfen nicht als IP-Adresse formatiert sein, z. B. <code>192.168.5.4</code> • Beim Verwenden von virtuellen gehosteten Buckets mit SSL (Secure Sockets Layer) trifft das SSL-Wildcard-Zertifikat nur auf Buckets zu, die keine Punkte enthalten. Um dies zu umgehen, verwenden Sie HTTP oder schreiben Sie Ihre eigene Zertifikatsprüfungs-Logik. Bei virtuellen gehosteten Buckets empfehlen wir keine Punkte (.) im Bucket-Namen zu verwenden.

bucketVersion	<p>Bucket-Versionierung, um mehrere Varianten eines Objekts im selben Bucket zu halten</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• OFF: Die Versionierung ist deaktiviert (Standard)  Ist die Versionierung für ein Bucket aktiviert, kann sie nicht mehr deaktiviert und auf den Status OFF gesetzt werden. Die Versionierung kann lediglich ausgesetzt und auf den Status SUSPENDED gesetzt werden.• ENABLED: Die Bucket-Versionierung ist aktiviert• SUSPENDED: Die Bucket-Versionierung wird ausgesetzt, wenn die Versionierung vorher aktiviert war																		
publicAccess	Gibt an, ob Buckets oder Dateien bei der Operation AddGrantsPermission öffentlich zugänglich sind oder nicht																		
publicPermission	<p>Konfiguration für den öffentlichen Zugang</p> <p>Mögliche Werte:</p> <table><tr><th>Berechtigung</th><th>Bei einem Bucket gestattet</th><th>Bei einem Objekt gestattet</th></tr><tr><td>READ</td><td>Erlaubt es, die Objekte im Bucket aufzulisten</td><td>Erlaubt es, die Objekt-Daten und -Metadaten zu lesen</td></tr><tr><td>WRITE</td><td>Erlaubt es, ein beliebiges Objekt im Bucket zu erstellen, zu überschreiben und zu löschen</td><td>Nicht anwendbar</td></tr><tr><td>READ_ACP</td><td>Erlaubt es, die Bucket-ACP zu lesen</td><td>Erlaubt es, die Objekt-ACP zu lesen</td></tr><tr><td>WRITE_ACP</td><td>Erlaubt es, die ACP für den jeweiligen Bucket zu schreiben</td><td>Erlaubt es, die ACP für das jeweilige Objekt zu schreiben</td></tr><tr><td>FULL_CONTROL</td><td>Erlaubt Zugriff auf die Bucket-Berechtigungen READ, WRITE, READ_ACP und WRITE_ACP</td><td>Erlaubt Zugriff auf die Objekt-Berechtigungen READ, WRITE, READ_ACP und WRITE_ACP</td></tr></table>	Berechtigung	Bei einem Bucket gestattet	Bei einem Objekt gestattet	READ	Erlaubt es, die Objekte im Bucket aufzulisten	Erlaubt es, die Objekt-Daten und -Metadaten zu lesen	WRITE	Erlaubt es, ein beliebiges Objekt im Bucket zu erstellen, zu überschreiben und zu löschen	Nicht anwendbar	READ_ACP	Erlaubt es, die Bucket-ACP zu lesen	Erlaubt es, die Objekt-ACP zu lesen	WRITE_ACP	Erlaubt es, die ACP für den jeweiligen Bucket zu schreiben	Erlaubt es, die ACP für das jeweilige Objekt zu schreiben	FULL_CONTROL	Erlaubt Zugriff auf die Bucket-Berechtigungen READ, WRITE, READ_ACP und WRITE_ACP	Erlaubt Zugriff auf die Objekt-Berechtigungen READ, WRITE, READ_ACP und WRITE_ACP
Berechtigung	Bei einem Bucket gestattet	Bei einem Objekt gestattet																	
READ	Erlaubt es, die Objekte im Bucket aufzulisten	Erlaubt es, die Objekt-Daten und -Metadaten zu lesen																	
WRITE	Erlaubt es, ein beliebiges Objekt im Bucket zu erstellen, zu überschreiben und zu löschen	Nicht anwendbar																	
READ_ACP	Erlaubt es, die Bucket-ACP zu lesen	Erlaubt es, die Objekt-ACP zu lesen																	
WRITE_ACP	Erlaubt es, die ACP für den jeweiligen Bucket zu schreiben	Erlaubt es, die ACP für das jeweilige Objekt zu schreiben																	
FULL_CONTROL	Erlaubt Zugriff auf die Bucket-Berechtigungen READ, WRITE, READ_ACP und WRITE_ACP	Erlaubt Zugriff auf die Objekt-Berechtigungen READ, WRITE, READ_ACP und WRITE_ACP																	
file	<p>Name der zu speichernden Datei, siehe Objekte</p> <p>Folgende Zeichensätze sind im Allgemeinen in Schlüssel-Namen zulässig:</p> <ul style="list-style-type: none">• Alphanumerische Zeichen [0-9a-zA-Z]• Sonderzeichen: !, -, _, ., *, ', (, und) <p>Mögliche Werte: Beliebige Zeichenkette mit dem Objekt-Schlüssel-Namen, z. B.</p> <ul style="list-style-type: none">• <i>4my-organization</i>• <i>my.great_photos-2014/jan/myvacation.jpg</i>• <i>videos/2014/birthday/video1.wmv</i>																		

path	Pfad der hochzuladenden Datei Mögliche Werte: Beliebiger gültiger Pfad, z. B. <ul style="list-style-type: none"> • X4DB-Pfad: <i>x4db://1/.....</i> • XSTORE-Pfad <i>xstore://</i> • Vollständiger Pfad: <i>C:/Documents/....</i>
overwrite	Information, ob die Datei überschrieben werden soll oder nicht Mögliche Werte: <ul style="list-style-type: none"> • <i>true</i>: Überschreibt die Datei • <i>false</i>: Überschreibt die Datei nicht
targetBucket	Name des Buckets, in dem die Datei kopiert oder verschoben werden soll Mögliche Werte: Siehe Parameter <i>bucket</i>
targetFile	Name der Datei, wenn sie unter dem neuen Pfad kopiert oder verschoben wird Mögliche Werte: Siehe Parameter <i>file</i>
includeObjectsMetadata	Gibt an, ob Objekt-Metadaten als Teil der Antwort einbezogen werden oder nicht Possible values: <ul style="list-style-type: none"> • <i>true</i>: Objekt-Metadaten werden als Teil der Antwort einbezogen • <i>false</i>: Objekt-Metadaten werden nicht als Teil der Antwort einbezogen

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt
0	Die Ausführung der Operation wurde noch nicht abgeschlossen oder es wurde kein Ergebnis ausgegeben
-1	Bei der Ausführung des Adapters kam es zu einem Fehler

Input

Der Adapter erwartet ein beliebiges Input-Dokument, z. B. ein Binär- oder ein Base-64-Dokument (XML oder Text).

- Operation *Upload*: Ist der Parameter *path* angegeben, wird kein Input-Dokument vorausgesetzt. In allen anderen Fällen wird ein Input-Dokument erwartet.
- Operation *Download* und *Search*: Es wird kein Input-Dokument vorausgesetzt.
- Operation *AddGrantsPermission*:

Beispiel-Input für die Operation AddGrantsPermission

```
<?xml version='1.0' encoding='UTF-8'?>
<RootElement>
  <Grant id="Canonical User ID">
    <permission> Read </permission>
    <permission> Write </permission>
    <permission> WriteAcp </permission>
    <permission> ReadAcp </permission>
    <permission> FullControl </permission>
  </Grant>
</RootElement>
```

i Um Objekt-Berechtigungen hinzuzufügen, sind die Parameter `bucket` und `file` erforderlich. Um Bucket-Berechtigungen hinzuzufügen, ist der Parameter `bucket` erforderlich und `file` muss leer sein.

- Operation `RevokeGrantsPermission`:

Beispiel-Input für die Operation RevokeGrantsPermission

```
<?xml version='1.0' encoding='UTF-8'?>
<RootElement>
  <Grant id="Canonical User ID">
  </Grant>
</RootElement>
```

i Um Objekt-Berechtigungen hinzuzufügen, sind die Parameter `bucket` und `file` erforderlich. Um Bucket-Berechtigungen hinzuzufügen, ist der Parameter `bucket` erforderlich und `file` muss leer sein.

- Für die Operation `AddBucketPolicy` muss der Input im JSON-Format erfolgen
- i** Unter folgenden Links finden Sie Zusatzinformationen, wie [Beispiele für Bucket-Richtlinien](#), [Berechtigungen in einer Richtlinien](#) und [Bedingungen in Richtlinien](#).
- Operation `DeleteMultiFiles`:

Beispiel-Input für die Operation DeleteMultiFiles

```
<?xml version='1.0' encoding='UTF-8'?>
<RootElement>
  <Object name="File Name"></Object>
  <Object name="File Name"></Object>
  <Object name="File Name"></Object>
  <Object name="File Name"></Object>
</RootElement>
```

Ausgabe

Der Adapter gibt abhängig von der ausgeführten Operation verschiedene Dokumente aus:

- Operation Upload:
 - Wurde der Adapter erfolgreich ausgeführt (Status 1), wird ein XML-Dokument ausgegeben.
 - Kam es zu einem Fehler während der Adapter-Ausführung (Status -1), wird nichts ausgegeben.
- Operation Search:
 - Wurde der Adapter erfolgreich ausgeführt (Status 1), wird ein XML-Dokument mit den verwendeten Filtern ausgegeben.

Beispiel-Output für die Operation SEARCH

```
<?xml version='1.0' encoding='UTF-8'?>
<ListBucket>
  <Bucket name="test-malaga2" region="eu-west-1">
    <Object name="myFile.xml" url="https://s3-eu-west-1.amazonaws.com/
test-malaga2/myFile.xml"/>
    <Object name="myFile2.xml" url="https://s3-eu-
west-1.amazonaws.com/test-malaga2/myFile2.xml"/>
    <Object name="pic2.jpg" url="https://s3-eu-west-1.amazonaws.com/
test-malaga2/pic2.jpg"/>
    <Object name="testing_folders/" url="https://s3-eu-
west-1.amazonaws.com/test-malaga2/testing_folders/">
    <Object name="testing_folders/personalData.txt" url="https://s3-
eu-west-1.amazonaws.com/test-malaga2/testing_folders/personalData.txt"/>
  </Bucket>
</ListBucket>
```

- Wurde der Adapter erfolgreich ausgeführt (Status 1) und der Parameter IncludeObjectsMetada ist ausgewählt , werden zusätzlich die Metadaten für jedes Objekt ausgegeben:

Beispiel-Output für die Operation SEARCH inklusive Metadaten

```
<?xml version='1.0' encoding='UTF-8'?>
<ListBucket>
  <Bucket name="test-malaga2" region="eu-west-1">
    <Object name="myFile.xml" url="https://s3-eu-west-1.amazonaws.com/
test-malaga2/myFile.xml">
      <Metadata lastModified="2018-12-10 13:42:38.000 +0100"
instanceLength="158" eTag="a933d9ca99b473c7921c496727a32f91" contentType="
application/octet-stream" contentLength="158"/>
    </Object>
    <Object name="myFile2.xml" url="https://s3-eu-
west-1.amazonaws.com/test-malaga2/myFile2.xml">
      <Metadata lastModified="2018-12-10 12:02:07.000 +0100"
instanceLength="158" eTag="a933d9ca99b473c7921c496727a32f91" contentType="
application/octet-stream" contentLength="158" contentLanguage="English"/>
    </Object>
    <Object name="pic2.jpg" url="https://s3-eu-west-1.amazonaws.com/
test-malaga2/pic2.jpg">
      <Metadata lastModified="2018-12-10 11:37:52.000 +0100"
instanceLength="59580" eTag="fb913694d969758b431277b366f53d7" contentType="
image/jpeg" contentLength="59580" contentLanguage="Spanish"/>
    </Object>
    <Object name="testing_folders/" url="https://s3-eu-
west-1.amazonaws.com/test-malaga2/testing_folders/">
      <Metadata lastModified="2018-12-10 12:24:54.000 +0100" eTag="d
41d8cd98f00b204e9800998ecf8427e" contentType="application/x-directory"/>
    </Object>
    <Object name="testing_folders/personalData.txt" url="https://s3-
eu-west-1.amazonaws.com/test-malaga2/testing_folders/personalData.txt">
      <Metadata lastModified="2018-12-10 12:30:35.000 +0100"
instanceLength="33" eTag="7960fb01e91834eb9d5ccbacdf45d0e5" contentType="a
pplication/octet-stream" contentLength="33" contentLanguage="Spanish"/>
    </Object>
  </Bucket>
</ListBucket>
```

- Kam es zu einem Fehler während der Adapter-Ausführung (Status -1), wird nichts ausgegeben.
- Operation Download:
 - Wurde der Adapter erfolgreich ausgeführt (Status 1), wird der heruntergeladene Blob (Text, XML, Binärdokument etc.) ausgegeben.
 - Existieren das Objekt oder der Bucket nicht, wird der Status -1 zurückgegeben.
- Operation GetAccessControlList:

Beispiel-Output für die Operation GetAccessControlList

```
<AccessControlList>
  <Object Grant="e0188cd1cbfbb71222450e8e416e138172b036cdf3646ab51fb0ae6334d5
ae51" Permission="FULL_CONTROL"/>
  <Object Grant="7779a9de4e188ca2c3fd5ef47f12f2a0fd3b9b30d283e343e0a55746d8df
4802" Permission="READ"/>
  <Object Grant="7779a9de4e188ca2c3fd5ef47f12f2a0fd3b9b30d283e343e0a55746d8df
4802" Permission="READ_ACP"/>
</AccessControlList>
```


- Operation GetBucketPolicy :

```
{
  "Version":"2012-10-17", "Statement":[
  ]
  "Sid":"AddPerm",
  "Effect":"Allow",
  "Principal": "*",
  "Action":["s3:GetObject"],
  "Resource":["arn:aws:s3:::softprojectamazons3input/*"]
}
```





6.3.2 Amazon SimpleDB Connector

Amazon SimpleDB Connector: Amazon SimpleDB ist ein hochverfügbarer NoSQL-Datenspeicher, der die Arbeit der Datenbankverwaltung entlastet.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • CreateDomain: Legt eine Domain mit Datensätzen an. Erforderliche Parameter: domain • DomainMetadata: Ruft Metadaten zu einer Domain auf, siehe DomainMetadata im offiziellen Handbuch. Erforderliche Parameter: domain • DeleteDomain: Löscht eine Domain. Erforderliche Parameter: domain • ListDomain: Listet alle Domains auf. • Put: Fügt ein Element und seine Attribute hinzu oder aktualisiert sie, oder fügt Attribut-Wert-Paare zu einem bereits vorhandenen Element hinzu; Das Element wird beim Empfang automatisch indiziert. Gemäß dem Parameter <code>replaceExistingAttributes</code> ersetzt der Inhalt bestehende Attribute oder erstellt neue Attribute, wobei auch die aktuellen Attribute erhalten bleiben. Erforderliche Parameter: domain, <code>replaceExistingAttributes</code> • BatchPutAttributes: Fügt bis zu 25 Put-Operationen in einem einzigen Anruf hinzu oder aktualisiert sie. Elemente werden beim Empfang automatisch indiziert. Gemäß dem Parameter <code>replaceExistingAttributes</code> ersetzt der Inhalt bestehende Attribute oder erstellt neue Attribute, wobei auch die aktuellen Attribute erhalten bleiben, siehe BatchPutAttributes im offiziellen Handbuch. Erforderliche Parameter: domain, <code>replaceExistingAttributes</code> • GetItem: Ruft ein Element und alle oder eine Teilmenge seiner Attribute und Werte ab. Erforderliche Parameter: domain, <code>itemName</code> • DeleteItem: Löscht ein Element, ein Attribut oder einen Attribut-Wert. Erforderliche Parameter: domain, <code>itemName</code> • BatchDeleteAttributes: Löscht bis zu 25 Delete-Operationen in einem einzigen Anruf, siehe BatchDeleteAttributes im offiziellen Handbuch. Erforderliche Parameter: domain • Select: Fragt einen Datensatz gemäß der typischen Abfragesyntax <i>“select target from domain_name where query_expression”</i> ab. Unterstützte Wertetests sind: <code>=</code>, <code>!=</code>, <code>like</code>, <code>not like</code>, <code>between</code>, <code>is null</code>, <code>is not null</code>, und <code>every ()</code>. <i>Beispiel: select * from mydomain where every(keyword) = ‘Book’.</i>  Ordnen Sie Ergebnisse mit dem Operator SORT und zählen Sie Elemente, welche die Abfragebedingung(en) erfüllen, mit dem Operator Count. • StoreAmazonS3ObjectMetadata: Speichert die Metadaten zu einem AmazonS3-Objekt in der im Parameter domain spezifizierten Domain. Erforderliche Parameter: domain, <code>replaceExistingAttributes</code>
-----------	---

Parameter

Adapter	<p>Adapter-Hauptklasse (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.amazonsimpledb.AmazonSimpleDBAdapter</i>: Hauptklasse (Standard)</p>
keyId	<p>Zugangsschlüssel-ID für Anfragen an AWS (Amazon Web Services)</p> <p> Die Zugangsdaten werden über Parameter an den Adapter gesendet. Dabei gibt es verschiedene Konfigurationsmöglichkeiten, siehe Zugangsdaten einrichten.</p> <p>Mögliche Werte: Beliebige Zeichenkette mit der Zugangsschlüssel-ID, z. B. <i>AKIAIOSFODNN7EXAMPLE</i></p>
secretKey	<p>Geheimer Zugangsschlüssel für Anfragen an AWS</p> <p> Die Zugangsdaten werden über Parameter an den Adapter gesendet. Dabei gibt es verschiedene Konfigurationsmöglichkeiten, siehe Zugangsdaten einrichten.</p> <p>Mögliche Werte: Beliebige Zeichenkette mit der geheimen Zugangsschlüssel, z. B. <i>wJa1rXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY</i></p>
region	<p>Region der Domains</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Namen der Region, z. B. <i>us-east-2</i></p> <p> Buckets, die in einer bestimmten Region erstellt wurden, können nicht von einer anderen Region aus aufgerufen werden, siehe Regionen und Endpoints.</p>
domain	<p>Domain-Name, der durch einen eindeutigen Schlüssel (Namen) identifiziert wird.</p> <p>Domains werden durch Domain-Registerkarten am unteren Rand der Tabelle dargestellt und ähneln Tabellen mit ähnlichen Daten. Der Name muss zwischen 3 und 255 Zeichen lang sein und kann folgende Zeichen enthalten: a-z, A-Z, 0-9, '_', '-' und '.'.</p> <p> Es können Abfragen gegen eine Domain, jedoch nicht zwischen verschiedenen Domains ausgeführt werden.</p>
itemName	<p>Element-Name</p> <p>Elemente werden durch Zeilen dargestellt und sind Einzelobjekte, die ein oder mehrere Attribut-Namens-Wertepaare enthalten.</p>
replaceExistingAttributes	<p>Gibt an, ob der zu setzende Inhalt den bestehenden überschreibt, wenn das zu setzende Element) ein gemeinsames Attribut enthält.</p> <p>Gilt für die Operationen Put und BatchPutAttributes.</p>

Status

1	Die Adapter-Operation wurde erfolgreich ausgeführt
---	--


0	Die Domain existiert nicht
-1	Bei der Ausführung des Adapters kam es zu einem Fehler

Input

Der Adapter erwartet ein XML-Dokument als Input


- Operation Put:

```
<?xml version="1.0" encoding="UTF-8"
<ListAttributes>
  <item name="..Item name..">
    <Attribute name="..Name of Attribute.." value="..Value of Attribute.."/
  >
    <!-- As many attributes as required -->
  </item>
</ListAttributes>
```

-  Bei der Operation Put sollte nur ein Element-Attribute angegeben werden. Wenn Operationen für mehr als ein Element ausgeführt werden sollen, die Operation BatchPutAttributes verwenden. Es gibt keine Einschränkungen bei den Elementnamen, allerdings ist das Attribut name erforderlich.


- Operation BatchPutAttributes :

```
<?xml version="1.0" encoding="UTF-8"
<ListAttributes>
  <item name="..Item name..">
    <Attribute name="..Name of attribute.." value="..Value of attribute.."/
  >
    <!-- As many attributes as required -->
  </item>
  <!-- As many items as required -->
</ListAttributes>
```

-  Bei der Operation BatchPutAttributes sollten mehrere Element-Attribute angegeben werden. Wenn Operationen für ein einziges Element ausgeführt werden sollen, die Operation Put verwenden. Es gibt keine Einschränkungen bei den Elementnamen, allerdings ist das Attribut name erforderlich.

- Operation Select:

```
<?xml version="1.0" encoding="UTF-8"
<RootElement>
  <queryStatement>
    select * from 'DomainName'
  </queryStatement>
</RootElement>
```


 Das Element `queryStatement` wird benötigt. Für eine korrekte Abfrage, folgende [Anleitung](#) befolgen.

- Operation `BatchDeleteAttributes`:

```
<?xml version="1.0" encoding="UTF-8"
<Items>
  <item name="HR employee">
    <Attribute1 name="ID" value="00002"/>
  </item>
  <item3 name="DEV employee">
    <Attribute1 name="ID" value="00003"/>
    <Attribute2 name="Address" value="C\Ettlingen"/>
    <Attribute3 name="Name" value="Lorena"/>
    <Attribute4 name="Frameworks" value="Java-Hibernate"/>
    <Attribute4 name="Frameworks" value="Angular"/>
  </item>
</Items>
```

- Operation `StoreAmazonS3ObjectMetadata`:

```
<?xml version='1.0' encoding='UTF-8'?>
<ListBucket>
  <Bucket name="SoftProject-bucket" region="eu-west-1">
    <Object name="myFile.xml" url="https://s3-eu-west-1.amazonaws.com/test-
malaga2/myFile.xml">
      <Metadata lastModified="2018-12-10 13:42:38.000 +0100"
instanceLength="158" eTag="a933d9ca99b473c7921c496727a32f91" contentType="appli
cation/octet-stream" contentLength="158"/>
    </Object>
    <Object name="myFile2.xml" url="https://s3-eu-west-1.amazonaws.com/
test-malaga2/myFile2.xml">
      <Metadata lastModified="2018-12-10 12:02:07.000 +0100"
instanceLength="158" eTag="a933d9ca99b473c7921c496727a32f91" contentType="appli
cation/octet-stream" contentLength="158" contentLanguage="English"/>
    </Object>
  </Bucket>
</ListBucket>
```

-  • Die erwartete Input-XML-Struktur kann mit Hilfe der Operation `Search` des [Amazon S3 Simple Storage Connectors](#) erzeugt werden. Dabei muss die Adapter-Eigenschaft `includeObjectsMetadata` aktiviert sein.
- Stored attributes for each object metadata need to match with the Amazon metadata fields, see [Object Key and Metadata](#).

Additionally to the provided metadata fields, a new field will be stored in order to be able to filter the metadata. The field `metadataType` provides the values `STANDARD_METADATA` or `USER_METADATA` or both, if there are standard fields and user fields, see [Object metadata](#).

```
<?xmlversion='1.0'encoding='UTF-8'?>
<ListBucket>
  <Bucketname="testmalaga2"region="euwest1">
    <Objectname="testing_folders/personalData.txt"url="https://
s3euwest1.amazonaws.com/testmalaga2/testing_folders/personalData.txt">
      <MetadataLastModified="2018-12-10 12:30:35.000 +0100"instanceLength="
33"eTag="7960fb01e91834eb9d5ccbacdf45d0e5"contentType="application/octetstream"c
ontentLength="33"contentLanguage="Spanish"/>
      <UserMetadataciudad="Malaga"/>
    </Object>
  </Bucket>
</ListBucket>
```

Ausgabe

Der Adapter gibt je nach ausgeführter Operation unterschiedliche Arten von Dokumenten zurück:

- Operation ListDomain:

```
<?xml version="1.0" encoding="UTF-8"
<ListDomain>
  <Domain name="Domain Name">
</ListDomain>
```

- Operation GetItem:

```
<?xml version="1.0" encoding="UTF-8"
<Result>
  <Item id="Item Name">
    <Attribute name="Attribute name" value="Attribute value"/>
    <Attribute name="Attribute name" value="Attribute value"/>
    <Attribute name="Attribute name" value="Attribute value"/>
  </Item>
</Result>
```

- Operation Select:

```
<?xml version="1.0" encoding="UTF-8"
<Result>
  <Item id="Item1 Name">
    <Attribute name="Attribute name" value="Attribute value"/>
    <Attribute name="Attribute name" value="Attribute value"/>
    <Attribute name="Attribute name" value="Attribute value"/>
  </Item>
  <Item id="Item2 Name">
    <Attribute name="Attribute name" value="Attribute value"/>
    <Attribute name="Attribute name" value="Attribute value"/>
    <Attribute name="Attribute name" value="Attribute value"/>
  </Item>
  <Item id="Item3 Name">
    <Attribute name="Attribute name" value="Attribute value"/>
    <Attribute name="Attribute name" value="Attribute value"/>
    <Attribute name="Attribute name" value="Attribute value"/>
  </Item>
</Result>
```

- Alle anderen Operationen geben das Input-Dokument zurück.

6.3.3 Dropbox Connector

Dropbox Connector: Stellt eine Verbindung zum Dropbox-Cloud-Dienst her und bietet Funktionen zur Verwaltung von Cloud-Speicherdaten.



Für die Verwendung des Dropbox Connectors wird ein Dropbox-Konto (Standard oder Business) benötigt.

Vor der ersten Verwendung des Adapters:

Wenn der Adapter Dropbox Connector zum ersten Mal verwendet werden soll, dann müssen zuvor folgende Schritte ausgeführt werden:

1. Die Website <https://www.dropbox.com/developers> aufrufen.
2. Auf **App Console** klicken. Der Button befindet sich oben rechts.
3. Auf **Create app** klicken.
 - a. Für **Choose an API** Dropbox API auswählen.
 - b. Für **Choose the type of access you need** Full Dropbox auswählen.
 - c. Für **Name your app** einen beliebigen Namen eingeben.
 - d. Auf **Create app** klicken
 - e. Die App wird erzeugt und die App-Einstellungen öffnen sich.
4. Unter **Generated access token** auf **Generate** klicken.
5. Der Access Token wird erzeugt und angezeigt. Dieser Access Token wird für die Verwendung des Dropbox Connectors benötigt.

⚠ Über die API wird in der zugeordneten Dropbox ein Ordner Apps erstellt, dieser bildet den Anwendungsordner. Jeder, der im Besitz des generierten Access Token ist, hat Zugriff auf diesen Ordner! Achten Sie darauf, wer in den Besitz des Access Token gelangt!

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • CopyFile: Kopiert eine Datei innerhalb der Dropbox. • CreateFile: Fügt der Dropbox eine Datei hinzu. • DeleteFile: Löscht eine Datei innerhalb der Dropbox. • GetFileContent: Gibt den Inhalt einer Datei aus. • GetFileMetadata: Gibt die Metadaten einer Datei aus. • ListFolder: Listen den Inhalt eines Ordners auf. • UpdateFile: Aktualisiert eine Datei.
-----------	--

Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.x4.adapter.dropbox.DropBoxConnector</code>: Hauptklasse (Standard)</p>
accessToken	<p><i>Erforderlich.</i> Wird benötigt, um eine Verbindung mit dem Dropbox-Konto herzustellen.</p> <p>Mögliche Werte: eindeutiger Access Token</p> <div style="border: 1px solid #add8e6; padding: 5px; margin-top: 10px;"> <p>❗ Der Access Token muss in der Dropbox generiert werden.</p> </div>
folderPath	<p>Pfad zu einem Ordner in der Dropbox.</p> <p>Mögliche Werte: Ordnerpfad (Standard: / (oberste Ordner Ebene))</p> <div style="border: 1px solid #ffcc00; padding: 5px; margin-top: 10px;"> <p>⚠ Der Ordnerpfad muss immer mit einem / beginnen und enden.</p> </div> <div style="border: 1px solid #add8e6; padding: 5px; margin-top: 10px;"> <p>❗ Wenn während der Operation CreateFile Ordner verwendet werden, die nicht existieren, dann werden diese automatisch erstellt.</p> </div>

fileName	Dateiname der Datei, die verarbeitet werden soll. Mögliche Werte: Dateiname inkl. Dateiendung
sourcePath	<i>Erforderlich für die Operation CopyFile.</i> Gibt an, welche Datei kopiert werden soll. Mögliche Werte: Dateipfad inkl. Dateiname und Dateiendung, z. B. folder/file.txt
destinationPath	<i>Erforderlich für die Operation CopyFile.</i> Gibt an, welche Datei kopiert werden soll. Mögliche Werte: Dateipfad
overwrite	Definiert, ob vorhandene Dateien beim Kopieren oder Erstellen überschrieben werden. Mögliche Werte: <ul style="list-style-type: none"> • true: Wenn eine Datei mit dem gleichen Namen an diesem Speicherort bereits existiert, dann wird diese Datei überschrieben. • false: Wenn eine Datei mit dem gleichen Namen an diesem Speicherort bereits existiert, dann wird die neue Datei umbenannt, z. B. von File.txt zu Copy_of_File.txt.

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt.
-1	Die Operation konnte nicht ausgeführt werden. Die Fehlermeldung wird als Textdokument und im Server-Log ausgegeben.

Input

Der Adapter erwartet für die Operationen CreateFile und UpdateFile eine Datei als Input.

Output:

Der Adapter liefert abhängig von der ausgeführten Operation unterschiedliche Daten als Output.

- Operationen CopyFile, CreateFile und DeleteFile: Textdokument mit Information
- Operationen ListFolder, GetFileMetadata: XML-Dokument mit entsprechendem Inhalt

Beispieloutput GetFileMetadata

```
<Metadata isFolder="false">
  <Name>new.txt</Name>
  <Path>/new.txt</Path>
  <Id>id:SLz0S6oc-FAAAAAAAAAALw</Id>
  <LastModified>2019-06-26T11:55:33</LastModified>
  <Size>4</Size>
</Metadata>
```

Beispieloutput ListFolder

```
<Result>
  <Entry isFolder="true">Folder</Entry>
  <Entry isFolder="false">Image.png</Entry>
  <Entry isFolder="false">file.txt</Entry>
  <Entry isFolder="false">Document.docx</Entry>
</Result>
```


- Operation GetFileContent: Inhalt der ausgelesenen Datei

 Es werden nicht alle Dateiformate unterstützt!

6.3.4 Google Cloud Storage Objects Connector

Google Cloud Storage Objects Connector: Ermöglicht das Speichern, Suchen, Herunterladen und das Löschen von Objekten in der Google Cloud.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Search: Sucht nach Informationen (Buckets oder Objekte) im Google-Cloud-Speicher Erforderliche Parameter: bucketName, objectName • Upload: Lädt Buckets oder Objekte in den Google-Cloud-Speicher hoch oder legt Buckets oder Objekte im Google-Cloud-Speicher an Erforderliche Parameter: bucketName, objectName, bucketRegion, storageClass, fileLocation, contentType • Download: Lädt ein Objekt vom Google-Cloud-Speicher herunter Erforderliche Parameter: bucketName, objectName • Delete: Löscht ein Bucket oder Objekt im Google-Cloud-Speicher Erforderliche Parameter: bucketName, objectName • CopyObject: Kopiert ein Objekt von einem Bucket zu einem anderen. Es gibt mehrere Möglichkeiten, ein Objekt zu kopieren: Erforderliche Parameter: bucketName, objectName <ul style="list-style-type: none"> • Objekt innerhalb eines Buckets kopieren und den Namen ändern. Erforderliche Parameter: targetObjectName • Objekt in ein anderes Bucket kopieren und den Namen ändern. Erforderliche Parameter: targetObjectName, targetBucketName • Objekt in ein anderes Bucket kopieren und den Namen beibehalten. Erforderliche Parameter: targetBucketName • MoveOrRenameBucket: Verschiebt ein komplettes Bucket oder benennt es um. Erforderliche Parameter: bucketName, targetBucketName, targetBucketRegion <div data-bbox="558 1686 1436 1953"> <p> • Wenn der Name des Ziel-Buckets bereits existiert, dann wird eine Fehlermeldung ausgegeben.</p> <p>• Wenn eines der Objekte im Ziel-Bucket bereits existiert und die <code>overwrite</code>-Option gewählt ist, dann werden die vorhandenen Objekte überschrieben. Wenn die <code>overwrite</code>-Option nicht gewählt ist, dann wird eine Fehlermeldung ausgegeben.</p> </div>
-----------	---

❗ Da ein Bucket zu einer anderen geografischen Region verschoben werden kann, muss auf die Beschränkungen in Verbindung mit dem Verschieben in andere geografische Regionen geachtet werden. Weitere Informationen zu [Bucket-Standorten](#).

- **MoveOrRenameObject:** Verschiebt ein Objekt oder benennt es um.
Erforderliche Parameter: bucketName, objectName, targetBucketName, targetObjectName

❗ Wenn das Objekt im Zielbucket bereits existiert und die overwrite-Option gewählt ist, dann wird das vorhandene Objekt überschrieben. Wenn die overwrite-Option nicht gewählt ist, dann wird eine Fehlermeldung ausgegeben.

- **UpdateObjectStorageClass:** Ändert die zum Objekt zugeordnete storageClass. Die Eigenschaft storageClass ist eng mit der Bucket-Region verbunden. Beim Anlegen eines Buckets muss beispielsweise die Region angegeben werden.
Erforderliche Parameter: bucketName, objectName, storageClass

❗ Nicht alle möglichen Werte sind für jedes Objekt/Bucket erlaubt. Es hängt von der Bucket-Spezifikation und dem Objekt selbst ab. Weitere Informationen zu [Storage-Klassen](#).

❗ Ein regionales Bucket kann nur regionale Objekte speichern. In diesem Fall kann die storageClass eines Objektes nur in nearline oder coldline geändert werden.

Ein multi-regionales Bucket kann nur multi-regionale Objekte speichern. In diesem Fall kann die storageClass eines Objektes nur in nearline oder coldline geändert werden.

Mögliche Werte:

- Objekt ist in einem regionalen Bucket oder einem multi-regional Bucket gespeichert: coldline, nearline
- Objekt ist in einem nearline-Bucket gespeichert, Bucket-Region ist regional oder multi-regional: coldline, regional. Wenn die Bucket-Region nicht regional oder multi-regional ist: nur coldline
- Objekt ist in einem coldline-Bucket gespeichert, Bucket-Region ist regional oder multi-regional: nearline, regional. Wenn die Bucket-Region nicht regional oder multi-regional ist: nur nearline

❗ Wenn eine regionale Region spezifiziert wird, z. B. europe-west1, wird implizit festgelegt, dass die Default-storageClass regional, nearline oder coldline ist, aber nie multi-regional.

Das selbe passiert, wenn eine Region als multiregional, z. B. us, spezifiziert wird: Es wird implizit festgelegt, dass die Default-storageClass multi-regional, nearline oder coldline ist, aber nie regional.

- **AddGrantsPermission:** Fügt eine ACL-Erlaubnis zu einem Objekt oder Bucket hinzu. Wenn das Objekt oder Bucket bereits eine ACL-Erlaubnis hat, passiert nichts.

❗ Erforderliche Parameter:

- Für diesen Vorgang sind die Parameter `permissionEntity`, `permissionRole`, `permissionName` und falls vergeben `permissionProjectRole` erforderlich.
- Erlaubnis einem Objekt hinzufügen: `objectName` und `bucketName`
- Erlaubnis einem Bucket hinzufügen: `bucketName`

- ❗** Wenn eine neue Erlaubnis zu einem Bucket hinzugefügt wird, erhält das Bucket die Erlaubnis, aber nicht die enthaltenen Objekte. Das bedeutet, dass bei einem öffentlichen Bucket jeder das Bucket und die Liste der enthaltenen Objekte sehen kann, aber nicht auf die Objekte zugegriffen werden kann. Dafür wird eine zusätzliche Erlaubnis benötigt.


- **RevokeGrantsPermission:** Entfernt eine ACL-Erlaubnis von einem Objekt oder Bucket. Wenn das Objekt oder Bucket keine ACL-Erlaubnis hat, passiert nichts.

❗ Erforderliche Parameter:

- Für diesen Vorgang sind die Parameter `permissionEntity`, `permissionRole`, `permissionName` und falls vergeben `permissionProjectRole` erforderlich.
- Erlaubnis von einem Objekt entfernen: `objectName` und `bucketName`
- Erlaubnis von einem Bucket entfernen: `bucketName`

Parameter

Adapter	<p>Adapter-Hauptklasse (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.google.cloud.storage.GoogleCloudStorageObjectAdapter</i>: Hauptklasse (Standard)</p>
---------	---

serviceAccountFile	<p>Pfad zur JSON-Datei mit dem Service-Account-Schlüssel zur Authentifizierung gegenüber dem Google-Cloud-Speicher.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • System-Pfad, z. B. <i>C:\X4\V_5.9\Server\X4DB\1\ExampleProcess\GoogleCloud\Config\serviceAccountFile.json</i> • X4db-URL, z. B. <i>x4db://1/ExampleProcess/GoogleCloud/Config/serviceAccountFile.json</i> • Xstore-URL, z. B. <i>xstore://ExampleProcess/GoogleCloud/Config/serviceAccountFile.json</i>
bucketName	<p>Name des Buckets, in dem die Objekte organisiert sind</p> <p>Mögliche Werte: Zeichenkette mit dem Namen</p> <p>Bucket-Namen müssen den folgenden Namenskonventionen entsprechen:</p> <ul style="list-style-type: none"> • Bucket-Namen dürfen nur Kleinbuchstaben, Zahlen, Bindestriche (-), Unterstriche (_) und Punkte (.) enthalten.  Namen, die Punkte enthalten, müssen überprüft werden. • Bucket-Namen müssen mit einer Zahl oder einem Buchstaben beginnen und enden. • Bucket-Namen müssen zwischen 3 und 63 Zeichen enthalten. Namen, die Punkte enthalten, können aus bis zu 222 Zeichen bestehen, jedoch darf jeder durch einen Punkt getrennte Teil nicht länger als 63 Zeichen sein. • Bucket-Namen können nicht aus IP-Adressen mit dem Beispiel-Format <i>192.168.5.4</i> bestehen • Bucket-Namen dürfen nicht mit dem Präfix <i>goog</i> beginnen. • Bucket-Namen dürfen weder <i>google</i> noch ähnliche Schreibweisen für Google wie beispielsweise <i>g00glе</i> enthalten.
targetBucketName	<p><i>Optional.</i></p> <p>Name des Ziel-Buckets.</p> <p>Für Operationen, die ein Bucket kopieren/verschieben/umbenennen. Der Name unterliegt denselben Beschränkungen wie der Parameter <i>bucketName</i>.</p>
bucketRegion	<p>Region, in der die Buckets definiert sind</p> <p>Mögliche Werte: Zeichenkette mit der Region</p> <p><i>Beispiel:</i></p> <ul style="list-style-type: none"> • Regionaler Standort: <i>europe-west3</i> (Europa, Frankfurt) • Multi-regionaler Standort: <i>eu</i> (Rechenzentren in der Europäischen Union)

targetBucketRegion	<p>Name der Region des Ziel-Buckets.</p> <p>Für Operationen, die ein Bucket verschieben/umbenennen und für die eine Zielregion angegeben werden kann. Der Parameter unterliegt denselben Beschränkungen wie der Parameter bucketRegion.</p>
storageClass	<p>Speicher-Klasse des Buckets und der Objekte. Dieser Parameter ist eng verbunden mit Regionen, da ein storageClass-Wert nur einem Bucket oder einem Objekt zugeordnet werden kann, das mit dem gewählten bucketRegion-Wert verbunden ist.</p> <p>Mögliche Werte: Auswahl mit folgenden Werten</p> <ul style="list-style-type: none"> • multi_regional • regional • nearline • coldline
objectName	<p>Name des Objekts. Erforderlich für die Operationen AddGrantsPermission und RevokeGrantsPermission.</p> <p>Mögliche Werte: Zeichenkette mit dem Namen</p> <p>Objekt-Namen müssen den folgenden Namenskonventionen entsprechen:</p> <ul style="list-style-type: none"> • Objekt-Namen können eine beliebige Folge von gültigen Unicode-Zeichen mit einer Länge zwischen 1-1024 Bytes enthalten, wenn sie UTF-8-kodiert sind. • Objekt-Namen dürfen keine Zeilenumbrüche oder Zeilenumbruch-Zeichen enthalten. • Objekt-Namen dürfen nicht mit .well-known/acme-challenge beginnen. • Objekt-Namen dürfen nicht . oder .. benannt werden. <p><i>Beispiele:</i></p> <ul style="list-style-type: none"> • <i>document</i> • <i>folder/subfolder/document</i> • <i>document.txt</i>
targetObjectName	<p>Für Operationen, die ein Objekt kopieren/verschieben/umbenennen. Der Parameter unterliegt denselben Beschränkungen wie der Parameter objectName.</p>

fileLocation	<p>Pfad der Datei, die in den Google-Cloud-Speicher hochgeladen werden soll; Dieser Parameter ist nur für die Operation Upload relevant</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • System-Pfad, z. B. C: <code>\X4\V_5.9\Server\X4DB\1\ExampleProcess\GoogleCloudStorageObject/Data/TextDocument.txt</code> • X4db-Pfad, z. B. x4db://1/ExampleProcess/ <code>GoogleCloudStorageObject/Data/TextDocument.txt</code> • Xstore-Pfad, z. B. xstore://ExampleProcess/ <code>GoogleCloudStorageObject/Data/TextDocument.txt</code>
contentType	<p>Inhalts-Typ des hochzuladenden Objekts</p> <p>Mögliche Werte: Zeichenkette mit einem gültigen Mime-Typ für das hochzuladende Objekt, z. B. <code>text/plain, application/xml</code> etc.</p>
overwrite	<p>Gibt an, ob während einer Operation die ein Objekt oder Bucket verschiebt/kopiert/umbenennt, Buckets oder Objekte mit demselben Namen überschrieben werden. Wenn diese Option nicht gewählt ist und am Zielort Objekte/Buckets mit demselben Namen existieren, enthält der Output eine Fehlermeldung.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Checked</i>: Vorhandene Objekte/Buckets werden überschrieben. • <i>Unchecked</i>: Vorhandene Objekte/Buckets werden nicht überschrieben, der Output enthält eine Fehlermeldung.
permissionEntity	<p>Typ der ACL-Erlaubnis, die erstellt wird.</p> <p>Mögliche Werte: Auswahl mit folgenden Werten</p> <ul style="list-style-type: none"> • Project: Spezifiziert eine Erlaubnis auf Basis einer Liste von Projektmitgliedern. Der Parameter <code>permissionProjectRole</code> muss ebenfalls angegeben werden, um festzulegen, aus welcher Unterliste die Mitglieder gewählt werden sollen. • User: Spezifiziert eine Erlaubnis auf Basis einer bestimmten E-Mail-Adresse, eine Erlaubnis für öffentlichen Zugriff oder eine Erlaubnis für alle Google-Konten. • Domain: Spezifiziert eine Erlaubnis auf Basis einer Internetdomäne, die mit G Suite oder Cloud Identity verbunden ist. • Group: Spezifiziert eine Erlaubnis auf Basis einer Gruppe von Google-Benutzern. Verwendet eine bestimmte E-Mail-Adresse, die mit einer Google-Gruppe verbunden ist.

permissionRole	<p>Spezifiziert, ob die Erlaubnis einen Lese- oder einen Schreibzugriff gewährt.</p> <p>Mögliche Werte: Auswahl mit folgenden Werten</p> <ul style="list-style-type: none"> • READER: Lesezugriff für Objekt/Bucket • WRITER: Lese- und Schreibzugriff für Objekt/Bucket • OWNER: Verwalten des Objekts/Buckets. Enthält die Rechte von WRITER.
permissionName	<p>Angabe eines bestimmten Wertes, der in jeder Art von Berechtigung verwendet wird.</p> <p>Mögliche Werte: abhängig von der Angabe für permissionEntity</p> <ul style="list-style-type: none"> • permissionEntity="Project": Projektnummer. Erzeugt eine Zugriffserlaubnis für die diesem Projekt zugeordneten OWNERS, EDITORS und VIEWERS. • permissionEntity="User": <ul style="list-style-type: none"> • allUsers: Erzeugt eine öffentliche Zugriffserlaubnis • allAuthenticatedUsers: Erzeugt eine Zugriffserlaubnis für alle Google-Konten • <email address>: Erzeugt eine Zugriffserlaubnis für den Benutzer, der mit dieser E-Mail-Adresse registriert ist. • permissionEntity="Domain": <USERNAME>@<YOUR_DOMAIN>.com: Erzeugt eine Zugriffserlaubnis für Inhaber einer E-Mail-Adresse, die einer bestimmten Internetdomäne zugeordnet sind. • permissionEntity="Group": E-Mail-Adresse der Google-Gruppe. Erzeugt eine Zugriffserlaubnis für alle Mitglieder dieser Google-Gruppe. Weitere Informationen zu Google-Gruppen.
permissionProjectRole	<p>Parameter wirkt sich auf die Parameter permissionName und permissionEntity, wenn permissionEntity="Project".</p> <p>Mögliche Werte: Auswahl mit folgenden Werten</p> <ul style="list-style-type: none"> • OWNERS • EDITORS • VIEWERS

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt.
-1	Bei der Ausführung des Adapters kam es zu einem Fehler.
0	<ul style="list-style-type: none"> • Operation Upload: Das hochzuladende Objekt existiert bereits • Operation Download: Das herunterzuladende Objekt existiert nicht • Operation Delete: Das zu löschende Objekt oder Bucket existiert nicht

Input

Der Adapter erwartet für manche Operationen ein beliebiges Input-Dokument (Binär- oder Base64-basiertes Dokument):

- Operation Upload : Ist der Parameter `fileLocation` angegeben, wird kein Input-Dokument vorausgesetzt. Andernfalls wird ein Input vorausgesetzt.
- Alle anderen Operationen setzen kein Input-Dokument voraus.

Ausgabe

Je nach Operation gibt der Adapter verschiedene Ergebnisse aus:

- Operation Upload :
 - Wurde die Adapter-Operation erfolgreich ausgeführt (Status 1), gibt der Adapter das Input-Dokument zurück.
 - Kam es bei der Ausführung des Adapters zu einem Fehler (Status -1), gibt der Adapter ein neues Dokument mit den Fehler-Details aus.
 - Existiert das hochzuladende Objekt bereits (Status 0), entspricht die Ausgabe dem Input.
- Operation Search:
 - Wurde die Adapter-Operation erfolgreich ausgeführt (Status 1), gibt der Adapter ein XML-Dokument mit allen Buckets und ihren Objekten aus:


Beispiel-Ausgabe für die Operation Search

```
<?xml version="1.0" encoding="UTF-8"
<ListObjects>
  <Bucket name="first_bucket">
    <Object contentType="text/plain">example/first_blob</Object>
  <Object contentType="text/plain">file.xml</Object>
    <Object contentType="application/x-www-form-
urlencoded; charset=UTF-8">newFolder</Object>
    <Object contentType="text/xml">newFolder/xml1 (1).xml</Object>
  <Object contentType="text/plain">new_object</Object>
  <Object contentType="application/x-www-form-
urlencoded; charset=UTF-8">otherFolder</Object>
  </Bucket>
  <Bucket name="second_bucket">
    <Object contentType="text/plain">text/Hello</Object>
  </Bucket>
  <Bucket name="x4_upload_test_bucket">
    <Object contentType="image/png">Logo.png</Object>
    <Object contentType="application/xml">inputDocumentX4db.xml</
Object>
    <Object contentType="text/plain">testUploadProcess/Input/textFile</
Object>
    <Object contentType="application/xml">testUploadProcess/Input/
xmlFile.xml</Object>
    <Object>wordDocument2</Object>
    <Object contentType="application/vnd.openxmlformats-
officedocument.wordprocessingml.document">wordDocumentFromLocationFile</
Object>
  </Bucket>
</ListObjects>
```

- Kam es bei der Ausführung des Adapters zu einem Fehler (Status -1), gibt der Adapter ein neues Dokument mit den Fehler-Details aus.
- Operation Download:
 - Wurde die Adapter-Operation erfolgreich ausgeführt (Status 1), ist die Ausgabe das heruntergeladene Objekt (Text, XML, Binär-Dokument etc.).
 - Kam es bei der Ausführung des Adapters zu einem Fehler (Status -1), gibt der Adapter ein neues Dokument mit den Fehler-Details aus.
 - Existiert der Blob oder Container nicht (Status 0), entspricht die Ausgabe dem Input.
- Operation Delete:
 - Wurde die Adapter-Operation erfolgreich ausgeführt (Status 1), entspricht die Ausgabe dem Input.
 - Kam es bei der Ausführung des Adapters zu einem Fehler (Status -1), entspricht die Ausgabe dem Input.
 - Existieren der zu löschende Bucket oder das Objekt nicht (Status 0), entspricht die Ausgabe dem Input.
- Operation CopyObject, MoveOrRenameBucket, MoveOrRenameObject, UpdateObjectStorageClass:
 - Bei regulärer Ausführung kein Output. Wenn die Operation nicht ausgeführt werden kann, wird eine Fehlermeldung ausgegeben.

6.3.5 Google Drive Connector


Google Drive Connector: Stellt eine Verbindung zum Google Drive Cloud-Service her und bietet Cloud-Storage-Datenverwaltungsfunktionen.






 Google Drive erlaubt es, mehrere Objekte mit dem gleichen Namen zu haben. Beim Ausführen von Operationen, die sich auf mehrfach vorhandene Objekte beziehen, wird aktuell eine Fehlermeldung ausgegeben und die Operation wird nicht ausgeführt.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • CopyFile: Kopiert eine bereits vorhandene Datei innerhalb des Google-Drive-Kontos • CreateComment: Legt einen Kommentar an • CreateFolder: Legt einen neuen Ordner im Google-Drive-Konto an • DeleteFolder: Löscht einen bereits vorhandenen Ordner • DeleteFile: Löscht eine bereits vorhandene Datei • ExtractArchiveToFolder: Extrahiert Dateien innerhalb eines ZIP-Ordners in einen Ordner • GetFileContent: Ruft die Inhalte eines bereits vorhandenen Dokuments ab • GetFileMetadata: Ruft die Metadaten einer Datei ab • ListFiles: Führt alle Dateien und Ordner des Wurzelverzeichnisses auf • ListFilesInFolder: Führt alle Dateien und Ordner innerhalb eines angegebenen Ordners auf • ReadComments: Liest bereits vorhandene Kommentare aus • RevokePermission: Widerruft Berechtigung gemäß der Parameterkonfiguration • SetPermission: Legt an oder aktualisiert Berechtigungen gemäß der Parameterkonfiguration • UpdateFile: Aktualisiert den Inhalt oder den Namen einer Datei • UpdateFolderName: Aktualisiert den Namen eines bereits vorhandenen Ordners • UploadFile: Lädt eine Datei in das Google-Drive-Konto hoch
-----------	---

Parameters

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.google.cloud.drive.GoogleCloudDriveAdapter</i>: Hauptklasse (Standard)</p>
serviceAccountFile	<p>Pfad der Servicekontodatei für die Authentifizierung</p> <p>Mögliche Werte: Zeichenkette mit dem Pfad der Servicekontodatei</p>
folderPath	<p>Ordnerpfad, in dem die extrahierte Datei abgelegt wird</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> • Ist der Pfad nicht spezifiziert, wird der Pfad des Wurzelverzeichnisses verwendet.</p> <p>• Dieser Parameter wird für die Operationen ListFilesInFolder, DeleteFile, GetFileContent und GetFileMetadata verwendet.</p> </div> <p>Mögliche Werte: Zeichenkette mit dem Ordnerpfad; Die Pfadangabe muss mit einem Schrägstrich / enden.</p>

sourcePath	<p>Pfad der Quelldatei</p> <div>  Dieser Parameter wird für die Operationen UploadFile, UpdateFile, CopyFile und ExtractArchiveToFolder verwendet. </div> <p>Mögliche Werte: Zeichenkette mit dem Pfad zur Quelldatei</p>
destinationPath	<p>Pfad zum Zielordner oder zur Zieldatei</p> <div>  Dieser Parameter wird für die Operationen CopyFile, UploadFile, UpdateFile und ExtractArchiveToFolder verwendet. </div> <p>Mögliche Werte: Zeichenkette mit dem Pfad zur Zieldatei oder zum Zielordner; Die Pfadangabe muss mit einem Schrägstrich / enden.</p>
fileName	<p>Dateiname – In einigen Fällen wird der Parameter zum Setzen oder Ändern des Dateinamen verwendet.</p> <div>  Dieser Parameter wird für die Operationen UploadFile, DeleteFile, GetFileContent, GetFileMetadata, UpdateFile und CopyFile verwendet. </div> <p>Mögliche Werte: Zeichenkette mit dem Dateinamen</p>
folderName	<p>Ordnername</p> <div>  <ul style="list-style-type: none"> Für die Operation ExtractArchiveToFolder wird der Parameter verwendet, um einen neuen Ordner anzulegen, in dem die Dateien aus dem ZIP-Ordner entpackt werden. Der Parameter wird zudem für die Operationen CreateFolder, UpdateFolderName und DeleteFolder verwendet. </div> <p>Mögliche Werte: Zeichenkette mit dem Ordnernamen</p>
comment	<p>Kommentar</p> <div>  Dieser Parameter wird für <i>create</i>- und <i>read</i>-Operationen verwendet. </div> <p>Mögliche Werte: Zeichenkette mit dem Kommentar</p>

overwrite	<p>Angabe, ob bereits vorhandene Dateien überschrieben werden sollen</p> <div>  Dieser Parameter wird außer für die Operation <code>CreateFolder</code> für alle <i>create</i>-, <i>copy</i>- oder <i>update</i>-Operationen verwendet. Für <i>false</i> wird eine Meldung ausgegeben, wenn die Datei/der Ordner bereits vorhanden ist. </div> <p>Mögliche Werte: Boolescher Wert <i>true</i> oder <i>false</i>.</p>
scope	<p>Geltungsbereich der Berechtigung</p> <div>  Dieser Parameter wird für die Operation <code>SetPermission</code> verwendet. </div> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • DOMAIN • GROUP • USER • ANYONE
role	<p>Rolle der Berechtigung</p> <div>  Dieser Parameter wird für die Operation <code>SetPermission</code> verwendet. </div> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • FILE_ORGANIZER • READER • OWNER • WRITER • COMMENTER
account	<p>Domain oder E-Mail-Adresse des Benutzers oder der Gruppe</p> <div>  Dieser Parameter wird für die Operationen <code>SetPermission</code> und <code>RevokePermission</code> verwendet. </div> <p>Mögliche Werte: Zeichenkette mit der Domain oder E-Mail-Adresse</p>

Statuswerte values

1	Die Adapter-Operation wurde erfolgreich ausgeführt
-1	Bei der Ausführung des Adapters kam es zu einem Fehler

Input

Der Adapter wird über Parameter verwaltet. Für einige Operationen wie `ExtractArchiveToFolder`, `UploadFile` oder `UpdateFile` kann jedoch ein Input übergeben werden, der den hochzuladenden

oder zu aktualisierenden Inhalt darstellt. Wenn bei diesen Operationen kein Input übergeben wird, wird der Parameter `sourcePath` verwendet.

Ausgabe

Der Adapter gibt folgende Struktur aus, wenn eine `/list`-Operation verwendet wird:

Ausgabe für list-Operationen

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <Files>
3      <File FullName="X4-ADK.pdf" id="1_nNnJVwCyNXrFWiGDok1GNKmdZeAcXK7"/
4  >
5      <File FullName="X4-Activities.pdf" id="1soJS2I4JgG4pNxybI9MnTKSON1
6  iLWcI"/>
7      <File FullName="X4-Release-Notes.pdf" id="1C8dh8I6r2LDsPP3avdkYNIIVL
8  CGmvf54T"/>
9      <File FullName="X4-ESB.pdf" id="1lPatZJT2yxI1IdfzLYy8hIN056T0hCdC"/
10 >
11     <File FullName="X4-BPM.pdf" id="1R5_mJ4x6_qcPZhrnaP-z_zoFd6kJJmAn"/
12 >
13     <File FullName="X4-Administration-Guide.pdf" id="1di3dhqUe3bLXx80WQ
14 qtqcBLcIeXPS4N-"/>
15 </Files>

```

Für die Operation `GetFileMetadata` werden folgende Informationen ausgegeben:

Ausgabe für die Operation GetFileMetadata

```


1  <?xml version="1.0" encoding="UTF-8"?>
2  <File FullName="X4-Release-Notes-Comment.pdf" Id="18upYSTk8pkGMm8YVT46LHa9
3  R9myYzf7I">
4      <Kind>drive#file</Kind>
5      <Parents>
6          <Id>0A0vK1n-M5IHmUk9PVA</Id>
7      </Parents>
8      <Permissions>
9          <Permission>
10             <Id>03478937985708787222</Id>
11             <Account>service-account@shaped-
12 fx-259808.iam.gserviceaccount.com</Account>
13             <Scope>user</Scope>
14             <Role>owner</Role>
15         </Permission>
16     </Permissions>
17     <Created>2020-01-07T10:45:33.338</Created>
18     <Size>4376271</Size>
19 </File>

```

6.3.6 Microsoft Azure Blob Storage Connector

Microsoft Azure Blob Storage Connector: Ermöglicht die Anbindung an einen Microsoft Azure Blob Storage.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> Die Parameter <code>containerName</code> und <code>blobName</code> sind für alle Operationen erforderlich.</p> </div> <ul style="list-style-type: none"> • <code>Search</code>: Sucht nach Informationen (Container oder Blobs) im Microsoft-Azure-Konto • <code>Append</code>: Fügt Informationen zum ausgewählten Blob im Microsoft-Azure-Konto hinzu Erforderliche Parameter: <code>fileLocation</code> • <code>Upload</code>: Lädt Container oder Blobs hoch oder legt Container oder Blobs im Microsoft-Azure-Konto an Erforderliche Parameter: <code>fileLocation</code> • <code>Download</code>: Lädt ein Blob vom Microsoft-Azure-Konto herunter • <code>Delete</code>: Löscht ein Container oder Blob vom Microsoft-Azure-Konto • <code>UpdateDirectory</code>: Aktualisiert den Namen des Quellverzeichnisses mit dem Namen des Zielverzeichnisses Erforderliche Parameter: <code>sourceDirectoryName</code>, <code>targetDirectoryName</code> • <code>MoveDirectory</code>: Verschiebt den Inhalt vom Quellverzeichnis in das Zielverzeichnis Erforderliche Parameter: <code>sourceDirectoryName</code>, <code>targetDirectoryName</code> • <code>DeleteDirectory</code>: Löscht den Inhalt des Quellverzeichnisses Erforderliche Parameter: <code>sourceDirectoryName</code>, <code>targetDirectoryName</code> • <code>MoveBlobDirectory</code>: Verschiebt einen bestimmten Blob vom Quellverzeichnis in das Zielverzeichnis Erforderliche Parameter: <code>sourceDirectoryName</code>, <code>targetDirectoryName</code>
-----------	---

Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.microsoft.azure.blob.MicrosoftAzureBlobStorageAdapter</code>: Hauptklasse (Standard)</p>
accountName	<p>Name des Microsoft-Azure-Kontos</p> <p>Mögliche Werte: Beliebige Zeichenkette, z. B. <code>softproject</code></p>

accountKey	<p>Passwort für das Microsoft-Azure-Konto</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Schlüssel z. B. <i>AD9ohFhJ6Hve90DJEXTw53MjR1tjoUZ8lwFDlw5UFqGMih3VtUCapIwIfEMBMITeShQN9MbCITE4SzxTcGIybw==</i></p>
containerName	<p>Name des Containers in dem die Blobs organisiert werden</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Namen</p> <p>Container-Namen müssen folgende Namenskonventionen erfüllen:</p> <ul style="list-style-type: none"> • Container-Namen müssen mit einem Buchstaben oder einer Zahl beginnen und können nur Buchstaben, Zahlen oder Bindestriche (–) enthalten • Jeder Bindestrich (–) muss unmittelbar vor und nach einem Buchstaben oder einer Zahl stehen; aufeinanderfolgende Bindestriche sind in Container-Namen nicht zulässig • Container-Namen dürfen nur Kleinbuchstaben enthalten • Container-Namen müssen zwischen 3 und 63 Zeichen enthalten
blobName	<p>Name der gespeicherten Blobs</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Blob-Namen</p> <p>Blob-Namen müssen folgende Namenskonventionen erfüllen:</p> <ul style="list-style-type: none"> • Blob-Namen können eine beliebige Kombination aus Zeichen enthalten • Blob-Namen müssen mindestens ein Zeichen und dürfen nicht mehr als 1.024 Zeichen lang sein • Bei Blob-Namen wird zwischen Groß- und Kleinbuchstaben unterschieden • Reservierte URL-Zeichen müssen ordnungsgemäß maskiert sein • Die Anzahl der Pfad-Segmente, die den Blob-Namen umfassen, darf 254 nicht überschreiten. Ein Pfad-Segment ist die Zeichenkette zwischen aufeinanderfolgenden Begrenzungszeichen (z. B. der Schrägstrich '/'), die dem Namen eines virtuellen Verzeichnisses entsprechen.
fileLocation	<p>Pfad der hochzuladenden Datei; Dieser Parameter ist nur für die Operation Upload gültig.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • System-Pfad, z. B. <i>C:\X4\V_5.9\Server\X4DB\1\ExampleProcess\MAzureBlobStore\Data\TextDocument.txt</i> • X4db-Pfad, z. B. <i>x4db://1/ExampleProcess/MAzureBlobStore/Data/TextDocument.txt</i> • Xstore-Pfad, z. B. <i>xstore://ExampleProcess/MAzureBlobStore/Data/TextDocument.txt</i>
overwrite	<p>Information, ob Blobs überschrieben werden oder nicht</p>
sourceDirectoryName	<p>Name des Quellverzeichnisses für alle Directory-Operationen</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette mit einem Abschließenden /, z. B. <i>sourceFolder/</i></p>

targetDirectoryName	Name des Zielverzeichnisses für alle Directory-Operationen Mögliche Werte: Beliebige gültige Zeichenkette mit einem Abschließenden / , z. B. <i>targetFolder/</i>
---------------------	---

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt
-1	Es kam zu einem Fehler während der Ausführung
0	<ul style="list-style-type: none"> • Operation Download: Der Blob, der heruntergeladen werden soll, existiert nicht • Operation Delete: Der Blob, der gelöscht werden soll, existiert nicht • Operationen UpdateDirectory, MoveDirectory, DeleteDirectory, MoveBlobDirectory: <ul style="list-style-type: none"> • das Quellverzeichnis existiert nicht • das Zielverzeichnis konnte nicht gefunden werden (Operationen MoveDirectory, MoveBlobDirectory) • der Blob existiert nicht im Quellverzeichnis (Operation MoveBlobDirectory)

Input

Der Adapter erwartet ein beliebiges Input-Dokument, z. B. ein Binär- oder ein Base64-Dokument (XML oder Text).

- Operation Upload: Ist der Parameter `fileLocation` angegeben, so kann auf das Input-Dokument verzichtet werden. In allen anderen Fällen wird ein Input-Dokument vorausgesetzt.
- Für alle anderen Operationen wird kein Input-Dokument vorausgesetzt.

Output

Abhängig von der gewählten Operation, gibt der Adapter verschiedene Dokumente aus:

Operation Upload:

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter ein XML-Dokument mit der URL der hochgeladenen Datei oder dem Dokument aus.

Operation Upload: Status 1
<pre><?xml version="1.0" encoding="UTF-8" <BLOB> <Upload>https://storage_account_name.blob.core.windows.net/container_name/ blob_name</Upload> </BLOB></pre>

- Bei einem Fehler (Status -1), entspricht die Ausgabe dem Input-Dokument.

Operation Search:

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter ein XML-Dokument mit allen Containern und ihren Blob-URI-Objekten aus.

Operation Search: Status 1

```
<?xml version="1.0" encoding="UTF-8"
<ListBlobs>
  <Container name="pc-files">
    <Blob name="C-structure.xml">https://softproject.blob.core.windows.net/
pc-files/C-structure.xml</Blob>
    <Blob name="XStore_WordDocument.doc">https://
softproject.blob.core.windows.net/pc-files/XStore_WordDocument.doc</Blob>
    <Blob name="jar_files(1).zip">https://
softproject.blob.core.windows.net/pc-files/jar_files(1).zip</Blob>
    <Blob name="x4db/HipChat-4.30.6.1676.exe">https://
softproject.blob.core.windows.net/pc-files/x4db/HipChat-4.30.6.1676.exe</Blob>
    <Blob name="xstore/MWDocument.doc">https://
softproject.blob.core.windows.net/pc-files/xstore/MWDocument.doc</Blob>
  </Container>
  <Container name="testcontainer-prototype">
    <Blob name="base64Files/file1">https://
softproject.blob.core.windows.net/testcontainer-prototype/base64Files/file1</
Blob>
    <Blob name="documents/doc1">https://softproject.blob.core.windows.net/
testcontainer-prototype/documents/doc1</Blob>
    <Blob name="documents/doc1.docx">https://
softproject.blob.core.windows.net/testcontainer-prototype/documents/doc1.docx</
Blob>
    <Blob name="imagesFiles/image1">https://
softproject.blob.core.windows.net/testcontainer-prototype/imagesFiles/image1</
Blob>
    <Blob name="textFiles/subTextFiles/text5.txt">https://
softproject.blob.core.windows.net/testcontainer-prototype/textFiles/
subTextFiles/text5.txt</Blob>
    <Blob name="xml1.xml">https://softproject.blob.core.windows.net/
testcontainer-prototype/xml1.xml</Blob>
    <Blob name="xmlFiles/xml1">https://softproject.blob.core.windows.net/
testcontainer-prototype/xmlFiles/xml1</Blob>
    <Blob name="xmlFiles/xml1.xml">https://
softproject.blob.core.windows.net/testcontainer-prototype/xmlFiles/xml1.xml</
Blob>
  </Container>
  <Container name="test">... </Container>
</ListBlobs>
```

- Bei einem Fehler (Status -1), entspricht die Ausgabe dem Input-Dokument.

Operation Download:

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt der Adapter den heruntergeladenen Blob (Text, XML, Binärdokument etc.) aus.
- Bei einem Fehler (Status -1), entspricht die Ausgabe dem Input-Dokument.
- Existieren Blob oder Container nicht (Status 0), entspricht die Ausgabe dem Input-Dokument.

Operation

UpdateDirectory

:

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt das Output-Dokument nach dem Aktualisieren des Quellordners mit dem Namen des Zielordners alle betroffenen Blobs aus.

Operation UpdateDirectory: Status 1

```
<?xml version="1.0" encoding="UTF-8">
<UpdateDirectory targetDirectory="testFolderR/" sourceDirectory="testFolder/">
  <Blob newName="testFolderR/C-structure.xml" name="testFolder/C-
structure.xml"/>
  <Blob newName="testFolderR/X4_1_Configuration.txt" name="testFolder/
X4_1_Configuration.txt"/>
</UpdateDirectory>
```

- Bei einem Fehler (Status -1), entspricht die Ausgabe dem Input-Dokument.
- Existiert das Quellverzeichnis nicht (Status 0), ist das Output ein leeres XML-Dokument.

Operation UpdateDirectory: Status 0

```
<?xml version="1.0" encoding="UTF-8">
<UpdateDirectory targetDirectory="otherFiles/testSubFolderR/" sourceDirectory="
textFiles/subTextFiles"/>
```

Operation MoveDirectory:

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt das Output-Dokument nach dem Verschieben des Quellordners in den Zielordner alle betroffenen Blobs aus.

Operation MoveDirectory: Status 1

```
<?xml version="1.0" encoding="UTF-8">
<MoveDirectory targetDirectory="textFiles/" sourceDirectory="testFolderR/">
  <Blob newName="textFiles/C-structure.xml" name="testFolderR/C-
structure.xml"/>
  <Blob newName="textFiles/X4_1_Configuration.txt" name="testFolderR/
X4_1_Configuration.txt"/>
</MoveDirectory>
```

- Bei einem Fehler (Status -1), entspricht die Ausgabe dem Input-Dokument.
- Existiert das Quell- oder Zielverzeichnis nicht (Status 0), ist das Output ein leeres XML-Dokument.

Operation MoveDirectory: Status 0 - Zielverzeichnis existiert nicht

```
<?xml version="1.0" encoding="UTF-8"
<MoveDirectory targetDirectory="otherFiles/testSubFolderR/" sourceDirectory="textFiles/subFolderR"> Target Directory not found for Move operation </MoveDirectory>
```

Operation MoveDirectory: Status 0 - Quellverzeichnis existiert nicht

```
<?xml version="1.0" encoding="UTF-8"
<MoveBlobDirectory targetDirectory="textFiles/" sourceDirectory="otherFiles/testSubFolderR"/>
```

Operation DeleteDirectory:

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt das Output-Dokument nach dem Löschen des Quellordners und seines Inhalts alle betroffenen Blobs aus.

Operation DeleteDirectory: Status 1

```
<?xml version="1.0" encoding="UTF-8"
<DeleteDirectory targetDirectory="" sourceDirectory="imagesFiles2/">
  <Blob newName="" name="imagesFiles2/image1.png"/>
</DeleteDirectory>
```

- Bei einem Fehler (Status -1), entspricht die Ausgabe dem Input-Dokument.
- Existiert das Quellverzeichnis nicht (Status 0), ist das Output ein leeres XML-Dokument.

Operation DeleteDirectory: Status 0

```
<?xml version="1.0" encoding="UTF-8"
<DeleteDirectory targetDirectory="" sourceDirectory="imagesFiles2"/>
```

Operation MoveBlobDirectory:

- Wurde die Operation erfolgreich ausgeführt (Status 1), gibt das Output-Dokument alle Blobs aus, die vom Quell- in den Zielordner verschoben wurde.

Operation MoveBlobDirectory: Status 1

```
<?xml version="1.0" encoding="UTF-8"
<MoveBlobDirectory targetDirectory="/" sourceDirectory="textFiles/">
  <Blob newName="X4_1_Configuration.txt" name="textFiles/X4_1_Configuration.txt"/>
</MoveBlobDirectory>
```

- Bei einem Fehler (Status -1), entspricht die Ausgabe dem Input-Dokument.
- Existieren Quell- oder Zielverzeichnis nicht (Status 0), ist das Output ein leeres XML-Dokument.

Operation MoveBlobDirectory: Status 0 - Zielverzeichnis existiert nicht

```
<?xml version="1.0" encoding="UTF-8"
<MoveBlobDirectory targetDirectory="otherFiles/testSubFolderR/" sourceDirectory="
textFiles/subFolderR"> Target Directory not found for Move operation </
MoveDirectory>
```

Operation MoveBlobDirectory: Status 0 - Quellverzeichnis existiert nicht

```
<?xml version="1.0" encoding="UTF-8"
<MoveBlobDirectory targetDirectory="textFiles/" sourceDirectory="otherFiles/
testSubFolderR"/>
```

6.3.7 Microsoft Azure Table Storage Connector


Microsoft Azure Table Storage Connector: Ermöglicht die Verbindung zu Microsoft Azure, einer Cloud-Computing-Plattform von Microsoft, die Cloud Cosmos DB Table Storage zur Verfügung stellt.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • CreateTable: Legt eine neue Tabelle an Erforderliche Parameter: tableName • ListTable: Listet alle Tabellen auf • DeleteTable: Löscht eine bereits vorhandene Tabelle Erforderliche Parameter: tableName • InsertEntity: Fügt die Entität ein, wenn die Entität noch nicht vorhanden ist, oder ersetzt eine bestehende Entität. Nachdem die Operation erfolgreich ausgeführt wurde, enthält die Tabelle die neue Entität mit den definierten Eigenschaften und die vorherige Entität und ihre Eigenschaften werden ersetzt. Erforderliche Parameter: tableName, partitionKey, rowKey • InsertBatchEntities: Fügt mehrere Entitäten mit einem Schreibvorgang ein, oder ersetzt bestehende Entitäten. Nutzt die InsertOrReplace-Entität. Erforderliche Parameter: tableName • RetrieveEntity: Ruft eine einzelne, spezifizierte Entität ab Erforderliche Parameter: tableName, partitionKey, rowKey • RetrieveAllEntities: Ruft alle Entitäten von einem bestimmten partitionKey ab. Erforderliche Parameter: tableName, partitionKey • RetrieveRangeEntities: Ruft eine Reihe von Entitäten mit dem partitionKey ab und filtert nach einer Reihe von rowKey-Werten. Der partitionKey-Wert identifiziert eine bestimmte Partition und die rowKey-Werte identifizieren eine Teilmenge der Entitäten dieser Partition. Erforderliche Parameter: tableName, partitionKey, rowKey, filterCondition • DeleteEntity: Löscht eine einzelne Entität. Erforderliche Parameter: tableName, partitionKey, rowKey
-----------	--

	<ul style="list-style-type: none"> • ModifyEntity: Ändert eine Entität und wählt zwischen einer Zusammenführen- und einer Ersetzen-Operation mit der Parameterprüfung <code>replaceExistingItems</code>. Wenn die Ersetzen-Operation ausgewählt ist (<code>replaceExistingItems="true"</code>), dann bleiben nach der Operation nur die angegebenen Eigenschaften erhalten, alle anderen Eigenschaften werden gelöscht. Wenn die Zusammenführen-Operation gewählt ist (<code>replaceExistingItems="false"</code>), dann bleiben alle Eigenschaften erhalten und die bereits vorhandenen werden aktualisiert. Erforderliche Parameter: <code>tableName</code>, <code>partitionKey</code>, <code>rowKey</code>, <code>replaceExistingItems</code> • QuerySubsetOfEntityProperties: Ruft eine Teilmenge von Eigenschaften der Entitäten gemäß eines XML-Inputs ab. Wenn <code>rowKey</code> und <code>partitionKey</code> angegeben sind, dann werden nur die passenden Eigenschaften der Entitäten abgerufen. Wenn weder <code>rowKey</code> noch <code>partitionKey</code> angegeben sind, dann werden alle Eigenschaften der Tabellenentitäten abgerufen. Die Ausgabe dieser Operation ist eine XML-Datei, die die Teilmenge der Eigenschaften innerhalb eines übergeordneten Elements für jede zugehörige Entität enthält. Erforderliche Parameter: <code>tableName</code>, <code>partitionKey</code>, <code>rowKey</code>
--	---

Parameter

Adapter	Adapter Hauptklasse (nicht ändern!) Mögliche Werte: <code>de.softproject.integration.adapter.microsoft.azure.table.MicrosoftAzureTableStorageAdapter</code> : Hauptklasse (Standard)
accountName	Name des Microsoft Azure Kontos Mögliche Werte: Beliebige gültige Zeichenkette, z. B. <code>softproject</code>
accountKey	Passwort zum Microsoft Azure Konto Mögliche Werte: Beliebige Zeichenkette mit dem von Microsoft Azure bereitgestellten Schlüssel, z. B. <code>AD9ohFhJ6Hve90DJEXTw53MjR1tjoUZ8lwFDlw5UFqGMih3VtUCapIwIfEMBmITeShQN9MbCITE4SzxTcGIybw==</code>
endpoint	Azur Cosmos DB Endpunkt
tableName	Name der Tabelle, in der die Entitäten gespeichert sind Mögliche Werte: Zeichenkette mit der ID für die neue Tabelle  Tabellen-Namen müssen die gleichen Namenskonventionen erfüllen wie Datenbank-IDs. Datenbank-Namen müssen zwischen 1 und 255 Zeichen lang sein und dürfen weder <code>/</code> <code>\</code> <code>#</code> <code>?</code> noch ein Leerzeichen enthalten.

partitionKey	<p>Element des Primärschlüssels</p> <p>Entitäten mit dem gleichen Partitionsschlüssel können schneller abgefragt und in atomaren Operationen eingefügt/aktualisiert werden.</p> <p>Mögliche Werte: Beliebige Zeichenkette, die den Partitionsschlüssel enthält</p>
rowKey	<p>Element des Primärschlüssels</p> <p>Der Zeilenschlüssel einer Entität ist ihre eindeutige Kennung innerhalb einer Partition.</p> <p>Mögliche Werte: Beliebige Zeichenkette, die den Zeilenschlüssel enthält</p>
filterCondition	<p>Vergleicht partitionKey und rowKey</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • EQUAL • GREATER_THAN • GREATER_THAN_OR_EQUAL • LESS_THAN • LESS_THAN_OR_EQUAL • NOT_EQUAL
replaceExistingItems	<p>Gibt an, ob ModifyEntity eine Zusammenführen- oder Ersetzen-Operation ausführt. Weitere Informationen auf http://azure.github.io/azure-storage-java/.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • true: Ersetzen-Operation ausführen • false: Zusammenführen-Operation ausführen

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt
-1	Bei der Ausführung des Adapters kam es zu einem Fehler
0	<p>Operation CreateTable: Die Tabelle konnte nicht angelegt werden, aber es wurde weder ein Fehler noch eine Exception zurückgegeben.</p> <p>Operation DeleteTable: Er angegebene tableName existiert nicht.</p>

Input

Der Adapter erwartet ein XML-Dokument als Input.

- Für die InsertEntity-Operation wird eine XML-Datei mit folgender Struktur erwartet:

Erwartete Struktur für die Operation InsertEntity

```
<?xml version="1.0" encoding="UTF-8"
<Entity>
  <Properties1 name="Department" value="ITAdministration"/>
  <Properties2 name="Address" value="C\Pirandello"/>
  <Properties3 name="Name" value="Paolo"/>
  <Properties4 name="Surname" value="Green"/>
</Entity>
```

 Die Eigenschaft name darf kein geschütztes Wort wie beispielsweise *Id* sein.

- Für die ModifyEntity-Operation wird eine XML-Datei mit folgender Struktur erwartet, Properties mit gleichem Namen und Wert werden aktualisiert:

Erwartete Struktur für die Operation ModifyEntity

```
<?xml version="1.0" encoding="UTF-8" ?>
<Entity>
  <Properties name="Name" value="Carlos" />
</Entity>
```

- Für die QueryOfSubsetEntityProperties-Operation wird eine XML-Datei erwartet, in der die erforderlichen Properties spezifiziert sind:

Erwartete Struktur für die Operation QueryOfSubsetEntityProperties

```
<?xml version="1.0" encoding="UTF-8" ?>
<Entity>
  <Property name="Department" />
  <Property name="Surname" />
  <Property name="nonExistingProperty" />
</Entity>
```

Ausgabe

Der Adapter gibt je nach ausgeführter Operation unterschiedliche Arten von Dokumenten zurück:

- Ausgabe der RetrieveEntity-Operation:

Ausgabe für die Operation RetrieveEntity

```
<?xml version="1.0" encoding="UTF-8"
<Entity partitionKey="ITEmployee" rowKey="ref-0001">
  <Properties name="Department" value="ITAdministration"/>
  <Properties name="Address" value="C\Pirandello"/>
  <Properties name="Surname" value="Green"/>
  <Properties name="Name" value="Paolo"/>
</Entity>
```

- Ausgabe der RetrieveAllEntities- und RetrieveRangeEntities-Operation:


Ausgabe für die Operation RetrieveAllEntities und RetrieveRangeEntities

```
<?xml version="1.0" encoding="UTF-8" ?>
<ListEntities>
  <Entity partitionKey="ITEmployee" rowKey="ref-0001">
    <Properties name="Department" value="ITAdministration" />
    <Properties name="Address" value="C\Pirandello" />
    <Properties name="Surname" value="Green" />
    <Properties name="Name" value="Paolo" />
  </Entity>
  <Entity partitionKey="ITEmployee" rowKey="ref-0002">
    <Properties name="Department" value="Developments" />
    <Properties name="Address" value="C\Francisco" />
    <Properties name="Surname" value="Lopez" />
    <Properties name="Name" value="Luis" />
  </Entity>
  <Entity partitionKey="ITEmployee" rowKey="ref-0003">
    <Properties name="Department" value="Sales" />
    <Properties name="Address" value="C\Espada" />
    <Properties name="Surname" value="Red" />
    <Properties name="Name" value="Paula" />
  </Entity>
</ListEntities>
```

- Ausgabe der QueryOfSubsetEntityProperties-Operation:

Ausgabe für die Operation QueryOfSubsetEntityProperties

```
<?xml version="1.0" encoding="UTF-8" ?>
<ListEntities>
  <Entity>
    <Property partitionKey="ITEmployee" />
    <Property rowKey="ref-0001" />
    <Property name="Department" value="ITAdministration" />
    <Property name="nonExistingProperty" />
    <Property name="Surname" value="Green" />
  </Entity>
</ListEntities>
```

 Die beiden Properties `partitionKey` und `rowKey` werden immer zurückgegeben, da mit ihnen die Entität eindeutig identifiziert werden kann. Da die Operation in der Lage ist, mehrere Properties von mehr als einem Objekt zurückzugeben, ist es wichtig, die Eigenschaften der dazugehörigen Entität zuzuordnen.

6.4 Communication

6.4.1 AS/400 PCML



AS/400 PCML: Ruft Programme auf AS/400-Systemen (*IBM eServer iSeries* bzw. *System i*) über die *Program Call Markup Language* (PCML) auf und gibt die Ergebnisse des ausgeführten Programms als XML-Dokument zurück.

Ehemalige Bezeichnung bis Version 4.5.1: *AS/400 PCML Connection Adapter*

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>Execute</i> : Funktions-Adapter ausführen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.PCMLAdapter</i> : Hauptklasse (Standard)
<i>systemName</i>	Host-Name des AS/400-Systems Mögliche Werte: Beliebiger Hostname (z. B. <i>localhost</i>)
<i>userId</i>	Benutzername für die Server-Authentifizierung Mögliche Werte: <ul style="list-style-type: none"> • Beliebige Zeichenkette • <i>*CURRENT</i>: aktuelle Benutzer-ID aufrufen
<i>password</i>	Passwort für die Server-Authentifizierung Mögliche Werte: Beliebige Zeichenkette

<i>encoding</i>	Zeichenkodierung des Input- und Output-XML-Dokuments Mögliche Werte: Beliebige gültige Zeichenkodierung (z. B. <i>UTF-8</i>)
-----------------	---

Input

Dieser Adapter erwartet eine adapterspezifische Input-XML-Struktur, wobei jedes Element `<program>` einem Programm entspricht, jedes Element `<data>` darin ein Feld definiert und jedes Element `<structure>` als Datenstruktur-Element oder als ein Argument für ein Programm dienen kann:

```
<pcml version="1.0">
  <program>
    <data></data>
    <struct>
      <data></data>
    </struct>
  </program>
</pcml>
```

Ausgabe

Der erfolgreich ausgeführte PCML-Adapter gibt ein XML-Dokument mit folgendem Muster zurück:

```
<pcml>
  <program_name>
    <data_values></data_values>
    <struct_name1></struct_name1>
  </program_name>
</pcml>
```

6.4.2 File System



File System: Liest, löscht oder listet Dateien und Ordner, die sich im Dateisystem des X4 Servers befinden und speichert Dateien und Ordner im Dateisystem. Ordnerinhalte können über die Operation `Send` in Kombination mit der Eigenschaft `Delete` verschoben werden.

⚠ Der Adapter *File System* darf nicht verwendet werden, um Dateien im X4 Repository zu verändern, da er nicht auf das X4 Repository, sondern direkt auf das Dateisystem zugreift. Wird der Adapter hierfür verwendet, kann es zu Fehlermeldungen und Störungen kommen. Um Dateien im X4 Repository zu verändern, verwenden Sie den Adapter [X4 Projects](#) bzw. [Project Access](#).



- Der Adapter *File System* schreibt mit der Operation *Send* Dateien auf ein gewünschtes Verzeichnis. Dabei wird im Ziel-Verzeichnis eine temporäre Datei (.tmp) erstellt und so die *Create*-Funktion angewendet. Solange das Schreiben noch aktiv ist, wird der Zugriff für mögliche Verwender gesperrt und erst nach vollständiger Erstellung erfolgt ein *Rename* auf die Datei und das Suffix .tmp wird entfernt. Ein Benutzer, der jedoch nur auf *Create*-Dateien prüft, wird somit die Datei nicht mehr verarbeiten. Als Alternative kann der Adapter *File System Bulk* verwendet werden, da dieser nicht mit temporären Dateien arbeitet.
- Will man mit dem Adapter *File System* prüfen, ob eine bestimmte Datei in einem Verzeichnis vorhanden ist, sollte nicht die Operation *Receive* verwendet werden, da im X4 Server ein Fehler (Status 0) ausgegeben wird aus dem ersichtlich ist, ob die Datei fehlt oder nicht lesbar ist. Stattdessen kann über die Operation *List* und dem gewünschten Dateinamen im Parameter *filter* nach der Datei gesucht werden.

Eigenschaften (Properties)

I/O Mode <i>(input_type)</i>	Art der Quelle der zu verarbeitenden Daten Mögliche Werte: <ul style="list-style-type: none"> • Actions: Einzelne Datei, deren Name in Eigenschaft <code>File name</code> angegeben wird, verarbeiten • Directory: Alle Dateien eines angegebenen Ordners verarbeiten (oder Dateien gemäß Filterregel in Parameter <code>filter</code>)
File name <i>(documentname)</i>	Name der zu verarbeitenden Datei (wenn I/O Mode auf <i>Actions</i> gesetzt ist). <div data-bbox="483 707 1434 1030" style="border: 1px solid #add8e6; padding: 10px;"> <p>i Wenn Sie diese Eigenschaft dynamisch parametrieren möchten, müssen Sie im zugehörigen XSL-Mapping deren technischen Namen <i>documentname</i> verwenden z.B.</p> <div data-bbox="560 857 1410 1010" style="border: 1px solid #add8e6; padding: 10px; margin: 10px 0;"> <pre><xsl:processing-instruction name="DynamicParameter"> documentname=" <xsl:value-of select="."/"/>" </xsl:processing-instruction></pre> </div> </div> Mögliche Werte: <ul style="list-style-type: none"> • Beliebiger Dateiname z.B. <i>Data.xml</i> • Mehrere Dateinamen, durch Semikolon getrennt <div data-bbox="526 1182 1434 1377" style="border: 1px solid #add8e6; padding: 10px; margin-top: 10px;"> <p>i Werden über <code>Fork/Join</code> mehrere Input-Dokumente übergeben, können auch mehrere Dateien gemäß der Priorität der Verzweigungen nacheinander verarbeitet werden z.B. <i>File1.xml;File2.xml</i>.</p> </div>
Delete <i>(delete)</i>	Quelldatei(en) nach erfolgreich abgeschlossener Verarbeitung im Dateisystem löschen

Operation <i>(action)</i>	<p>Bestimmt, welche Dateisystem-Operation mit den verarbeiteten Dateien durchgeführt wird. Je nach Wert in <i>I/O Mode</i> unterscheidet sich die Funktionsweise.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Delete: Datei löschen, deren Dateiname in Eigenschaft <i>Filename</i> festgelegt wird oder sämtliche Dateien gemäß Filterregel in Parameter <i>filter</i> • List: Liste der Dateien des in Parameter <i>path</i> angegebenen Ordners als XML-Dokument ausgeben • Make directory: Ordner erzeugen, dessen Name in Eigenschaft <i>Filename (documentname)</i> festgelegt wird • Delete directory: Ordner löschen, dessen Name in Eigenschaft <i>Filename (documentname)</i> festgelegt wird • Send: <ul style="list-style-type: none"> • <i>I/O Mode Directory:</i> Alle Dateien von Parameter <i>dir</i> nach <i>path</i> kopieren, die <i>filter</i> entsprechen • <i>I/O Mode Action:</i> Daten aus dem vorangegangenen Prozess-Schritt in den in Parameter <i>path</i> angegebenen Ordner speichern, dabei den in Eigenschaft <i>Filename</i> festgelegten Namen verwenden. • Append: <ul style="list-style-type: none"> • <i>I/O Mode Directory:</i> Wie <i>Send</i>, allerdings werden die Inhalte der gesendeten Dateien den Zielformateien angehängt, wenn bereits gleichnamige Zielformateien besteht • <i>I/O Mode Action:</i> Wie <i>Send</i>, allerdings werden die gesendeten Daten der Zielformatei angehängt, wenn bereits eine gleichnamige Zielformatei besteht. • Receive: <ul style="list-style-type: none"> • <i>I/O Mode Directory:</i> Alle Dateien, die <i>filter</i> entsprechen, von Parameter <i>path</i> nach <i>dir</i> kopieren • <i>I/O Mode Action:</i> Datei in den Prozess einlesen, deren Pfad in Parameter <i>path</i> und deren Dateiname in Eigenschaft <i>Filename</i> festgelegt wurde. Wenn in Parameter <i>filter</i> ein Filter gesetzt ist, wird stattdessen der erste Treffer als Dateiname verwendet.
-------------------------------------	--

Parameter

path	<p>Pfad im Dateisystem des X4 Servers oder zu einem verbundenen Windows- oder Samba-Netzlaufwerk, von dem gelesen bzw. in das geschrieben werden soll; Hierzu benötigt der X4 Server je nach Operation entsprechende Lese-/Schreibberechtigungen auf den Zielordner.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige Zeichenkette, ggf. inklusive Platzhalter z.B. <code>%USER_COLLECTION%\Data\</code> • <code>c:\</code>: Auf die Windows-Systempartition C:\ zugreifen (Standard) • Beliebige UNC-Netzlaufwerksangabe nach dem Schema <code>\\Server\Freigabe\Ordner\</code> mit beliebigen Leer- und Sonderzeichen z.B. <code>\\SP-PM-01\share\Öffentlicher Ordner\</code>
filter	<p>Nach Dateinamen filtern über die Platzhalter * und ?; Um mehrere Filterausdrücke zu definieren, als Trennzeichen ein Semikolon (;) verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebiger Dateiname mit oder ohne Platzhalter, so gilt beispielsweise <code>*.dat</code> nur für Dateien mit der Dateiendung <code>*.dat</code> • <code>*</code>: Alle Dateien (kein Filter) (Standard)
dir	<p>Lokaler Ordner für Dateisystem-Operationen; je nach Operation Quellordner oder Zielordner der Dateisystem-Operation</p> <p>Mögliche Werte: Beliebige Zeichenkette z.B. <code>%USER_COLLECTION%\X4Experience</code></p>
max	<p>Anzahl der zu verarbeitenden Dateien begrenzen (wenn I/O Mode auf Directory gesetzt ist und somit mehrere Dateien eines Ordners verarbeitet werden sollen)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • <code>0</code>: keine Begrenzung der Anzahl • <code>1</code>: maximal eine Datei verarbeiten (Standard)
encoding	<p>Zeichensatz für verarbeitete Text-Dateien setzen (veraltet seit X4 Server-Version 3.8; Stattdessen Charset Changer verwenden); Nicht für Binärdokumente verwenden!</p>


6.4.3 File System Bulk




File System Bulk: Vereinfacht die Nutzung von Datei-Operationen in technischen Prozessen. Mit ihm lassen sich einfach Aktionen wie das Auflisten, Verschieben und Kopieren von mehreren Dateien oder Verzeichnissen durchführen. Durch die Kompatibilität mit dem [ZIP in Memory Compression \(Veraltet\)](#) Adapter lassen sich auch einfach multiple Dateien und Verzeichnisse (ent-)packen.

Eigenschaften (Properties)

Operation	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • Read: Dateien auslesen • Write: Dateien und Verzeichnisse erzeugen/ersetzen • Delete: Dateien und Verzeichnisse löschen • GetMetadata: Metadaten von Dateien und Verzeichnissen auslesen • Copy: Dateien und Verzeichnisse kopieren • Move: Dateien und Verzeichnisse verschieben • List: Dateien und Verzeichnisse listen • ListAndRead: Dateien und Verzeichnisse listen und den Inhalt der Dateien gleich mit auslesen. (Kombiniert List mit Read) • ListWithMetadata: Dateien und Verzeichnisse listen und die Metadaten gleich mit auslesen. (Kombiniert List mit GetMetadata)
-----------	---

 Die Input- und Output-Strukturen sind kompatibel mit denen des [ZIP in Memory Compression \(Veraltet\)](#) Adapters.

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <code>de.softproject.integration.adapter.filesystem.Bulk</code> : Hauptklasse (Standard)
<i>baseDirectory</i>	Basis-Verzeichnis für alle Operationen  Alle Operationen können nur in diesem Verzeichnis und seinen Unterverzeichnissen arbeiten. Mögliche Werte: Absoluter Pfad zum Basis-Verzeichnis

<i>subDirectory</i>	<p>Verzeichnis relativ zum <code>baseDirectory</code> als Basis für alle Operationen.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code><leer></code>: die Operationen starten im <code>baseDirectory</code> • <code>..</code>: die Operationen starten im <code>baseDirectory</code> • <code>a</code>: die Operationen starten in <code><baseDirectory>/a</code> • <code>..</code> oder <code>a/..</code>: Fehler, das <code>baseDirectory</code> kann auf diesem Weg nicht verlassen werden
<i>limit</i>	<p>Begrenz die Anzahl der zurückgegebenen Treffer einer List-Operation.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code><positive Zahl></code>: maximale Anzahl der zurückgelieferten Treffer • <code>0</code>: Ergebnis ist eine leere Liste • <code><negative Zahl></code>: maximale Treffer-Anzahl ist unbegrenzt <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> Bei der Suche nach Dateien wird empfohlen, den Parameter <code>skipDirectories</code> auf <code>true</code> zu setzen, da sonst die gewünschten Treffer von Verzeichnissen verdrängt werden könnten.</p> </div>
<i>maxLevel</i>	<p>Beschränkt die Rekursionstiefe für List-Operationen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code><positive Zahl></code>: maximale Tiefe, die eine List-Operation in Verzeichnisse hinabsteigt • <code>1</code>: nur der direkte Inhalt des Verzeichnisses, das durch <code>baseDirectory/subDirectory</code> angegeben ist • <code>0</code>: Ergebnis ist eine leere Liste • <code><negative Zahl></code>: die Rekursionstiefe ist unbegrenzt
<i>filter</i>	<p>Regulärer Ausdruck, der auf den relativen Pfad eines Eintrags bei List-Operationen angewendet wird. Wenn der reguläre Ausdruck zutrifft, wird der Pfad in die Ergebnismenge der Liste übernommen.</p>
<i>skipDirectories</i>	<p>Definiert, ob eine List-Operation Verzeichnis-Elemente im Ergebnis ausgibt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>true</code>: Verzeichnisse werden nicht in die Ergebnismenge von List-Operationen übernommen (nützlich in Verbindung mit <code>limit</code>) • <code>false</code>: Verzeichnisse werden in die Ergebnismenge von List-Operationen übernommen

<i>overwrite</i>	<p>Definiert, ob bei den Operationen <code>Write</code>, <code>Move</code> oder <code>Copy</code> vorhandene Dateien überschrieben werden dürfen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Bestehende Dateien dürfen bei <code>Write</code>, <code>Copy</code> und <code>Move</code> überschrieben werden. • <i>false</i>: Bestehende Dateien dürfen nicht überschrieben werden.
<i>strict</i>	<p>Definiert, ob für <code>Write</code>-Operationen ein Elternverzeichnis vorhanden sein muss und ob das Verzeichnis für die Operation <code>Delete</code> leer sein muss.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: <ul style="list-style-type: none"> • Bei <code>Copy</code> und <code>Move</code> muss das Zielverzeichnis existieren, in welches hineinkopiert wird. • Bei <code>Delete</code> müssen die Verzeichnisse leer sein. • Bei <code>Write</code> muss das Zielverzeichnis existieren, in das hineingeschrieben wird. • <i>false</i>: Die Elternverzeichnisse werden für <code>Write</code>-Operationen angelegt. Zudem werden Verzeichnisse auch gelöscht, wenn sie noch nicht leer sind.

Statuswerte

1	Alle Einträge für die Operation konnten erfolgreich durchgeführt werden.
0	Nicht alle Einträge für die Operation konnten erfolgreich durchgeführt werden, d.h. das Ergebnis enthält Error-Elemente.
-	Bei der Ausführung kam es zu anderen technischen Problemen, z.B. ungültiger Input.
1	

Input

Input-Struktur für Read, Write, GetMetadata und Delete


```
<any>
  <!-- any mixture of file and directory elements -->
  <File name="relativePath">...Base64 content for Write...</File>
  <Directory name="relativePath" />
</any>
```

Input-Struktur für Copy und Move

```
<any>
  <!-- any mixture of file and directory elements -->
  <File name="relativePath" target="relativePath" />
  <Directory name="relativePath" target="relativePath" />
</any>
```

Output

```
<Data>
  <!-- any mixture of file, directory and error elements -->
  <File name="relativePath" [size="size of file in bytes"]
    [creationTime=""] [creationTimeAsMillis=""]
    [lastModifiedTime=""] [lastModifiedTimeAsMillis=""]
    [lastAccessTime=""] [lastAccessTimeAsMillis=""]>
    <!-- base64 conten, if operation Read or ListAndRead -->
    <!-- xml structures containing metadata, if operation GetMetadata or
ListWithMetadata -->
  </File>
  <Directory name="relativePath" [size="0"]
    [creationTime=""] [creationTimeAsMillis=""]
    [lastModifiedTime=""] [lastModifiedTimeAsMillis=""]
    [lastAccessTime=""] [lastAccessTimeAsMillis=""]>
    <!-- xml structures containing metadata, if operation GetMetadata or
ListWithMetadata -->
  </Directory>
  <Error name="relativePath" [target="relativePath"] exception="fully qualified
classname of exception"
    [size="size of file in bytes"]
    [creationTime=""] [creationTimeAsMillis=""]
    [lastModifiedTime=""] [lastModifiedTimeAsMillis=""]
    [lastAccessTime=""] [lastAccessTimeAsMillis=""]>
    <!-- base64 content if operation Write -->
  </Error>
</Data>
```

 Write, Delete, Copy und Move haben nur dann Error-Elemente als Ergebnis, wenn es tatsächlich Fehler gab.

Beispiel

Im Demosystem *X4 Experience* finden Sie den Beispielprozess `FilesystemBulk.wrf` zur Verwendung des Adapters.

6.4.3.1 Beispiele für List

6.4.3.1.1 Voraussetzung

Folgende Verzeichnisstruktur ist gegeben:

```

/
+ a/
| + index.html
| + logo.gif
| + content/
|   + toc.html
|   + page1.html
|   + page2.html
|   + images/
|     + pic1.png
|     + pic2.png
+ b/
  + some.html
  + other.html
  + files.html

```

6.4.3.1.2 Szenarien

6.4.3.1.2.1 Alles listen

Adapter-Konfiguration	Output
<ul style="list-style-type: none"> • baseDirectory = /a • subDirectory = • limit = -1 • maxLevel = -1 • filter = • skipDirectories = false 	<pre> <Data> <File name="index.html" ... /> <File name="logo.gif" ... /> <Directory name="content" ... /> <File name="content/toc.html" ... /> <File name="content/page1.html" ... /> <File name="content/page2.html" ... /> <Directory name="content/images" ... /> <File name="content/images/pic1.png" ... /> <File name="content/images/pic2.png" ... /> </Data> </pre>

6.4.3.1.2.2 Die ersten 5 Einträge

Adapter-Konfiguration	Output
<ul style="list-style-type: none"> • baseDirectory = /a • subDirectory = • limit = 5 • maxLevel = -1 • filter = • skipDirectories = false 	<pre> <Data> <File name="index.html" ... /> <File name="logo.gif" ... /> <Directory name="content" ... /> <File name="content/toc.html" ... /> <File name="content/page1.html" ... /> </Data> </pre>

6.4.3.1.2.3 Die ersten fünf Dateien

Adapter-Konfiguration	Output
<ul style="list-style-type: none"> • baseDirectory = /a • subDirectory = • limit = 5 • maxLevel = -1 • filter = • skipDirectories = true 	<pre> <Data> <File name="index.html" ... /> <File name="logo.gif" ... /> <File name="content/toc.html" ... /> <File name="content/page1.html" ... /> <File name="content/page2.html" ... /> </Data> </pre>

6.4.3.1.2.4 Nur das aktuelle Verzeichnis

Adapter-Konfiguration	Output
<ul style="list-style-type: none"> • baseDirectory = /a • subDirectory = • limit = -1 • maxLevel = 1 • filter = • skipDirectories = false 	<pre> <Data> <File name="index.html" ... /> <File name="logo.gif" ... /> <Directory name="content" ... /> </Data> </pre>

6.4.3.1.2.5 Alle Dateien und Verzeichnisse, die einen Filter erfüllen

Adapter-Konfiguration	Output
<ul style="list-style-type: none"> • baseDirectory = /a • subDirectory = • limit = -1 • maxLevel = -1 • filter = tent • skipDirectories = false 	<pre> <Data> <Directory name="content" ... /> <File name="content/toc.html" ... /> <File name="content/page1.html" ... /> <File name="content/page2.html" ... /> <Directory name="content/images" ... /> <File name="content/images/pic1.png" ... /> <File name="content/images/pic2.png" ... /> </Data> </pre>

6.4.3.1.2.6 Alle Dateien, die einen Filter erfüllen (ohne Verzeichnisse)

Adapter-Konfiguration	Output
<ul style="list-style-type: none"> • baseDirectory = /a • subDirectory = • limit = -1 • maxLevel = -1 • filter = tent • skipDirectories = true 	<pre> <Data> <File name="content/toc.html" ... /> <File name="content/page1.html" ... /> <File name="content/page2.html" ... /> <File name="content/images/pic1.png" ... /> <File name="content/images/pic2.png" ... /> </Data> </pre>

6.4.3.1.2.7 Alles unterhalb eines Unterverzeichnisses

Adapter-Konfiguration	Output
<ul style="list-style-type: none"> • baseDirectory = /a • subDirectory = content • limit = -1 • maxLevel = -1 • filter = • skipDirectories = false 	<pre> <Data> <File name="toc.html" ... /> <File name="page1.html" ... /> <File name="page2.html" ... /> <Directory name="images" ... /> <File name="images/pic1.png" ... /> <File name="images/pic2.png" ... /> </Data> </pre>

6.4.3.1.2.8 Alles unterhalb des aktuellen Verzeichnisses und den direkten Kindverzeichnissen

Adapter-Konfiguration	Output
<ul style="list-style-type: none"> • baseDirectory = /a • subDirectory = • limit = -1 • maxLevel = 2 • filter = • skipDirectories = false 	<pre> <Data> <File name="index.html" ... /> <File name="logo.gif" ... /> <Directory name="content" ... /> <File name="content/toc.html" ... /> <File name="content/page1.html" ... /> <File name="content/page2.html" ... /> <Directory name="content/images" ... /> </Data> </pre>

6.4.3.1.2.9 Das Startverzeichnis kann tief im Verzeichnisbaum liegen

Adapter-Konfiguration	Output
<ul style="list-style-type: none"> • baseDirectory = /a • subDirectory = content/images • limit = -1 • maxLevel = -1 • filter = • skipDirectories = false 	<pre><Data> <File name="pic1.png" ... /> <File name="pic2.png" ... /> </Data></pre>

6.4.4 FTP

6.4.4.1 FTP Connector



FTP Connector: Unterstützt die Protokolle FTP (*File Transfer Protocol*), SFTP (*SSH File Transfer Protocol*) und FTPS (*File Transfer Protocol Secure* bzw. *FTP-SSL*) zum Senden, Empfangen, Löschen und Verschieben von einer oder mehreren Dateien und Ordnern sowie zum Auflisten von Ordnerinhalten. Der Adapter unterstützt Proxy-Server vom Typ HTTP, SOCKS4 und SOCKS5.

Ehemalige Bezeichnung bis Version 4.5.1: *File Transfer Adapter*




Dieser Funktions-Adapter ersetzt den *FTP Transfers* sowie den *SFTP Communication* Adapter.

Eigenschaften (Properties)

<i>Operation (action)</i>	<p>Bestimmt, welche Dateitransfer-Operation mit den verarbeiteten Dateien oder Ordnern durchgeführt wird.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • List: Liste der Dateien des in Parameter <code>remotePath</code> angegebenen Ordners als XML-Dokument ausgeben • Send: Daten aus dem vorangegangenen Prozess-Schritt in den in Parameter <code>remotePath</code> angegebenen Ordner speichern, dabei den in Parameter <code>documentName</code> festgelegten Dateinamen verwenden. Oder mehrere Dateien von Quellordner in Parameter <code>localPath</code> an <code>remotePath</code> versenden, indem die Parameter <code>directoryMode</code> und <code>multipleFiles</code> auf <code>true</code> gesetzt werden; in Parameter <code>filter</code> kann zudem ein Filterausdruck angegeben werden. • Receive: Datei in den Prozess einlesen, deren Pfad in Parameter <code>remotePath</code> und deren Dateiname in Parameter <code>documentName</code> festgelegt wurde. Oder mehrere Dateien empfangen, indem die Parameter <code>directoryMode</code> und <code>multipleFiles</code> auf <code>true</code> und <code>localPath</code> auf den lokalen Zielordner gesetzt werden; in Parameter <code>filter</code> kann zudem ein Filterausdruck angegeben werden. • Delete: In Parameter <code>documentName</code> festgelegte Datei oder Ordner auf dem FTP(S)/SFTP-Server löschen, deren Pfad in Parameter <code>remotePath</code> festgelegt wird. Um nicht-leere Ordner zu löschen, zusätzlich Parameter <code>recurse</code> aktivieren. • Move: In Parameter <code>documentName</code> festgelegte Datei oder Ordner innerhalb des FTP(S)/SFTP-Servers von <code>oldPath</code> nach <code>remotePath</code> verschieben • Make directory: Ordner auf dem FTP(S)/SFTP-Server erzeugen, dessen Name in Parameter <code>documentName</code> und dessen Pfad in Parameter <code>remotePath</code> festgelegt wird • Delete directory: Ordner auf dem FTP(S)/SFTP-Server löschen, dessen Name in Parameter <code>documentName</code> und dessen Pfad in Parameter <code>remotePath</code> festgelegt wird
---------------------------	---

Parameter



<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.transferadapter.TransferAdapter</code>; Hauptklasse (Standard)</p>
----------------	--

<i>protocol</i>	<p>Dateitransfer-Protokoll</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>FTP</i>: Operationen über das <i>File Transfer Protocol</i> ausführen (Standard) • <i>FTPS</i>: Operationen über das <i>File Transfer Protocol Secure</i> ausführen • <i>SFTP</i>: Operationen über das <i>SSH File Transfer Protocol</i> ausführen
<i>host</i>	<p>IP-Adresse oder Host-Name des FTP(S)/SFTP-Servers</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>server.tld</i>)</p>
<i>port</i>	<p>Port-Nummer des FTP(S)/SFTP-Servers</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 22: SFTP-Standardport (Standard)
<i>connectMode</i>	<p>Verbindungsmodus</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ACTIVE</i>: Aktiver Verbindungsmodus • <i>PASSIVE</i>: Passiver Verbindungsmodus (standard)
<i>controlEncoding</i>	<p>Verwendete Kodierung für den Kommando-Kanal. Hiermit wird beeinflusst wie Umlaute in Dateinamen übertragen werden. Dieser Wert ist abhängig vom FTP-Server.</p> <p>Mögliche Werte: UTF-8 (Standard)</p> <p> Ist kein <code>controlEncoding</code> gesetzt oder das <code>controlEncoding</code> entspricht dem Leerstring, wird die Verbindung nicht modifiziert und das <code>controlEncoding</code> ist undefiniert. Sonst wird der Wert als <code>controlEncoding</code> auf die Verbindung gesetzt.</p>
<i>login</i>	Benutzername für die FTP(S)/SFTP-Verbindung
<i>password</i>	Passwort für die FTP(S)/SFTP-Verbindung; Wenn Parameter <code>privateKeyFile</code> gesetzt ist: Passwort des <i>Private Keys</i>
<i>accountInfo</i>	Zusätzliche Informationen zum Konto, siehe auch Dokumentation der FTP-Bibliothek

<i>loginCommandsUrl</i>	<p>URL zu der im XStore liegenden XML-Datei, welche die Login-Commands enthält</p> <p>Mögliche Werte: Beliebige XStore-URL</p>
<i>remotePath</i>	<p>Pfad auf dem FTP(S)/SFTP-Server, dessen Zielordner per FTP(S)/SFTP manipuliert werden soll</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <i>Folder/Subfolder</i>)</p>
<i>documentname</i>	<p>Name des Ordners oder der Datei, der/die verarbeitet wird</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <i>Data.xml</i>)</p>
<i>oldPath</i>	<p>Pfad zum Ordner oder zur Datei auf dem FTP(S)/SFTP-Server, der/die verschoben werden soll (nur bei Operation Move)</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <i>Folder/Subfolder</i>)</p>
<i>oldDocumentName</i>	<p>Name der Datei, die verschoben werden soll (nur bei Operation Move)</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <i>Data.xml</i>)</p>
<i>recurse</i>	<p>Pfad rekursiv löschen, d.h. inklusive aller Unterordner und Dateien (nur bei Operation Delete)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • True: Alle beinhalteten Ordner (und Unterordner) und Dateien löschen • False: Nur Ordner und Dateien im aktuellen Pfad löschen (Standard)
<i>directoryMode</i>	<p>Ordner-Modus aktivieren, um den Inhalt von Ordnern zu verarbeiten; falls Parameter <i>multipleFiles</i> aktiviert ist, werden alle Elemente verarbeitet, die dem Filterkriterium in Parameter <i>filter</i> entsprechen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • true: Ordner verarbeiten • false: Nur einzelne Dateien verarbeiten (Standard)

<i>localPath</i>	<p>Lokaler Ordner zum Senden oder Empfangen von mehreren Dateien (nur wenn die Parameter <code>directoryMode</code> und <code>multipleFiles</code> aktiviert sind)</p> <p>Mögliche Werte: Beliebige gültige Pfadangabe (z.B. <code>C:\Directory</code>)</p>
<i>multipleFiles</i>	<p>Alle Dateien verarbeiten, die dem Filterkriterium in Parameter <code>filter</code> entsprechen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>true</code>: Alle zutreffenden Dateien verarbeiten • <code>false</code>: Nur die Datei verarbeiten, die in Parameter <code>documentName</code> angegeben wurde, oder wenn Parameter <code>directoryMode</code> aktiviert ist, die erste Datei (falls das Filterkriterium auf mehrere Dateien zutrifft) (Standard)
<i>filter</i>	<p>Filterausdruck, der die Platzhalter <code>*</code> und <code>?</code> enthalten kann, um die Menge der verarbeiteten Dateien einzuschränken. Dies gilt für die Operation <code>list</code> und die Operationen <code>receive</code> und <code>send</code> (nur wenn die Parameter <code>multipleFiles</code> und <code>directoryMode</code> aktiviert sind).</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <code>*.xml</code>)</p>
<i>deleteAfterReceive</i>	<p>Datei oder Ordner nach dem Empfangen vom FTP(S)/SFTP-Server löschen (nur bei Operation <code>Receive</code>)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>true</code>: Ursprüngliche Datei oder Ordner nach dem Empfangen auf dem FTP(S)/SFTP-Server löschen • <code>false</code>: Ursprüngliche Datei oder Ordner behalten (Standard)
<i>deleteAfterSend</i>	<p>Datei oder Ordner nach dem Senden zum FTP(S)/SFTP-Server löschen (nur bei Operation <code>Send</code>)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>true</code>: Ursprüngliche Datei oder Ordner nach dem Senden zum FTP(S)/SFTP-Server löschen • <code>false</code>: Ursprüngliche Datei oder Ordner behalten (Standard)

<i>append</i>	<p>Gesendete Daten an die Zielfeile anhängen, wenn bereits eine gleichnamige Zielfeile besteht (nur bei Operation Send)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>true</code>: Daten in der Zielfeile (falls vorhanden) anhängen • <code>false</code>: Zielfeile (falls vorhanden) überschreiben (Standard)
<i>transferMode</i>	<p>Übertragungsart</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>BINARY</code>: Als Binärdatei übertragen (Standard) • <code>TEXT</code>: Als Text übertragen
<i>maxTransferRate</i>	<p>Maximale Übertragungsrate in Bytes/Sekunde</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • <code>0</code>: keine Begrenzung (Standard)
<i>sleepEnabled</i>	<p>Ruhemodus (<i>Sleep Mode</i>) aktivieren (wenn Parameter <code>multipleFiles</code> aktiviert ist)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>true</code>: Ruhemodus aktivieren (ermöglicht die Definition einer Ruhedauer in Parameter <code>sleepTime</code>) • <code>false</code>: Ruhemodus deaktivieren (Standard)
<i>sleepTime</i>	<p>Wartezeit in Sekunden zwischen der Übertragung der Dateien (wenn Parameter <code>multipleFiles</code> und <code>sleepEnabled</code> aktiviert sind)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl (Standard) • <code>0</code>: Keine Ruhezeit verwenden
<i>timeOut</i>	<p>Zeitbeschränkung für den TCP-Verbindungsaufbau und die Übertragung in Millisekunden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl (z.B. <code>50000</code>) • <code>0</code>: Keine Zeitbeschränkung (Standard)

<i>hostValidationEnabled</i>	<p>Host-Validierung aktivieren, um sicherzustellen, dass der angefragte FTP(S)/SFTP-Server auch dem FTP(S)/SFTP-Server entspricht, mit dem verbunden wird. Idealerweise nur für Testzwecke deaktivieren!</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>true</code>: Host-Validierung aktivieren (Flag setzen)  Ist die Host-Validierung aktiviert, muss der Parameter <code>hostsFileName</code> gesetzt sein. • <code>false</code>: Host-Validierung deaktivieren (nicht empfohlen für Produktivsysteme)(Standard)
<i>implicitFTPS</i>	<p>Implicit FTPS aktivieren</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>true</code>: Implicit FTPS aktivieren (Flag setzen) • <code>false</code>: Implicit FTPS deaktivieren (Standard)
<i>securityMechanism</i>	<p>Sicherheitsmechanismus</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>Explicit_TLS</code> • <code>Explicit_SSL</code> • <code>Explicit_TLS_C</code> • <code>Implicit_PRIVATE</code> • <code>Implicit_CLEAR</code>
<i>privateKeyFile</i>	<p>Pfad zu einer <i>Private Key</i>-Datei (nicht: <i>Java Keystore</i>). Wenn Sie sich auf dem FTP(S)/SFTP-Server mit einem <i>Private Key</i> authentifizieren möchten, dann dient der Parameter <code>password</code> als Passwort für Ihre <i>Private Key</i>-Datei.</p> <p>Mögliche Werte: Beliebiger gültiger Pfad (z.B. <code>C:\Data\X4Keys\id_rsa</code>)</p>
<i>hostsFileName</i>	<p>Pfad zur <i>Known Hosts</i>-Datei, die eine Liste von bestätigten Hosts und deren zugehörigen Public Keys enthält</p> <p>Mögliche Werte: Beliebiger gültiger Pfad (z.B. <code>C:\Data\known_hosts</code>)</p> <p> Die <i>Known Hosts</i>-Datei muss wie folgt aufgebaut sein:</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <pre>host,IP Verschlüsselung Host_PublicKey ftp.softproject.de,217.160.223.49 ssh-rsa AAAAB3NzaC1yc2EAAA...</pre> </div>

<i>useProxy</i>	Proxy-Server für den Verbindungsaufbau verwenden Mögliche Werte: <ul style="list-style-type: none"> • <code>true</code>: Proxy-Server verwenden • <code>false</code>: Keinen Proxy-Server verwenden (Standard)
<i>proxyType</i>	Proxy-Server-Typ Mögliche Werte: <ul style="list-style-type: none"> • <code>HTTP</code>: HTTP-Proxy verwenden (Standard) • <code>SOCKS4</code>: Proxy auf Basis des <i>SOCKS 4</i>-Protokolls verwenden • <code>SOCKS5</code>: Proxy auf Basis des <i>SOCKS 5</i>-Protokolls verwenden
<i>proxyAddress</i>	IP-Adresse oder Host-Name des Proxy-Servers Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>server.tld</i>)
<i>proxyPublicAddress</i>	Öffentliche IP-Adresse oder Host-Name des Proxy-Servers, wenn der Proxy keine Adresse zurückgibt (erforderlich z.B. bei aktiven FTP-Übertragungen, wenn der FTP(S)/SFTP-Server sich erneut mit dem Proxy verbinden muss) Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>server.tld</i>)
<i>proxyPort</i>	Port-Nummer des Proxy-Servers Mögliche Werte: Beliebige positive ganze Zahl
<i>proxyUsername</i>	Benutzername für den Proxy-Server
<i>proxyPassword</i>	Passwort für den Proxy-Server

Input

Im Datei-Modus (wenn Parameter `directoryMode` deaktiviert ist), wird für die Operation `Send` ein Input-Dokument erwartet, das versendet wird.

Die Unterschiedlichen Login-Operationen haben eine Reihenfolge. Wenn die jeweiligen Parameter gefüllt sind gilt:

1. Login-Commands werden verwendet
2. User/Passwort mit `AccountInfo` wird verwendet
3. User/Passwort wird verwendet

Definition der LoginCommands-XML:

```
<loginCommands>
  <loginCommand>cmd1</loginCommand>
  <loginCommand>cmd2</loginCommand>
</loginCommands>
```

Ausgabe

Im Datei-Modus (wenn Parameter `directoryMode` deaktiviert ist), wird für die Operation `Receive` das empfangene Dokument ausgegeben.

Für Operation `List` wird ein XML-Dokument mit der Liste der Ordner und Dateien ausgegeben.

6.4.4.2 FTP Transfer (veraltet)

FTP Transfer: Ermöglicht die Nutzung des *File Transfer Protocols* (FTP) zum Senden, Empfangen, Auflisten, Umbenennen oder Löschen von Dateien. Zudem können mehrere Dateien zwischen einem lokalen und einem externen Ordner verschoben werden.



Der *FTP Transfer* Adapter wird mit Version 6.2.0 der X4 Suite abgekündigt und in zukünftigen Versionen der X4 Suite nicht mehr unterstützt. Bitte verwenden Sie stattdessen den [FTP Connector](#).

Allgemeine Eigenschaften und Parameter**Eigenschaften (Properties)**

<i>I/O Mode (input_type)</i>	Art der Quelle der zu verarbeitenden Daten Mögliche Werte: <ul style="list-style-type: none"> • <i>Actions</i>: Einzelne Datei verarbeiten, deren Name in Eigenschaft <i>Filename</i> angegeben wird • <i>Directory</i>: Alle Dateien eines angegebenen Ordners verarbeiten (oder Dateien gemäß Filterregel in Parameter <i>filter</i>)
------------------------------	---

<i>File name (documentname)</i>	<p>Name der zu verarbeitenden Datei (wenn I/O Mode auf <i>Actions</i> gesetzt ist). Beachten Sie: Wenn Sie diese Eigenschaft dynamisch parametrieren möchten, müssen Sie im zugehörigen XSL-Mapping deren technischen Namen <i>documentname</i> verwenden (z.B. <code><xsl:processing-instruction name="DynamicParameter">documentname="<xsl:value-of select="."/"/>"</xsl:processing-instruction></code>).</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <i>Data.xml</i>)</p>
<i>Delete (delete)</i>	<p>Quelldatei(en) nach erfolgreich abgeschlossener Verarbeitung auf dem FTP-Server löschen</p>
<i>Operation (action)</i>	<p>Bestimmt, welche FTP-Operation mit den verarbeiteten Dateien durchgeführt wird. Je nach Wert in I/O Mode unterscheidet sich die Funktionsweise (siehe unten).</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Delete</i>: In Eigenschaft <i>Filename</i> festgelegte Datei in einem Ordner, der in Parameter <i>dir</i> festgelegt wird, oder sämtliche Dateien gemäß Filterregel in Parameter <i>filter</i> löschen • <i>List</i>: Liste der Dateien des in Parameter <i>dir</i> angegebenen Ordners als XML-Dokument ausgeben • <i>Make directory</i>: Ordner erzeugen, dessen Name in Eigenschaft <i>Filename</i> (<i>documentname</i>) festgelegt wird • <i>Delete directory</i>: Ordner löschen, dessen Name in Eigenschaft <i>Filename</i> (<i>documentname</i>) festgelegt wird • <i>Rename file</i>: Datei umbenennen, deren bisheriger Name in Parameter <i>filter</i> festgelegt wurde; als neuer Name wird der Wert von Eigenschaft <i>Filename</i> verwendet (ab X4-Version 4.0).

Parameter

<i>host</i>	<p>IP-Adresse oder Host-Name des FTP-Servers</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>ftp.server.tld</i>)</p>
<i>port</i>	<p>Port-Nummer des FTP-Servers</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl • 21: FTP-Port (Standard)

<i>login</i>	Benutzername für die FTP-Verbindung
<i>password</i>	Passwort für die FTP-Verbindung
<i>dir</i>	Ordner auf dem FTP-Server, dessen Inhalt per FTP manipuliert werden soll Mögliche Werte: Beliebige Zeichenkette (z.B. <i>Folder/Subfolder</i>)
<i>timeout</i>	Zeitbeschränkung für den Verbindungsaufbau und Übertragung in Millisekunden Mögliche Werte: <ul style="list-style-type: none"> • Beliebige ganze Zahl außer 0 • 50000: Zeitbeschränkung von 50 Sekunden (Standard)
<i>transfer_mode</i>	FTP-Übertragungsmodus Mögliche Werte: <ul style="list-style-type: none"> • <i>Passive</i>: Passiver Übertragungsmodus (Standard) • <i>Active</i>: Aktiver Übertragungsmodus
<i>max</i>	Anzahl der zu verarbeitenden Dateien begrenzen (wenn I/O Mode auf <i>Directory</i> gesetzt ist und somit mehrere Dateien eines Ordners verarbeitet werden sollen) Mögliche Werte: <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 0: keine Begrenzung der Anzahl • 1: maximal eine Datei verarbeiten (Standard)
<i>retries</i>	Anzahl der Wiederholung von Verbindungsversuchen, falls die Verbindung nicht direkt aufgebaut werden kann Mögliche Werte: <ul style="list-style-type: none"> • Beliebige ganze Zahl außer 0 • 3: maximal 3 Versuche, eine Verbindung aufzubauen (Standard)
<i>filter</i>	Nach Dateinamen filtern über die Platzhalter * und ? Mögliche Werte: <ul style="list-style-type: none"> • Beliebige Zeichenkette, so gilt beispielsweise *.dat nur für Dateien mit der Dateiendung .dat • *: Alle Dateien verwenden (Standard)

Eigenschaften

<i>Operation (action)</i>	<p>Bestimmt, welche FTP-Operation mit der in <code>Filename</code> (documentname) festgelegten Datei durchgeführt wird.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Send</i>: Daten aus dem vorangegangenen Prozess-Schritt in den in Parameter <code>Directory</code> angegebenen Ordner speichern, dabei den in Eigenschaft <code>Filename</code> festgelegten Namen verwenden • <i>Append</i>: Wie <i>Send</i>, allerdings werden die gesendeten Daten der Zielfeile angehängt, wenn bereits eine gleichnamige Zielfeile besteht. • <i>Receive</i>: Datei in den Prozess einlesen, deren Pfad in Parameter <code>dir</code> und deren Dateiname in Eigenschaft <code>Filename</code> festgelegt wurde.
---------------------------	--

Parameter

<i>toXML</i>	<p>Inhalt der verarbeiteten Datei Base64-kodiert als XML-Dokument inklusive Header ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: XML-Dokument ausgeben • <i>nein</i>: Kein XML-Dokument ausgeben (Standard)
--------------	--

I/O Mode: Directory**Eigenschaften**

<i>Operation (action)</i>	<p>Bestimmt, welche FTP-Operation mit dem in Parameter <code>Directory</code> festgelegten Ordner durchgeführt wird.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Send</i>: Alle Dateien von Parameter <code>localDir</code> nach <code>dir</code> kopieren, die <code>filter</code> entsprechen • <i>Append</i>: Wie <i>Send</i>, allerdings werden die Inhalte der gesendeten Dateien den Zielfeilen angehängt, wenn bereits gleichnamige Zielfeilen besteht. • <i>Receive</i>: Alle Dateien von Parameter <code>dir</code> nach <code>localDir</code> kopieren, die <code>filter</code> entsprechen
---------------------------	--

Parameter


<i>localDir</i>	<p>Lokaler Ordner auf dem X4 Server, etwa im X4 Repository, wohin heruntergeladene Dateien gespeichert werden oder von welchem Dateien auf einen FTP-Server hochgeladen werden sollen</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <code>%USER_COLLECTION%\Folder\</code>)</p>
-----------------	--

<i>toXML</i>	<p>XML-Dokument ausgeben, das eine Liste der verarbeiteten Dateien mit Änderungsdatum und Dateigröße enthält</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>ja</i>: XML-Dokument ausgeben• <i>nein</i>: Kein XML-Dokument ausgeben (Standard)
--------------	--

6.4.4.3 SFTP Communication (veraltet)



SFTP Communication: Ermöglicht die Nutzung des *SSH File Transfer Protocol* zum Senden, Empfangen, Auflisten oder Löschen von einer oder mehreren Dateien. Zudem können mehrere Dateien zwischen zwei Ordnern auf dem SFTP-Server verschoben und neue Ordner angelegt oder gelöscht werden.

 Der *SFTP Communication* Adapter wird mit Version 6.2.0 der X4 Suite abgekündigt und in zukünftigen Versionen der X4 Suite nicht mehr unterstützt. Bitte verwenden Sie stattdessen den [FTP Connector](#).

Eigenschaften (Properties)

<i>Operation (action)</i>	<p>Bestimmt, welche SFTP-Operation mit den verarbeiteten Dateien oder Ordnern durchgeführt wird.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>List</i>: Liste der Dateien des in Parameter <code>remotePath</code> angegebenen Ordners als XML-Dokument ausgeben • <i>Send</i>: Daten aus dem vorangegangenen Prozess-Schritt in den in Parameter <code>remotePath</code> angegebenen Ordner speichern, dabei den in Parameter <code>documentName</code> festgelegten Namen verwenden. Oder mehrere Dateien von Quellordner in Parameter <code>localPath</code> an <code>remotePath</code> versenden, indem die Parameter <code>directoryMode</code> und <code>multipleFiles</code> auf <i>true</i> gesetzt werden; in Parameter <code>filter</code> kann zudem ein Filterausdruck angegeben werden. • <i>Receive</i>: Datei in den Prozess einlesen, deren Pfad in Parameter <code>remotePath</code> und deren Dateiname in Parameter <code>documentName</code> festgelegt wurde. Oder mehrere Dateien empfangen, indem die Parameter <code>directoryMode</code> und <code>multipleFiles</code> auf <i>true</i> und <code>localPath</code> auf den lokalen Zielordner gesetzt werden; in Parameter <code>filter</code> kann zudem ein Filterausdruck angegeben werden. • <i>Delete</i>: In Parameter <code>documentName</code> festgelegte Datei oder Ordner auf dem SFTP-Server löschen, deren Pfad in Parameter <code>remotePath</code> festgelegt wird. Um nicht-leere Ordner zu löschen, zusätzlich Parameter <code>recurse</code> aktivieren. • <i>Move</i>: In Parameter <code>documentName</code> festgelegte Datei oder Ordner innerhalb des SFTP-Servers von <code>oldPath</code> nach <code>remotePath</code> verschieben • <i>Make directory</i>: Ordner auf dem SFTP-Server erzeugen, dessen Name in Parameter <code>documentName</code> und dessen Pfad in Parameter <code>remotePath</code> festgelegt wird • <i>Delete directory</i>: Ordner auf dem SFTP-Server löschen, dessen Name in Parameter <code>documentName</code> und dessen Pfad in Parameter <code>remotePath</code> festgelegt wird
---------------------------	---

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.sftpadapter.SFTPAdapter</code> : Hauptklasse (Standard)</p>
<i>host</i>	<p>IP-Adresse oder Host-Name des SFTP-Servers</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <code>sftp.server.tld</code>)</p>

<i>port</i>	<p>Port-Nummer des SFTP-Servers</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 22: SFTP-Port (Standard)
<i>login</i>	Benutzername für die SFTP-Verbindung
<i>password</i>	Passwort für die SFTP-Verbindung; Wenn Parameter <i>privateKeyFile</i> gesetzt ist: Passwort des <i>Private Keys</i>
<i>remotePath</i>	<p>Pfad auf dem SFTP-Server, dessen Ziel (Ordner oder Datei) per SFTP manipuliert werden soll</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <i>Folder/Subfolder</i>)</p>
<i>documentName</i>	<p>Name des Ordners oder der Datei, der/die verarbeitet wird oder einen Filterausdruck, der die Platzhalter * und ? enthalten kann.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige Zeichenkette (z.B. <i>Data.xml</i>) • Beliebigen Filterausdruck (z.B. <i>*.txt</i>)
<i>oldPath</i>	<p>Pfad zum Ordner oder zur Datei auf dem SFTP-Server, der/die verschoben werden soll (nur bei Operation <i>Move</i>)</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <i>Folder/Subfolder</i>)</p>
<i>oldDocumentName</i>	<p>Name des Ordners oder der Datei, der/die verschoben werden soll (nur bei Operation <i>Move</i>)</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <i>Data.xml</i>)</p>
<i>recurse</i>	<p>Pfad rekursiv (d. h. inklusive aller Unterordner und Dateien) löschen (nur bei Operation <i>Delete</i>)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Ja</i>: Alle beinhalteten Ordner (und Unterordner) und Dateien löschen • <i>Nein</i>: Nur Ordner und Dateien im aktuellen Pfad löschen (Standard)

<i>directoryMode</i>	<p>Ordner-Modus aktivieren, um den Inhalt von Ordnern zu verarbeiten; falls Parameter <i>multipleFiles</i> aktiviert ist, werden alle Elemente verarbeitet, die dem Filterkriterium in Parameter <i>filter</i> entsprechen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Ja</i>: Ordner verarbeiten • <i>Nein</i>: Nur einzelne Dateien verarbeiten (Standard)
<i>localPath</i>	<p>Lokaler Ordner zum Senden und Empfangen von mehreren Dateien (nur wenn die Parameter <i>directoryMode</i> und <i>multipleFiles</i> aktiviert sind)</p> <p>Mögliche Werte: Beliebige gültige Pfadangabe (z.B. <i>C:\Directory</i>)</p>
<i>multipleFiles</i>	<p>Alle Dateien verarbeiten, die dem Filterkriterium in Parameter <i>filter</i> entsprechen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Ja</i>: Alle zutreffenden Dateien verarbeiten • <i>Nein</i>: Nur die Datei verarbeiten, die in Parameter <i>documentName</i> angegeben wurde, oder wenn Parameter <i>directoryMode</i> aktiviert ist, die erste Datei (falls das Filterkriterium auf mehrere Dateien zutrifft) (Standard)
<i>filter</i>	<p>Filterausdruck, der die Platzhalter * und ? enthalten kann, um die Menge der verarbeiteten Dateien einzuschränken (nur wenn die Parameter <i>multipleFiles</i> und <i>directoryMode</i> aktiviert sind)</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <i>*.xml</i>)</p>
<i>deleteAfterReceive</i>	<p>Datei oder Ordner nach dem Empfangen vom SFTP-Server löschen (nur bei Operation <i>Receive</i>)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Ja</i>: Ursprüngliche Datei/Ordner nach dem Empfangen auf dem SFTP-Server löschen • <i>Nein</i>: Ursprüngliche Datei/Ordner behalten (Standard)

<i>append</i>	<p>Gesendete Daten an die Zieldatei anhängen, wenn bereits eine gleichnamige Zieldatei besteht (nur bei Operation <i>Send</i>)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>Ja</i>: Daten in der Zieldatei (falls vorhanden) anhängen• <i>Nein</i>: Zieldatei (falls vorhanden) überschreiben (Standard)
<i>maxTransferRate</i>	<p>Maximale Übertragungsrate in Bytes/Sekunde</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• Beliebige positive ganze Zahl• <i>0</i>: keine Begrenzung (Standard)
<i>sleepEnabled</i>	<p>Ruhemodus (<i>Sleep Mode</i>) aktivieren (wenn Parameter <i>multipleFiles</i> aktiviert ist)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>Ja</i>: Ruhemodus aktivieren (ermöglicht die Definition einer Ruhedauer in Parameter <i>sleepTime</i>)• <i>Nein</i>: Ruhemodus deaktivieren (Standard)
<i>sleepTime</i>	<p>Wartezeit in Sekunden zwischen der Übertragung der Dateien (wenn Parameter <i>multipleFiles</i> und <i>sleepEnabled</i> aktiviert sind)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• Beliebige positive ganze Zahl (Standard)• <i>0</i>: Keine Ruhezeit verwenden
<i>timeOut</i>	<p>Zeitbeschränkung für den TCP-Verbindungsaufbau und die Übertragung in Millisekunden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• Beliebige positive ganze Zahl (z.B. <i>50000</i>)• <i>0</i>: Keine Zeitbeschränkung (Standard)

<i>hostValidationEnabled</i>	<p>Host-Validierung aktivieren, um sicherzustellen, dass der angefragte SFTP-Server auch dem SFTP-Server entspricht, mit dem verbunden wird. Idealerweise nur für Testzwecke deaktivieren!</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Ja</i>: Host-Validierung aktivieren (Flag setzen) • <i>Nein</i>: Host-Validierung deaktivieren (nicht empfohlen für Produktivsysteme) (Standard)
<i>privateKeyFile</i>	<p>Pfad zu einer <i>Private Key</i>-Datei (nicht: <i>Java Keystore</i>). Wenn Sie sich auf dem SFTP-Server mit einem <i>Private Key</i> authentifizieren möchten, dann dient der Parameter <i>password</i> als Passwort für Ihren <i>Private Key</i>.</p> <p>Mögliche Werte: Beliebiger gültiger Pfad (z.B. <i>C:\Data\X4Keys\id_rsa</i>)</p>
<i>hostsFileName</i>	<p>Pfad zur <i>Known Hosts</i>-Datei, die eine Liste von bestätigten Hosts und deren zugehörigen Public Keys enthält</p> <p>Mögliche Werte: Beliebiger gültiger Pfad (z.B. <i>C:\Data\known_hosts</i>)</p>

6.4.5 iCalendar



iCalendar: Erzeugt aus einem adapterspezifischen XML-Dokument ein *iCalendar*-Ereignis gemäß RFC 5545 oder konvertiert ein *iCalendar*-Textdokument in ein XML-Dokument. Zudem kann ein *POP/SMTP*-Adapter dynamisch parametrisiert werden, sodass im Anhang einer E-Mail ein *iCalendar*-Ereignis (z. B. eine Termineinladung) versendet wird.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>iCal To XML</i>: <i>iCal</i>-Textdokument in XML konvertieren • <i>XML To iCal</i>: Adapterspezifische XML-Struktur in ein <i>iCalendar</i>-Textdokument konvertieren • <i>Create iXmap for E-Mail Adapter</i>: <i>iXmap</i>-XML-Struktur mit Parameterwerten und dem <i>iCal</i>-Ereignis als E-Mail-Body ausgeben, die zum dynamischen Parametrisieren eines <i>POP/SMTP</i>-Adapters dient; siehe <i>X4 Entwickler-Referenz – Dynamische Prozesse über iXmap</i>.
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.icalendar.ICalendarAdapter</i> : Hauptklasse (Standard)
<i>attachmentName</i>	Dateiname des E-Mail-Anhangs mit dem <i>iCal</i> -Kalenderelement (für Operation Create iXmap for E-Mail Adapter) Mögliche Werte: Beliebiger gültiger Dateiname

Beispiel

Im Demosystem *X4 Experience* finden Sie den Beispielprozess *iCalendarAdapter.wrf* zur Verwendung des Adapters.

Abhängigkeiten

Dieser Adapter folgende Bibliotheken:

- ical4j-1.0.jar
- backport-util-concurrent-3.1.jar
- icalendar.jar

6.4.5.1 Operation: iCal to XML

Mit Operation *iCal To XML* lässt sich ein gültiges *iCal*-Textdokument in XML konvertieren.

Input

Für Operation *iCal To XML* erwartet der Adapter ein beliebiges gültiges *iCal*-Textdokument.

Ausgabe

Für Operation *iCal To XML* gibt der Adapter ein XML-Dokument aus, das sämtliche Termin-Informationen enthält (siehe Beschreibung der Input-Struktur bei Operation *XML To iCal*).

6.4.5.2 Operation: XML to iCal

Mit Operation *XML To iCal* lässt sich eine adapterspezifische XML-Struktur in ein *iCalendar*-Textdokument konvertieren.

Input

Der Adapter erwartet ein Input-XML-Dokument gemäß folgendem Muster:

```

<iCalendarData>
  <!-- E-Mail-Metadaten: Nur relevant, wenn der Adapter ein iXmap
        für den POP/SMTP Adapter erzeugen soll -->
  <mail>
    <from><!--E-Mail-Adresse des Absenders--></from>
    <to><!--E-Mail-Adressen der Empfänger durch Semikolons (";") getrennt--></to>
    <cc><!--E-Mail-Adressen der CC-Empfänger durch Semikolons getrennt--></cc>
    <bcc><!--E-Mail-Adressen der BCC-Empfänger durch Semikolons getrennt--></bcc>
    <subject><!--Betreff--></subject>
  </mail>
  <prodId><!--Programm-ID, das die iCalendar-Daten erzeugt hat--></prodId>

  <!-- Veröffentlichungsart gemäß RFC 2446 (S.73)
        z. B. "REQUEST": Besprechungsanfrage;
        "PUBLISH": rein informative Terminveröffentlichung -->
  <method><!--Veröffentlichungsart--></method>
  <timezoneId><!--Beliebige Zeitzonen-ID von ical4j, z. B. Europe/Berlin--></
timezoneId>

  <!-- Ereignis: siehe RFC 2445 S. 51 -->
  <vevent>
    <uid><!--Global eindeutige ID für die iCalendar-Komponente (ggf. automatisch
erzeugt)--></uid>
    <organizer>
      <name><!--Name des Organisators--></name>
      <mail><!--E-Mail-Adresse des Organisators--></mail>

      <!-- zusätzliche Parameter je nach iCalendar-Eigenschaft
            (nur für Experten) -->
      <parameters>
        <parameter>
          <name><!--Parametername--></name>
          <value><!--Parameterwert--></value>
        </parameter>
      </parameters>
    </organizer>

    <!-- Teilnehmer: (von MS Outlook nur teilweise unterstützt) -->
    <!-- Outlook verwendet stattdessen die Empfängerdaten der E-Mail -->
    <attendees>
      <attendee>
        <name><!--Name des Teilnehmers--></name>
        <mail><!--E-Mail des Teilnehmers--></mail>

        <!-- Teilnehmerinformationen; siehe RFC 2445 S. 28 -->
        <role><!--REQ-PARTICIPANT, OPT-PARTICIPANT oder NON-PARTICIPANT--></role>
        <partStat><!--Teilnahmestatus (bei Antwort): ACCEPTED, DECLINED ...--></
partStat>
        <askForResponse><!--Antwort anfordern: true oder false--></
askForResponse>

        <!-- zusätzliche Parameter je nach iCalendar-Eigenschaft
              (nur für Experten) -->
        <parameters>
          <name><!--Parametername--></name>
          <value><!--Parameterwert--></value>

```

```

        </parameters>
    </attendee>
</attendees>

    <startTime timezoneId="Europe/London"><!--Startzeitpunkt (DATE oder DATE-
TIME)-->
    </startTime>
    <endTime><!--Endzeitpunkt des Termins (DATE oder DATE-TIME)--></endTime>

    <!-- Wenn ein Termin geändert wird: 0-n; wird bei
        jeder Änderung um 1 erhöht -->
    <sequence><!--Sequenz: 0-n--></sequence>
    <priority><!--Priorität von 1-9 (1=höchste Priorität, 0=keine Priorität)--></
priority>
    <summary><!--Titel des Termins--></summary>
    <description><!--Beschreibungstext des Termins--></description>
    <location><!--Ort--></location>
    <status><!--Status des Termins: CONFIRMED, TENTATIVE ...--></status>

    <!-- Hinweis: MS Outlook kann Termine mit gesetzten
        Elementen "rrule" und "exrule" nicht verarbeiten! -->
    <rrule><!--Wiederholungsregel--></rrule>
    <exrule><!--Ausnahmen der Wiederholungsregel--></exrule>

    <!-- siehe RFC 2445 S.66 -->
    <!-- Löst zu einem bestimmten Zeitpunkt eine Aktion aus;
        kann nur in "vevent" und "vtodo2" verwendet werden -->
    <valarm>

        <!-- siehe RFC 2445 S.70 -->
        <trigger>
            <value><!--Zeitangabe, deren Typ von Element "type" abhängt--></value>
            <type><!--Zeitangabentyp: DURATION (Standard), DATE-TIME oder DATE--></
type>

            <!-- wenn dieses Element "true" enthält und als Typ "DURATION"
                gewählt ist, dann wird das Element "value" relativ zum
                Endzeitpunkt der übergeordneten Komponente betrachtet,
                andernfalls relativ zum Startzeitpunkt -->
            <relativeToEnd>Relative Angabe: true oder false--></relativeToEnd>
        </trigger>

        <!-- siehe RFC 2445 S.69 -->
        <action><!--DISPLAY--></action>
        <repeat><!--Anzahl an Wiederholungen des Alarms--></repeat>

        <!-- Zeitabstand zwischen den Alarmen, wenn "repeat" größer 0
            ist (hier 30 Sekunden) -->
        <duration><!--PT30S--></duration>
        <summary><!--Titel der Alarmmeldung--></summary>
        <description><!--Beschreibung der Alarmmeldung--></description>

        <!-- Optional: Definition von "attendees" und "otherProperties"
            (Struktur wie in "vevent") -->
    </valarm>

    <categories>

```

```

        <!-- 1-n Kategorien, zu denen der Termin gehören soll -->
        <category><!--Kategorienname--></category>
    </categories>

    <!-- Sonstige Properties (nur Properties, die für
        diese Komponente definiert sind) -->
    <otherProperties>
        <property>
            <name><!--Eigenschaftsname--></name>
            <value><!--Eigenschaftswert--></value>
            <parameters>
                <parameter>
                    <name><!--Parametername--></name>
                    <value><!--Parameterwert--></value>
                </parameter>
            </parameters>
        </property>
    </otherProperties>
</vevent>

<!-- Aufgabe: siehe RFC 2445 S.54 -->
<vtodo>
    <due timezoneId="US-Eastern"><!--Endzeitpunkt im Format YYYYMMDDThhmmss--></
due>
    <duration><!--Dauer - nicht gemeinsam mit Element "due" verwenden--></duration>

    <!-- Hier können zusätzlich alle Elemente aus "vevent" außer "endTime"
        verwendet werden. -->
</vtodo>

<!-- Journal: siehe RFC 2445 S.55 -->
<vjournal>
    <!-- Alle Elemente aus "vevent" außer: "sequence", "priority",
        "location", "endTime" und "alarm" -->
</vjournal>

<!-- Belegt/Frei: siehe RFC 2445 S.57 -->
<vfree_busy>
    <uid>(siehe Element vevent)</uid>
    <organizer><!--siehe Element "vevent"--></organizer>
    <attendees><!--siehe Element "vevent"--></attendees>
    <startTime><!--siehe Element "vevent"--></startTime>
    <endTime><!--siehe Element "vevent"--></endTime>
    <duration><!--Dauer - nicht gemeinsam mit Element "endTime" verwenden--></
duration>
    <comment><!--Kommentartext--></comment>
    <otherProperties><!--siehe Element "vevent"--></otherProperties>
</vfree_busy>

    <otherProperties><!--siehe Element "vevent"--></otherProperties>
</iCalendarData>

```

Ausgabe

Für Operation XML To iCal gibt der Adapter ein entsprechendes *iCalendar*-Textdokument aus.

6.4.5.3 Operation: Create iXmap for E-Mail Adapter

Mit Operation Create iXmap for E-Mail Adapter lässt sich eine *iXmap*-XML-Struktur mit Parameterwerten und dem *iCal*-Ereignis als E-Mail-Body ausgeben, die zum dynamischen Parametrieren eines *POP/SMTP*-Adapters dient; siehe *X4 Developer Guide – Dynamische Prozesse über iXmap*.

Input

Für Operation Create iXmap for E-Mail Adapter erwartet der Adapter die gleiche XML-Struktur wie bei Operation XML To iCal.

Zusätzlich kann jedoch das Wurzel-Element als erstes Kindelement ein Element `<mail>` enthalten, in dem für jeden Parameter, der beim *POP/SMTP*-Adapter dynamisch gesetzt werden soll, ein gleichnamiges Element inklusive Wert definiert werden kann:

```
<iCalendarData>
  <mail>
    <!-- für jeden zu setzenden Parameter -->
    <Parametername 1><!--Wert 1--></Parametername 1>
    <Parametername n><!--Wert n--></Parametername n>
  </mail>
  <!-- Termindaten -->
</iCalendarData>
```

Ausgabe

Für Operation Create iXmap for E-Mail Adapter gibt der Adapter ein *iXmap*-XML-Dokument aus, das zum dynamischen Parametrieren eines *POP/SMTP*-Adapters dient; siehe *X4 Developer Guide – Dynamische Prozesse über iXmap*.

Damit die Prozess-Engine die *iXmap*-Struktur jedoch als solche erkennt, muss dieses XML-Dokument erneut via XSL transformiert werden (Identitätstransformation genügt). Ein darauf folgender *POP/SMTP*-Adapter wird dann dynamisch mit den im Input-XML-Dokument angegebenen Werten parametrier.

```
<iXmap>
  <Return>
    <!-- für jeden gesetzten Parameter -->
    <Parametername 1><!--Wert 1--></Parametername 1>
    <Parametername n><!--Wert n--></Parametername n>
  </Return>
</iXmap>
```

6.4.6 Ping



Ping: Pingt eine IP-Adresse über das *Internet Control Message Protocol* an. Dabei sendet er *ICMP ECHO REQUESTs* oder versucht alternativ, eine TCP-Verbindung auf Port 7 aufzubauen. Der

Adapter gibt ein XML-Dokument aus, welches das Ergebnis des Echos (*true* oder *false*) enthält.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Ping</i> : Ping ausführen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.icmpping.ICMPPing</i> : Hauptklasse (Standard)
<i>IPAddress</i>	IP-Adresse, an die der Ping gesendet wird Mögliche Werte: <ul style="list-style-type: none"> • Beliebige IP-Adresse (z.B. <i>127.0.0.1</i>) • <i>0.0.0.0</i>: Leere IP-Adresse (Standard)
<i>timeout</i>	Zeitbeschränkung für den Verbindungsaufbau in Millisekunden Mögliche Werte: <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • <i>0</i>: Keine Zeitbeschränkung (Standard)

6.4.7 SCP



SCP: Sendet und empfängt einzelne Dateien über SCP (*Secure CoPy*). Der Adapter unterstützt Proxy-Server vom Typ HTTP, SOCKS4 und SOCKS5.

Ehemalige Bezeichnung bis Version 4.5.1: *SCP Transfer Adapter*

Eigenschaften (Properties)

<i>Operation (action)</i>	<p>Bestimmt, welche Dateitransfer-Operation mit den verarbeiteten Dateien oder Ordnern durchgeführt wird.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Send</i>: Eine in Parameter <code>documentname</code> angegebene Datei, die sich in einem in Parameter <code>localPath</code> festgelegten lokalen Ordner befindet, an einen entfernten Ordner versenden, der in Parameter <code>remotePath</code> angegeben wurde. • <i>Receive</i>: Datei empfangen, deren Pfad in Parameter <code>remotePath</code> und deren Dateiname in Parameter <code>documentName</code> festgelegt wurde und in einem lokalen Ordner speichern, der in Parameter <code>localPath</code> angegeben wurde.
---------------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.scpadapter.SCPAdapter</code> : Hauptklasse (Standard)</p>
<i>host</i>	<p>IP-Adresse oder Host-Name des entfernten Rechners</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <code>server.tld</code>)</p>
<i>port</i>	<p>Port-Nummer des entfernten Rechners</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 22: SCP-Standardport (Standard)
<i>login</i>	Benutzername für die SCP-Verbindung
<i>password</i>	Passwort für die SCP-Verbindung; Wenn Parameter <code>privateKeyFile</code> gesetzt ist: Passwort des <i>Private Keys</i>
<i>remotePath</i>	<p>Pfad auf dem entfernten Rechners, mit dem eine Datei ausgetauscht werden soll</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <code>Folder/Subfolder</code>)</p>
<i>documentname</i>	<p>Name der Datei, die gesendet oder empfangen werden soll</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <code>Data.xml</code>)</p>

<i>localPath</i>	<p>Lokaler Ordner zum Senden und Empfangen einer Datei</p> <p>Mögliche Werte: Beliebige gültige Pfadangabe (z.B. <i>C:\Directory</i>)</p>
<i>timeOut</i>	<p>Zeitbeschränkung für den TCP-Verbindungsaufbau und die Übertragung in Millisekunden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl (z.B. <i>50000</i>) • <i>0</i>: Keine Zeitbeschränkung (Standard)
<i>hostValidationEnabled</i>	<p>Host-Validierung aktivieren, um sicherzustellen, dass der angefragte entfernte Rechner auch dem Rechner entspricht, mit dem verbunden wird. Idealerweise nur für Testzwecke deaktivieren!</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Ja</i>: Host-Validierung aktivieren (Flag setzen) • <i>Nein</i>: Host-Validierung deaktivieren (nicht empfohlen für Produktivsysteme) (Standard)
<i>privateKeyFile</i>	<p>Pfad zu einer <i>Private Key</i>-Datei (nicht: <i>Java Keystore</i>). Wenn Sie sich auf dem entfernten Rechner mit einem <i>Private Key</i> authentifizieren möchten, dann dient der Parameter <i>password</i> als Passwort für Ihre <i>Private Key</i>-Datei.</p> <p>Mögliche Werte: Beliebiger gültiger Pfad (z.B. <i>C:\Data\X4Keys\id_rsa</i>)</p>
<i>hostsFileName</i>	<p>Pfad zur <i>Known Hosts</i>-Datei, die eine Liste von bestätigten Hosts und deren zugehörigen Public Keys enthält</p> <p>Mögliche Werte: Beliebiger gültiger Pfad (z.B. <i>C:\Data\known_hosts</i>)</p>
<i>useProxy</i>	<p>Proxy-Server für den Verbindungsaufbau verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Ja</i>: Proxy-Server verwenden • <i>Nein</i>: Keinen Proxy-Server verwenden (Standard)

<i>proxyType</i>	Proxy-Server-Typ Mögliche Werte: <ul style="list-style-type: none"> • <i>HTTP</i>: HTTP-Proxy verwenden (Standard) • <i>SOCKS4</i>: Proxy auf Basis des <i>SOCKS 4</i>-Protokolls verwenden • <i>SOCKS5</i>: Proxy auf Basis des <i>SOCKS 5</i>-Protokolls verwenden
<i>proxyAddress</i>	IP-Adresse oder Host-Name des Proxy-Servers Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>server.tld</i>)
<i>proxyPublicAddress</i>	Öffentliche IP-Adresse oder Host-Name des Proxy-Servers, wenn der Proxy keine Adresse zurückgibt Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>server.tld</i>)
<i>proxyPort</i>	Port-Nummer des Proxy-Servers Mögliche Werte: Beliebige positive ganze Zahl
<i>proxyUsername</i>	Benutzername für den Proxy-Server
<i>proxyPassword</i>	Passwort für den Proxy-Server

Input

Der Adapter erwartet kein bestimmtes Input-Dokument. Ein Input-Dokument wird unverändert wieder ausgegeben.

Ausgabe

Ein Input-Dokument wird unverändert wieder ausgegeben.

6.4.8 SMS



SMS: Sendet Kurzmitteilungen via Short Message Service (SMS) über ein Mobiltelefon, das über eine serielle Schnittstelle mit der Hardware des X4 Servers verbunden ist.

i Hierzu muss die SMS-Funktion in der zentralen Konfigurationsdatei *<X4>/X4config.xml* mit *<sms>on</sms>* aktiviert sein, siehe *Handbuch X4 Server – iXServ-Konfiguration*.

Eigenschaften

<i>I/O Mode</i>	(keine Funktion)
-----------------	------------------

<i>File name</i>	(keine Funktion)
<i>Delete</i>	(keine Funktion)
<i>Operation</i>	Bestimmt, welche Operation durchgeführt wird. Mögliche Werte: <i>Send</i> : Kurznachricht senden

Parameter

<i>to</i>	Telefonnummer, an die eine Kurznachricht gesendet werden soll Mögliche Werte: Vollständige Telefonnummer mit Pluszeichen, Länderkürzel, Vorwahl (ohne führende Null) sowie die Rufnummer (z.B. <i>+49174123456</i>)
<i>text</i>	Standard-Text der Kurznachricht (wird nur verwendet, wenn das XML-Dokument im vorangegangenen Prozess-Schritt keinen Text im Wurzel-Element enthält) Mögliche Werte: Beliebige Zeichenfolge mit maximal 160 Zeichen
<i>port</i>	Serielle Schnittstelle, die mit einem geeigneten Mobiltelefon verbunden ist Mögliche Werte: <i>com1</i> : COM1-Anschluss am Rechner, auf dem der X4 Server läuft
<i>baud</i>	Datenübertragungsrate zur Kommunikation zwischen dem Mobiltelefon in Baud (hier: Bits pro Sekunde) Mögliche Werte: <ul style="list-style-type: none"> • Beliebige positive ganze Zahl in Abhängigkeit von der möglichen Übertragungsrate • <i>19200</i>: erfahrungsgemäß am besten funktionierende Baudrate (Standard)

6.4.9 TAPI Listener

TAPI Listener: Die TAPI-Prozess-Schnittstelle ermöglicht die Integration von Telefonanlagen in Geschäftsprozesse. Er reagiert dabei auf Telefonie-Ereignisse, etwa dem Verbindungsaufbau mit einem Gerät bzw. einem Anruf unter einer bestimmten Rufnummer. Über das *Telephony Application Programming Interface* (TAPI-Version 3.x) wird dann je nach Konfiguration des TAPI-Listeners ein entsprechender technischer Prozess gestartet.

Ehemalige Bezeichnung des Adapters bis Version 5.5.4: *TAPI Process Interface*

Einführung

X4 TAPI (*Telephony Application Programming Interface*) ermöglicht die nahtlose Integration von Telefonanlagen in Geschäftsprozesse. Dabei reagiert der *TAPI-Listener* der X4-Version 4.2 auf beliebige Telefonie-Events via TAPI-Schnittstelle und übermittelt die relevanten Informationen an einen weiterverarbeitenden technischen Prozess.

Eigenschaften von X4 TAPI

- Beliebig viele *TAPI Service Provider* bzw. Telefonanlagen können angebunden werden.
- Telefonie-Ereignisse werden für alle angeschlossenen Geräte oder für einzelne Geräte automatisch verarbeitet.
- Informationen zum jeweiligen Telefonie-Ereignis wie Anrufer und Angerufener werden als XML-Daten an den jeweiligen technischen Prozess übergeben.
- Telefonie-Geräte können einem oder mehreren *TAPI Service Providern* explizit zugewiesen werden.

Anwendungsbeispiel

Beispielsweise geht ein Anruf unter einer bestimmten Rufnummer ein. Die Telefonanlage routet den Anruf und stellt die entsprechenden Informationen bereit. Nun übernimmt *TAPI* und übergibt die Anruferdaten automatisch an einen technischen Prozess.

Dieser technische Prozess erstellt z.B. ein Support-Ticket oder bindet ein CRM-System an, um den technischen Prozess mit Zusatzinformationen anzureichern, z.B.:


- Kundennummer
- Zugehöriger Ansprechpartner
- Erstellte Angebote
- Bearbeitungs-Historie etc.

Nun wird eine entsprechende Oberfläche mit diesen Daten aufgerufen – noch bevor das Telefon des Sachbearbeiters klingelt.

Abschließend können Daten wie z. B. die Gesprächsdauer automatisch ermittelt und über technischen Prozesse verarbeitet werden.

Technische Beschreibung

Jedem Telefonie-Ereignis kann ein verarbeitender technischer Prozess zugeordnet werden. Dieser erhält als Input ein XML-Dokument mit der Telefonnummer des Anrufers und des Angerufenen.

 Die verfügbaren *TAPI Events* (Ereignisse) sind *TAPI Service Provider*- bzw. herstellerspezifisch. Für weitere Informationen zu den aktuell unterstützten *TAPI Events* wenden Sie sich bitte an support@softproject.de.

Beliebig viele *TAPI Service Provider* bzw. Telefonanlagen können angebunden werden. Die Ausführung des Listeners lässt sich auf einzelne Telefonie-Geräte beschränken, ebenso können Sie einzelnen Geräten ein oder mehrere *TAPI Service Provider* explizit zugewiesen.

TAPI basiert auf der *Java Telephony API (JTAPI) 1.4*, einer Java-Implementierung des *Telephony Application Programming Interface* von *Microsoft*. Daher wird in *x4.ear* in dessen Unterordner *lib* die Bibliothek *jtapi.jar* benötigt.

Weiterführende Informationen:

- *JTAPI 1.4* <http://www.oracle.com/technetwork/java/index-jsp-140696.html>
- *TAPI*: <http://msdn.microsoft.com/en-us/library/ms734273%28v=vs.85%29.aspx>

Konfiguration

Im Ordner <X4>/X4DB/0 wird in der Konfigurationsdatei `tapilistener.xml` verwaltet, bei welchen Ereignissen und für welche Geräte und *TAPI Service Provider* ein entsprechender technischer Prozess zur Weiterverarbeitung gestartet wird. Im selben Ordner befindet sich die zugehörige Schema-Definition `tapilistener.xsd`.

Das XML-Dokument `tapilistener.xml` besitzt folgende Struktur:

```
<tapilistener>
  <providers>
    <provider id="Provider-Name" name="IP-Adresse oder Hostname"
      user="Benutzername" password="Passwort"/>
  </providers>
  <events>
    <event name="TAPI-Ereignis"
      userId="Benutzer-ID"
      callable="Pfad zum technischen Prozess"
      description="Beschreibung der Zuordnung Ereignis/Prozess" />
  </events>
  <devices>
    <device name="Geräte-Nummer/Telefonnummer">
      <providerRef id="Provider-ID"/>
    </device>
  </devices>
</tapilistener>
```

Innerhalb des Elements `<providers>` wird in beliebig vielen Elementen `<provider>` jeweils ein *TAPI Service Provider* über folgende Attribute definiert:

- `id`: Eindeutiger Name des *TAPI Service Providers* (frei wählbar)
- `name`: IP-Adresse oder Hostname des *TAPI Service Providers*
- `user`: Benutzername für den *TAPI Service Provider*
- `password`: zugehöriges Passwort

Innerhalb des Elements `<events>` wird in jedem Element `<event>` ein Telefonie-Ereignis zu einem technischen Prozess über folgende Attribute zugeordnet:

- `name`: Telefonie-Ereignis, z. B. *ConnEv*
- `userId`: Benutzer-ID aus der zentralen X4-Konfigurationsdatei <X4>/X4DB/0/`tblAccess.xml`; Bestimmt das Benutzerverzeichnis (*User collection*), in dem der aufzurufende technische Prozess gespeichert ist
- `callable`: Pfad und Dateiname des zu startenden technischen Prozesses im Benutzer-Ordner des X4-Repositorys
- `description`: Beschreibungstext für die Zuordnung des Telefonie-Ereignisses zum technischen Prozess

Innerhalb des Elements `<devices>` kann in beliebig vielen Elementen `<device>` die TAPI-Listener-Konfiguration auf einzelne Telefonie-Geräte (d. h. Rufnummern) beschränkt werden.

- ❗ Wenn Element `<devices>` leer ist, dann gilt die TAPI-Listener-Konfiguration für alle angeschlossenen Telefonie-Geräte.

Für jedes Element `<device>` gilt:

- Attribut `name` enthält die Rufnummer des Telefonie-Geräts.
- Es darf beliebig viele Kind-Elemente `<providerRef id=" Provider-ID ">` enthalten, mit dem das Gerät jeweils auf einen bestimmten *TAPI Service Providers* über Attribut `id` festgelegt wird.
- Wenn Element `<device>` keine Kind-Elemente `<providerRef>` enthält, dann gilt die Geräte-Konfiguration für alle *TAPI Service Provider*, die in Element `<providers>` angegeben wurden.

Ausgabe

Der TAPI-Listener erzeugt für jedes verarbeitete Telefonie-Ereignis ein XML-Dokument, das als Input für den gestarteten technischen Prozess dient und die Telefonnummer des Anrufers und des angerufenen Geräts enthält:

```
<X4TAPI>
  <conn caller="Nummer des Anrufers" callee="Nummer des Angerufenen"/>
</X4TAPI>
```

Beispiel

Im folgenden Beispiel wird für die *TAPI Service Provider SP1* und *SP2* ein Anruf-Ereignis *ConnEv* definiert, das einen technischen Prozess `<X4>/X4DB/3/Project/Folder/ConnEvent.wrf` aufruft, wenn die (Durchwahl-) Nummer *115* angerufen wird.

```
<tapilistener xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="tapilistener.xsd">
  <providers>
    <provider id="SP1" name="192.168.2.0" user="test" password=""/>
    <provider id="SP2" name="192.168.2.1" user="test" password=""/>
  </providers>
  <events>
    <event name="ConnEv" userId="3" callable="Project/Folder/ConnEvent.wrf"
      description="Assigns a ConnEv event to the respective process"/>
  </events>
  <devices>
    <device name="115">
      <providerRef id="SP1"/>
      <providerRef id="SP2"/>
    </device>
  </devices>
</tapilistener>
```

Als Input erhält der technische Prozess folgendes XML-Dokument, wenn z.B. ein Anruf von 49724356175333 an die Rufnummer 115 eingeht:

```
<X4TAPI>
  <conn caller="49724356175333"
        callee="115"/>
</X4TAPI>
```

6.4.10 TCP/IP Connector/Listener

6.4.10.1 TCP/IP Communication



TCP/IP Communication: Führt synchrone oder asynchrone TCP-Socket-Aufrufe aus, öffnet dabei einen Socket und sendet die Input-Daten an den angegebenen Host.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: Approach: TCP-Socket aufrufen (Standard)
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.socketapproach.SocketApproach</i> : Hauptklasse (Standard)
<i>iPAddress</i>	IP-Adresse des Host-Rechners, an den Daten gesendet werden sollen Mögliche Werte: <ul style="list-style-type: none"> • Beliebige IP-Adresse (z.B. 127.0.0.1) • 0.0.0.0: Leere IP-Adresse (Standard)
<i>port</i>	Port des Host-Rechners Mögliche Werte: <ul style="list-style-type: none"> • Beliebige ganze positive Zahl (z.B. 8080) • 0: Port 0 (Standard)

<i>timeout</i>	Zeitbeschränkung für den Verbindungsaufbau in Millisekunden Mögliche Werte: <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • \emptyset: Keine Zeitbeschränkung (Standard)
----------------	--

6.4.10.2 TCP Process Interface

Mit der TCP Prozess-Schnittstelle können technische Prozesse über das *Transmission Control Protocol* (TCP) gestartet werden. Dabei kann für jeden Port genau ein technischer Prozess definiert werden. Jede an einen Port gesendete Byte-Folge wird vom entsprechenden *X4*-Prozess verarbeitet.

Der technische Prozess wird gestartet, wenn der Sender den Kanal zum *X4 Server* schließt (wobei die TCP-Verbindung aktiv bleibt). Nach der Ausführung des Prozesses wird der Ausführstatus des Prozesses als *signed 4 Byte Integer* in *Network Byte Order* gesendet, gefolgt von der Byte-Folge des Ausgabe-/Ergebnis-Dokument des technischen Prozesses. Nach Übermittlung des Ergebnisses wird der Antwortkanal geschlossen und die TCP-Verbindung beendet.

Konfiguration

Im Ordner `<X4>/X4DB/0` in der Datei `tcpstarter.xml` wird verwaltet, welche Prozesse auf welchem Port per TCP aufgerufen werden können. Dieses XML-Dokument besitzt folgende Struktur, wobei für jedes Element `<Port/>` ein zu startender technischer Prozess definiert werden kann:

```
<TcpStarter>
  <Port ip="IP-Adresse des Servers" port="Server-Port" x4User="Benutzer-ID"
    x4Process="Pfad zum Prozess">
</TcpStarter>
```

Parameter

Folgende Parameter müssen als Attribute für jeden zu startenden technischen Prozess definiert werden:

- `ip`: IP-Adresse des *X4 Servers*
- `port`: Für die TCP-Verbindung verwendeter Port
- `x4User`: Benutzer-ID aus der Konfigurationsdatei `<X4>/X4DB/0/tblAccess.xml`; Bestimmt das Benutzerverzeichnis (*User collection*), in dem der aufzurufende Prozess gespeichert ist
- `x4Process`: Pfad und Dateiname des zu startenden technischen Prozesses im Benutzer-Ordner des Repositorys
- `sendStatus`: Status des *X4*-Prozesses unmittelbar nach dessen Ausführung senden (*true*) oder keinen Ausführstatus senden (*false*); Standard ist *false*.

Beispiel

```
<TcpStarter>
  <Port ip="127.0.0.1" port="1234" x4User="1" x4Process="Project\Folder\Process.wrf"
    sendStatus="false">
  </TcpStarter>
```

6.4.11 UDP Listener

Mit der UDP Prozess-Schnittstelle können Systeme, die lediglich das Low-Level-Protokoll UDP (*User Datagram Protocol*) unterstützen, wie etwa Steuerungssysteme (SPS) in der Industrie, an den X4 ESB angebunden und Daten an verarbeitende X4-Prozesse übergeben werden.

Benennung des Adapters bis Version 5.5.4: *UDP Process Interface*

Dabei kann für jeden Port genau ein technischer Prozess definiert werden. Jede an einen Port gesendete Byte-Folge wird vom entsprechenden technischen Prozess verarbeitet.

Konfiguration

Im Ordner <X4>/X4DB/0 in der Datei `udpstarter.xml` wird verwaltet, welche technischen Prozesse auf welchem Port per UDP aufgerufen werden können. Dieses XML-Dokument besitzt folgende Struktur, wobei für jedes Element <Port/> ein zu startender technischer Prozess definiert werden kann:

```
<UdpStarter>
  <Port ip="IP-Adresse des Servers" port="Server-Port" x4User="Benutzer-ID"
    x4Process="Pfad zum technischen Prozess">
  </UdpStarter>
```

Parameter

Folgende Parameter müssen als Attribute für jeden zu startenden technischen Prozess definiert werden:

- `ip`: IP-Adresse des X4 Servers
- `port`: Für die UDP-Verbindung verwendeter Port
- `x4User`: Benutzer-ID aus der Konfigurationsdatei <X4>/X4DB/0/`tblAccess.xml`; Bestimmt das Benutzerverzeichnis (*User collection*), in dem der aufzurufende technische Prozess gespeichert ist
- `x4Process`: Pfad und Dateiname des zu startenden technischen Prozesses im Benutzer-Ordner des Repositorys

Ausgabe

Der UDP-Starter erzeugt aus den gesendeten Daten nach dem Abschicken ein XML-Dokument, das als Input für den gestarteten technischen Prozess dient. Dieses ist nach folgendem Muster aufgebaut:

```
<Udp>
  <Client>Host-Name</client>
  <Data>Base64-kodierte Daten</Data>
</Udp>
```

Beispiel

```
<UdpStarter>
  <Port ip="127.0.0.1" port="1234" x4User="1" x4Process="Project\Folder\Process.wrf"
  sendStatus="false">
</UdpStarter>
```

6.5 Compression

6.5.1 gzip Converter



gzip Converter: Komprimiert und dekomprimiert Input-Daten mit dem *gzip*-Algorithmus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>Encode</i>: Daten gzip-komprimieren • <i>Decode</i>: gzip-komprimierte Daten als MIME-Typ <i>Application/Octet- stream</i> dekomprimieren
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.GZipAdapter</i> : Hauptklasse (Standard)
----------------	--

Siehe auch:

- [ZIP in Memory Compression \(Veraltet\)](#)
- [Adapter](#)

6.5.2 ZIP

6.5.2.1 ZIP Compression (Veraltet)



ZIP Compression: Erzeugt ein ZIP-Archiv mit einer oder mehreren Dateien aus Daten, die in einer adapterspezifischen XML-Struktur abgelegt sind, und speichert das ZIP-Archiv in einem lokalen Zielordner.

⚠ Der *ZIP Compression* Adapter wird mit Version 6.2.0 der X4 Suite abgekündigt und in zukünftigen Versionen der X4 Suite nicht mehr unterstützt. Bitte verwenden Sie stattdessen den [Zip Converter](#).

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>zip</i> : ZIP-Archiv aus den Daten im Input-XML-Dokument erzeugen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.zipadapter.ZipAdapter</i> : Hauptklasse (Standard)
<i>directory</i>	Zielordner, in den das ZIP-Archiv gespeichert werden sollen <div style="border: 1px solid #add8e6; padding: 5px; margin: 5px 0;"> i Ab Version 5.0 der X4 Suite ist ein abschließender Schrägstrich nicht mehr erforderlich, um das Verzeichnis vom ZIP-Namen zu trennen. </div> Mögliche Werte: Beliebige gültige Zeichenkette (z. B. <i>%USER_COLLECTION%\Project\Folder\</i>)
<i>zipName</i>	Dateiname des erzeugten ZIP-Archivs Mögliche Werte: Beliebiger gültiger Dateiname (z. B. <i>Output.zip</i>)

Input

Dieser Funktions-Adapter erwartet eine adapterspezifische Input-XML-Struktur. Diese enthält für jede Datei, die im ZIP-Archiv enthalten sein soll, den Dateinamen und den Base64-kodierten Dateinhalt. Jedes Element `<document/>` steht hierbei für eine Datei im ZIP-Archiv.

```
<data>
  <documents>
    <document>
      <filename><!--Dateiname--></filename>
      <content><!--Base64-kodierte Daten--></content>
    </document>
  </documents>
</data>
```

6.5.2.2 Zip Converter



Zip Converter: Kann Dateien und Ordner im Speicher oder im Dateisystem zippen und entzippen.



Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> Zip: Komprimiert den bereitgestellten Input in einer Zip-Datei. Es können Verschlüsselungs- und Komprimierungseinstellungen vorgenommen werden. <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p>❗ Erforderliche Parameter</p> <p>Die Parameter <code>SourcePath</code> oder <code>InputName</code> werden für diesen Vorgang benötigt.</p> </div> <ul style="list-style-type: none"> Unzip: Dekomprimiert den bereitgestellten Input am angegebenen Speicherort. Es können auch passwortgeschützte Dateien dekomprimiert werden.
-----------	---

Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.x4.adapter.zipconverter.ZipConverter</code>: Hauptklasse (Standard)</p>
sourcePath	<p>Verzeichnispfad zur Datei, die komprimiert/dekomprimiert werden soll.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> Verzeichnispfad <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p>❗ Wenn dieser Parameter nicht gesetzt wird, dann wird die Datei über den Adapter-Input bereitgestellt. In diesem Fall ist der Parameter <code>inputName</code> für die Operation <code>Zip</code> erforderlich.</p> </div>

targetPath	<p>Verzeichnispfad zum Speicherort, an dem der Output gespeichert werden soll ("Zielverzeichnis").</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Verzeichnispfad <div> <p> Wenn dieser Parameter nicht gesetzt wird, dann wird die Datei über den Adapter-Output bereitgestellt.</p> </div>
overwrite	<p>Legt fest, ob vorhandene Dateien mit dem gleichen Namen überschrieben werden sollen.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Aktiviert <div> <p> <ul style="list-style-type: none"> • Wenn mergeContent aktiviert ist, dann wird im Output eine Fehlermeldung ausgegeben, da die Datei nicht weiterverarbeitet werden kann. • Wenn mergeContent deaktiviert ist, dann wird die vorhandene Datei überschrieben. </p> </div> • Deaktiviert <div> <p> <ul style="list-style-type: none"> • Wenn mergeContent aktiviert ist, dann wird werden die Dateien zusammengeführt. • Wenn mergeContent deaktiviert ist, dann wird im Output eine Fehlermeldung ausgegeben, da die Datei nicht weiterverarbeitet werden kann. </p> </div>
mergeContent	<p>Legt fest, ob vorhandene Dateien mit dem gleichen Namen zusammengeführt werden sollen.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Aktiviert <div> <p> <ul style="list-style-type: none"> • Wenn overwrite aktiviert ist, dann wird im Output eine Fehlermeldung ausgegeben, da die Datei nicht weiterverarbeitet werden kann. • Wenn overwrite deaktiviert ist, dann werden die Dateien zusammengeführt. </p> </div> • Deaktiviert <div> <p> <ul style="list-style-type: none"> • Wenn overwrite aktiviert ist, dann wird die vorhandene Datei überschrieben. • Wenn overwrite deaktiviert ist, dann wird im Output eine Fehlermeldung ausgegeben, da die Datei nicht weiterverarbeitet werden kann. </p> </div>

inputName	<p>Name der Input-Datei. Wird verwendet, wenn der sourcePath nicht gesetzt ist.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Dateiname der Datei, die als Input empfangen bzw. als Output ausgegeben werden soll
encryptedFiles	<p>Gibt an, ob in der Operation verschlüsselte Dateien verarbeitet werden sollen. Bei der Operation Zip wird die ausgegebene Zip-Datei passwortgeschützt, bei der Operation Unzip wird für das entzippen ein Passwort benötigt.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Aktiviert / Deaktiviert
password	<p>Passwort für verschlüsselte Dateien.</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
encryptionMethod	<p>Gibt an, welche Verschlüsselungsmethode beim Zippen verwendet wird, wenn encryptedFiles aktiviert ist.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • No encryption: Keine Verschlüsselung beim Zippen verwenden • AES: AES-Verschlüsselungsalgorithmus verwenden (Schlüssellänge: 256 Bit) • Standard: Standard-Verschlüsselungsalgorithmus verwenden
compressionMethod	<p>Gibt an, welcher Komprimierungsalgorithmus beim Zippen verwendet wird.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Deflate (Standard): Verlustfreie Komprimierung, die eine Kombination aus dem LZ77-Algorithmus und Huffman-Codierung verwendet. Weitere Informationen zu dieser Kompressionsstufe hier und hier. <div data-bbox="422 1406 1434 1536" style="border: 1px solid #add8e6; padding: 10px; margin: 10px 0;"> <p> Wenn Deflate ausgewählt ist, dann kann das Komprimierungslevel mit CompressionLevel spezifiziert werden.</p> </div> • Store: Daten werden nicht komprimiert, sondern in eine einzige Datei gepackt. <div data-bbox="422 1599 1434 1758" style="border: 1px solid #add8e6; padding: 10px; margin: 10px 0;"> <p> Das ist nützlich wenn mehrere Daten/Ordner gesendet/empfangen werden sollen und die Dateigröße vernachlässigt werden kann. Es wird keine Zeit für das Verpacken benötigt.</p> </div>

compressionLevel	<p>Gibt den Komprimierungsalgorithmus beim Zippen an, wenn für compressionMethod Deflate gewählt ist.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Fastest: Niedrigste Kompressionsstufe, aber sehr schnelle Kompression • Fast: Niedrige Kompressionsstufe, aber schnelle Kompression • Normal (Standard): Optimales Verhältnis zwischen Kompressionsstufe und -geschwindigkeit • Maximum: Hohe Kompressionsstufe, aber langsamere Kompression • Ultra: Höchste Kompressionsstufe, aber sehr langsame Kompression
------------------	---

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt.
-1	Bei der Ausführung des Adapters ist ein Fehler aufgetreten.

Eingabe

Der Adapter erwartet Input in folgender Struktur:

- Beide Operationen (Zip / Unzip) können mit zwei verschiedenen Herkunfts- bzw. Zielorten arbeiten:
 - Input/Output
 - Verzeichnispfad für Input/Output
- Basierend auf diesen zwei Möglichkeiten können für beide Operationen verschiedene Inputs verwendet werden:
 - Parameter sourcePath ist gesetzt: Input vom Verzeichnis, dass im Verzeichnispfad angegeben ist.
 - Parameter sourcePath ist nicht gesetzt: Input des Parameters
- Basierend auf diesen zwei Möglichkeiten können für beide Operationen verschiedene Outputs verwendet werden:
 - Parameter targetPath ist gesetzt: Das Ergebnis der Operation wird am Speicherort abgelegt, der als Verzeichnispfad angegeben ist.
 - Parameter targetPath ist nicht gesetzt:
 - Eine XML-Datei mit einem Eintrag für jede Input-Datei, die in der Input-Zip-Datei enthalten ist, wird ausgegeben.
 - Der Inhalt der Zip-Datei wird am Speicherort abgelegt, der als Verzeichnispfad angegeben ist.

Output

Der Adapter liefert abhängig von der ausgeführten Operation unterschiedliche Arten von Dokumenten zurück.

- Zip-Operation ohne targetPath: Zip-Datei wird im Adapter-Output bereitgestellt.
- Zip-Operation mit targetPath: Zip-Datei wird im Zielverzeichnis abgelegt.
- Unzip-Operation ohne targetPath: Eine XML-Datei, die die verschiedenen Dateien innerhalb der Zip-Datei enthält, wird im Adapter-Output bereitgestellt. Die XML-Datei hat folgende Struktur:

Beispiel-Output für die Operation Unzip

```
<?xml version="1.0" encoding="UTF-8" ?>
<Zip>
  <File size="209" name="testToZip.xml" comment="&#0;&#0;!"
>PD94bWwgdGVyc2lvngj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz4KPFJvb3RFbGVtZW50P</File>
</Zip>
```

- Unzip-Operation mit `targetPath`: Der Inhalt der Zip-Datei wird im Zielverzeichnis abgelegt.

6.5.2.3 ZIP Folder Compression (Veraltet)

ZIP Folder Compression: Erzeugt ein ZIP-Archiv von einem Ordner und gibt das ZIP-Archiv als Binärdokument mit MIME-Typ *Application/ZIP* aus.



Der *ZIP Folder Compression* Adapter wird mit Version 6.2.0 der X4 Suite abgekündigt und in zukünftigen Versionen der X4 Suite nicht mehr unterstützt. Bitte verwenden Sie stattdessen den [Zip Converter](#).

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>Execute</i> : ZIP-Archiv vom gewählten Ordner erzeugen
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.zipfolderadapter.ZipFolderAdapter</i> : Hauptklasse (Standard)
----------------	--

<i>zipFolderPath</i>	<p>Pfad zum Ordner, von dem ein ZIP-Archiv erstellt werden soll</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige gültige Zeichenkette (z.B. <code>%USER_COLLECTION%\Project\Folder\</code>) • Pfad ohne /bzw. \ am Ende: Das erzeugte ZIP-Archiv enthält als einzigen Unterordner den gewählten Ordner, und darin dessen Ordnerinhalte • Pfad mit /bzw. \ am Ende: Das erzeugte ZIP-Archiv enthält nur die Inhalte des gewählten Ordners, nicht jedoch den Ordner selbst • (<i>nicht gesetzt</i>): Es wird versucht, aus einem Input-XML-Dokument eine Zeichenkette zu ermitteln, die über den XPath-Ausdruck <code>/X4/ZipFolderAdapter/zipfolderpath</code> adressiert wurde
----------------------	--

6.5.2.4 ZIP in Memory Compression (Veraltet)



ZIP in Memory Compression: Erzeugt ein ZIP-Archiv mit einer oder mehreren Dateien aus Daten, die in einer adapterspezifischen XML-Struktur abgelegt sind, und hält die Daten komplett im Arbeitsspeicher – benötigt also keinen Zugriff auf das Dateisystem.

⚠ Der *ZIP in Memory Compression* Adapter wird mit Version 6.2.0 der X4 Suite abgekündigt und in zukünftigen Versionen der X4 Suite nicht mehr unterstützt. Bitte verwenden Sie stattdessen den [Zip Converter](#).

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Zip</i>: ZIP-Archiv aus den Daten im Input-XML-Dokument im Arbeitsspeicher erzeugen • <i>Unzip</i>: ZIP-Archiv in den Arbeitsspeicher entpacken
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.zipinmemory.ZipInMemory</code>: Hauptklasse (Standard)</p>
----------------	---

<i>encoding</i>	Zeichenkodierung für Datei-/Pfadnamen Mögliche Werte: <ul style="list-style-type: none"> • Beliebige gültige Zeichenkodierung (z.B. <i>UTF-8</i>) • <i>CP850</i>: MS-DOS Deutschland, wird von <i>MS Windows</i> so erzeugt (Standard) • (<i>leer</i>): UTF-8 (aus Abwärtskompatibilitäts-Gründen mit älteren <i>X4 Server</i>-Versionen)
-----------------	--

Input

Wenn die Eigenschaft *Operation* auf *Zip* gesetzt ist, erwartet dieser Adapter eine adapterspezifische Input-XML-Struktur. Diese enthält für jede Datei, die im ZIP-Archiv enthalten sein soll, den Dateinamen und den Base64-kodierten Dateiinhalt. Jedes Element `<File>` steht hierbei für eine Datei, jedes Element `<Directory>` für einen leeren Ordner im ZIP-Archiv.

```
<Archiv>
  <File name="Dateiname" comment="Optionaler Kommentar">
    <!--Base64-kodierte Daten-->
  </File>
  <Directory name="Ordnername/" />
</Archiv>
```

Ausgabe

Wenn die Eigenschaft *Operation* auf *Unzip* gesetzt ist, erzeugt dieser Funktions-Adapter aus einem ZIP-Archiv eine adapterspezifische XML-Struktur, beispielsweise:

```
<Archiv>
  <File name="Data.txt">Q29udGVudCBvZiB0aGlzIHRLeHQgZmlsZQ==</File>
  <Directory name="Folder/" />
</Archiv>
```

Erläuterung:

- Jedes Element `<Directory>` steht für einen (leeren) Ordner im ZIP-Archiv.
- Jedes Element `<File>` steht für eine Datei und speichert den Base64-kodierten Dateiinhalt als Wert.
- Attribut *name* gibt den Datei- oder Ordnernamen an.
- Attribut *size* gibt die Dateigröße an, sofern diese im ZIP-Archiv hinterlegt ist.

6.6 Databases

6.6.1 JDBC Connector



JDBC Connector: Stellt eine Verbindung zu einer JDBC-kompatiblen Datenbank her und führt ein oder mehrere Datenbank-Statements aus, die als Input-XML-Dokument mit adapterspezifischen Strukturen übergeben wurden.

Ehemalige Bezeichnung bis Version 4.5.1: *JDBC Database Connection Adapter*



- i** Voraussetzung ist eine im Applikations-Server bereitgestellte JDBC-DataSource, für welche die Verbindungs-URL, Zugangsdaten, Schema sowie Transaktionsverhaltensweisen etc. definiert wurden. Nähere Informationen zum Bereitstellen von JDBC-DataSources entnehmen Sie der Dokumentation Ihres Applikations-Servers.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>execute</i> : Funktions-Adapter ausführen
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.jdbc.DB</i> : Hauptklasse (Standard)
<i>JndiName</i>	JNDI-Name der DataSource i Wenn die Verbindung über den JNDI-Namen hergestellt wird, dann darf der Parameter <i>connectionURL</i> nicht ausgefüllt werden. Mögliche Werte: Gültiger JNDI-Name (z. B. <i>java:/DefaultDS</i>)

<i>ColumnCase</i>	<p>Steuert wie die Spaltennamen ins Ergebnis übernommen werden. Da Datenbanken unterschiedliches Verhalten betreffend der Groß-/Kleinschreibung von Spaltennamen haben, kann man hiermit die Weiterverarbeitung des Ergebnis unabhängig vom Datenbankverhalten machen.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>NO_CHANGE</i> : Ändert die Spaltennamen nicht, d.h. die Schreibweise im Ergebnis ist exakt wie von der Datenbank geliefert (Standard) • <i>TO_LOWER</i> : Ändert die Spaltennamen auf Kleinbuchstaben • <i>TO_UPPER</i> : Ändert die Spaltennamen auf Großbuchstaben <div> <p> Spaltenname, die keine gültigen Elementnamen (NCNAME) darstellen, werden durch <code>_<number></code> (e. g. <code>_2</code>) ersetzt, wo <code><number></code> der Index der Spalte ist.</p> </div>
<i>connectionURL</i>	<p>Verbindungs-URL für die JDBC-Datenbank</p> <p> Wenn die Verbindung über die Verbindungs-URL hergestellt wird, dann darf der Parameter <code>JndiName</code> nicht ausgefüllt werden.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Die Art des Aufbaus hängt von der Datenbank ab, zu der eine Verbindung aufgebaut werden soll. • Beispiel für sqlite: <code>jdbc:sqlite:C:/Temp/mydatabase.db</code>
<i>user</i>	Benutzername zur Authentifizierung bei Verwendung des Parameters <code>connectionURL</code>
<i>password</i>	Passwort zur Authentifizierung bei Verwendung des Parameters <code>connectionURL</code>
<i>rootElementName</i>	<p>Name des Wurzelements des Outputs.</p> <p>Mögliche Werte: Zeichenkette, die für XML-Elementnamen gültig ist. Standard: <i>Data</i></p>

<i>CommitMode</i>	<p>Steuert, welche Variante für Commits verwendet werden soll</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> <p><i>NONE</i> : Keinen Commit durchführen</p> <p><i>AUTO</i> : Automatischer Commit gemäß JDBC-API (d. h. nach jedem Batch)(Standard)</p> <p><i>STATEMENT</i> : Explizites Commit nach jedem Statement (d. h. nach allen Batches für dieses Statement)</p> <p><i>DOCUMENT</i> : Am Ende des Dokuments (d. h. nach Ausführung aller Statements)</p> <p><i>MANAGED</i> : Adapter in Transaktionen nutzen (für extern verwaltete Transaktionen)</p>
<i>AddMetaDataForQueries</i>	<p>(ab X4 4.1 Revision 32408) Steuert, ob Datenbank-Metadaten, siehe JDBC Metadata Explorer, zu jedem <i>Result Set</i> im Ergebnis-XML-Dokument des Adapters ausgegeben werden sollen, wenn der Adapter eine SQL-Abfrage ausführt (nicht der Fall bei UPDATE, DELETE etc.). Im Batch-Modus (StatementBatch, PreparedStatementBatch und CallableStatementBatch) ist dies ebenfalls nicht möglich.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> <p><i>ja</i> : Dem Ergebnisdokument Datenbank-Metadaten hinzufügen</p> <p><i>nein</i> : Ergebnisdokument ohne Datenbank-Metadaten ausgeben (Standard)</p>

Statuswerte

<i>1</i>	<p>Die Abfrage hat Datensätze geliefert bzw. die Anweisung hat Datensätze geändert</p> <p>Der Status 1 wird geliefert, wenn mehrere Abfragen bzw. Anweisungen im Dokument enthalten sind und mindestens eine Abfrage ein Ergebnis geliefert oder verändert hat.</p>
----------	---

0	<p>Die Abfrage hat keine Ergebnisse geliefert oder die Anweisung hat keine Datensätze geändert</p> <p>Der Status 0 wird geliefert, wenn mehrere Abfragen bzw. Anweisungen im Dokument enthalten sind und keine Abfrage ein Ergebnis geliefert oder verändert hat.</p>
-1	<p>Technischer Fehler: genauere Informationen zur Fehlerursache entnehmen Sie dem Server-Log</p>

Input

Der *JDBC Connector* erwartet spezifische Input-XML-Strukturen, die Datenbank-Statements enthalten. Diese Input-XML-Strukturen können über XSL-Mappings in Abhängigkeit von den verarbeiteten XML-Daten dynamisch generiert werden.

 **Siehe auch:**


6.6.1.1 Statement: Direktes SQL

Input

```
<Beliebiger Elementname>
  <statement resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      SQL-Anweisungen
    </sql>
  </statement>
</Beliebiger Elementname>
```

Für jedes Element `<statement/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.
- Es enthält mindestens ein Element `<sql/>`, das jeweils eine SQL-Anweisung enthält.

 Falls Sie innerhalb der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden `CDATA`-Block, z. B. `<![CDATA[SELECT * FROM TABLE WHERE ID<1]]>`


Beispiel

```
<select>
  <statement resulttag="result" timezone="UTC" format="dd.MM.yyyy">
    <sql>
      <![CDATA[SELECT * FROM JMS_USERS]]>
    </sql>
  </statement>
</select>
```

Output

Wurde die SQL-Anweisung erfolgreich ausgeführt, gibt der Datenbankverbindungs-Baustein ein XML-Dokument mit folgender Struktur zurück, das die Antwort der Datenbank als Zeichenkette (String) enthält:

```
<Data>
  <Ergebniselementname>
    <Feld1><!--Wert--></Feld1>
    <Feldn><!--Wert--></Feldn>
  </Ergebniselementname>
</Data>
```

 Bei SELECT-Statements zum Zugriff auf Datenbanktabellen mit Spaltennamen, die keine gültigen XML-Namen darstellen, werden im Ergebnis-XML-Dokument synthetische Elementnamen erzeugt, um die Wohlgeformtheit des XML-Dokuments sicherzustellen.

6.6.1.2 StatementBatch: Direktes SQL im Batch-Modus

Input

Der *JDBC Connector* führt mehrere SQL-Anweisungen auf einmal im Batch-Modus aus, wenn sie in folgender Input-XML-Struktur abgelegt werden. Dabei sind lediglich SQL-Anweisungen möglich, die kein ResultSet zurückliefern, also z. B. keine Abfragen mit SELECT:

```
<Beliebiger Elementname>
  <statementbatch resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      SQL-Anweisung 1
    </sql>
    <sql>
      SQL-Anweisung n
    </sql>
  </statementbatch>
</Beliebiger Elementname>
```

Für jedes Element <statementbatch/> gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.

- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.
- Es enthält mindestens ein Element `<sql/>`, das jeweils eine SQL-Anweisung enthält.

i Falls Sie innerhalb der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden `CDATA`-Block, z. B. `<![CDATA[UPDATE PROCESS SET X4_VERSION=4 WHERE X4_DURATION=100]]>`

Beispiel

```
<select>
  <statementbatch resulttag="Ergebniselementname" timezone="UTC" format="dd.MM.yyyy">
    <sql>
      <![UPDATE PROCESS SET X4_VERSION=4 WHERE X4_DURATION=100]]>
    </sql>
    <sql>
      <![UPDATE PROCESS SET X4_VERSION=4 WHERE X4_DURATION=200]]>
    </sql>
  </statementbatch>
</select>
```

Output

Wurde die Batch-Anweisung erfolgreich ausgeführt, gibt der Datenbankverbindungs-Baustein folgende XML-Struktur zurück:

```
<Data>
  <Ergebniselementname>
    <!--Status der Batch-Operation-->
  </Ergebniselementname>
</Data>
```

Als Status der Batch-Operation sind folgende Ausgaben möglich:

- *(Anzahl der Zeilen)* `rows affected.` oder
- `SUCCESS_NO_INFO` oder
- `EXECUTE_FAILED`

6.6.1.3 PreparedStatement: SQL mit Werten

Der *JDBC Connector* führt SQL-Anweisungen mit dynamischen Input-Daten aus, wenn sie in folgender Input-XML-Struktur abgelegt werden.

i Alternativ können Sie ab Release 5.0 der X4 Suite auch SQL-Anweisungen mit benannten Parametern ausführen, siehe [NamedPreparedStatement: SQL mit benannten Parametern](#).

Input


```

<Beliebiger Elementname>
  <preparedstatement resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      <!--SQL-Anweisungen-->
    </sql>
    <columns>
      <column mode="in" name="frei wählbarer NCName"
        index="Platzhalter-Index" type="Datentyp"
        timezone="Zeitzone" format="Datumsformat">
      </column>
    </columns>
    <rowset>
      <row>
        <Name><!--Wert--></Name>
      </row>
    </rowset>
  </preparedstatement>
</Beliebiger Elementname>

```

Für jedes Element `<preparedstatement/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.
- Es enthält mindestens ein Element `<sql/>`.

i Falls Sie innerhalb der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden `CDATA`-Block, z. B. `<![CDATA[SELECT * FROM TABLE WHERE ID<1]]>`

Für jedes Element `<column/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n$ -mal)
- Das optionale Attribut `mode` gibt die "Richtung" des Datenstroms an (*in*: Input-Parameter für die Datenbank; *out*: Ergebnisparameter aus der Datenbank; *inout*: sowohl Input-Parameter als auch Ergebnisparameter; Standard: *in*).
- Das Attribut `name` gibt einen frei wählbaren Namen (*NCName*) für ein Parameter an.
- Das Attribut `index` gibt einen ganzzahligen Platzhalterindex für den in Attribut `name` definierten Parameter an.
- Das optionale Attribut `type` gibt den Datentyp an (z. B. *INTEGER* oder *BLOB*; Standard: *VARCHAR*).
- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.

Für jedes Element `<rowset/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n\text{-mal}$)
- Es enthält mindestens ein Element `<row/>`, das die Statement-Ausführung steuert.

Für jedes Element `<row/>` gilt:

- Es darf beliebig viele Elemente enthalten, die den Namen der Tabellenspalte oder den im SQL-Statement gesetzten Spalten-Alias tragen.
- Die darin liegenden Elemente enthalten die jeweiligen Werte.

Beachten Sie Folgendes

- Die Attribute `mode` und `format` sind in *X4 Server* Revision 31469 Pflichtparameter; ab neueren Versionen sind diese optional.
- Zulässige Datentypen für das Attribut `type` zum Setzen (Input-Parameter) und Lesen (*ResultSet*) von Werten sind: *BIGINT*, *BINARY*, *BIT*, *BOOLEAN*, *BLOB*, *CHAR*, *CLOB*, *DATE*, *DECIMAL*, *DOUBLE*, *FLOAT*, *INTEGER*, *LONGNVARCHAR*, *LONGVARBINARY*, *LONGVARCHAR*, *NCHAR*, *NCLOB*, *NULL*, *NUMERIC*, *NVARCHAR*, *REAL*, *SMALLINT*, *SQLXML*, *TIME*, *TIMESTAMP*, *TINYINT*, *VARBINARY* und *VARCHAR* (Standard), *XML* (alias für *SQLXML*).

Beispiel

```
<Rootelement>
  <preparedstatement resulttag="abc" timezone="MEZ"
    format="dd.MM.yyyy">
    <sql>
    <![CDATA[SELECT * FROM JMS_USERS WHERE USERID=?]]>
    </sql>
    <columns>
      <column mode="in" name="id" index="1"
        type="INTEGER">
      </column>
    </columns>
    <rowset>
      <row>
        <id>23</id>
      </row>
    </rowset>
  </preparedstatement>
</Rootelement>
```

Output

Wurde die SQL-Anweisung erfolgreich ausgeführt, gibt der Adapter ein XML-Dokument mit folgender Struktur zurück, das die Antwort der Datenbank als Zeichenkette (String) enthält:

```
<Data>
  <Ergebniselementname>
    <Feld1><!--Wert--></Feld1>
    <Feldn><!--Wert--></Feldn>
  </Ergebniselementname>
</Data>
```

6.6.1.4 PreparedStatementBatch: SQL mit Werten im Batch-Modus

Der *JDBC Connector* führt mehrere SQL-Anweisungen auf einmal im Batch-Modus aus, wenn sie in folgender Input-XML-Struktur abgelegt werden. Dabei sind lediglich SQL-Anweisungen möglich, die kein Result Set zurückliefern, also z.B. keine Abfragen mit SELECT.

i Alternativ können Sie ab Release 5.0 der X4 Suite auch SQL-Anweisungen mit benannten Parametern mit oder ohne Batch-Modus ausführen, siehe [NamedPreparedStatementBatch: SQL mit benannten Parametern im Batch-Modus](#).

Input

```
<Beliebiger Elementname>
  <preparedstatementbatch resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      <!-- SQL-Anweisung -->
    </sql>
    <columns>
      <column mode="in" name="frei wählbarer NCName"
        index="Platzhalter-Index" type="Datentyp"
        timezone="Zeitzone" format="Datumsformat">
      </column>
    </columns>
    <rowset>
      <row>
        <Name><!--Wert--></Name>
      </row>
      <!-- Weitere <row>-Elemente für Batch-Modus -->
    </rowset>
  </preparedstatementbatch>
</Beliebiger Elementname>
```

Für jedes Element `<preparedstatementbatch/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.
- Es enthält mindestens ein Element `<sql/>`.

i Falls Sie innerhalb der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden *CDATA*-Block, z. B. `<![CDATA[UPDATE PROCESS SET X4_VERSION=4 WHERE X4_DURATION=?]]>`

Für jedes Element `<column/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n$ -mal)
- Das optionale Attribut `mode` gibt die "Richtung" des Datenstroms an (*in*: Input-Parameter für die Datenbank; *out*: Ergebnisparameter aus der Datenbank; *inout*: sowohl Input-Parameter als auch Ergebnisparameter; Standard: *in*).
- Das Attribut `name` gibt einen frei wählbaren Namen (*NCName*) für ein Parameter an.
- Das Attribut `index` gibt einen ganzzahligen Platzhalterindex für den in Attribut `name` definierten Parameter an.
- Das optionale Attribut `type` gibt den Datentyp an (z. B. *INTEGER* oder *BLOB*; Standard: *VARCHAR*).
- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. *GMT+2*.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.

Für jedes Element `<rowset/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n$ -mal)
- Es enthält beliebig viele Elemente `<row/>`, welche die Statement-Ausführung steuern.

Für jedes Element `<row/>` gilt:

- Es darf beliebig viele Elemente enthalten, die den Namen der Tabellenspalte oder den im SQL-Statement gesetzten Spalten-Alias tragen.
- Die darin liegenden Elemente enthalten die jeweiligen Werte.

Beachten Sie Folgendes

- Die Attribute `mode` und `format` sind in *X4 Server* Revision 31469 Pflichtparameter; ab neueren Versionen sind diese optional.
- Zulässige Datentypen für das Attribut `type` zum Setzen (Input-Parameter) und Lesen (*ResultSet*) von Werten sind: *BIGINT*, *BINARY*, *BIT*, *BOOLEAN*, *BLOB*, *CHAR*, *CLOB*, *DATE*, *DECIMAL*, *DOUBLE*, *FLOAT*, *INTEGER*, *LONGNVARCHAR*, *LONGVARBINARY*, *LONGVARCHAR*, *NCHAR*, *NCLOB*, *NULL*, *NUMERIC*, *NVARCHAR*, *REAL*, *SMALLINT*, *SQLXML*, *TIME*, *TIMESTAMP*, *TINYINT*, *VARBINARY* und *VARCHAR* (Standard), *XML* (alias für *SQLXML*).

Beispiel

```

<Rootelement>
  <preparedstatementbatch resulttag="Ergebniselementname" timezone="MEZ"
    format="dd.MM.yyyy">
    <sql>
    <![CDATA[ UPDATE PROCESS SET X4_VERSION=4 WHERE X4_DURATION=? ]]>
    </sql>
    <columns>
      <column mode="in" name="id" index="1"
        type="INTEGER">
      </column>
    </columns>
    <rowset>
      <row>
        <id>23</id>
      </row>
      <row>
        <id>25</id>
      </row>
    </rowset>
  </preparedstatementbatch>
</Rootelement>

```

Output

Wurde die Batch-Anweisung erfolgreich ausgeführt, gibt der Datenbankverbindungs-Baustein folgende XML-Struktur zurück:

```

<Data>
  <Ergebniselementname>
    <!--Status der Batch-Operation-->
  </Ergebniselementname>
</Data>

```

Als Status der Batch-Operation sind folgende Ausgaben möglich:

- (Anzahl der Zeilen) rows affected. oder
- SUCCESS_NO_INFO oder
- EXECUTE_FAILED

6.6.1.5 CallableStatement: Stored Procedures

Der *JDBC Connector* führt Stored Procedures über *JDBC Callablestatements* mit dynamischen Input-Daten aus, wenn sie in folgender Input-XML-Struktur abgelegt werden.

Input

```

<Beliebiger Elementname>
  <callablestatement resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      <!--SQL-Anweisungen -->
    </sql>
    <columns>
      <column mode="Modus" name="frei wählbarer Name" index="Platzhalter"
        type="Datentyp" timezone="Zeitzone" format="Datumsformat">
      </column>
    </columns>
    <rowset>
      <row>
        <Name><!--Wert--></Name>
      </row>
    </rowset>
  </callablestatement>
</Beliebiger Elementname>

```

Für jedes Element `<callablestatement/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.
- Es enthält mindestens ein Element `<sql/>`.

i Falls Sie innerhalb der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden `CDATA`-Block, z. B. `<![CDATA[SELECT * FROM TABLE WHERE ID<1]]>`

Für jedes Element `<column/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n$ -mal)
- Das optionale Attribut `mode` gibt die "Richtung" des Datenstroms an (*in*: Input-Parameter für die Datenbank; *out*: Ergebnisparameter aus der Datenbank; *inout*: sowohl Input-Parameter als auch Ergebnisparameter; siehe <http://download.oracle.com/javase/1.5.0/docs/guide/jdbc/getstart/callablestatement.html>); Standard: *in*.
- Das Attribut `name` gibt einen frei wählbaren Namen für ein Parameter an.
- Das Attribut `index` gibt einen ganzzahligen Platzhalterindex für den in Attribut `name` definierten Parameter an.
- Das optionale Attribut `type` gibt den Datentyp an (zulässige Datentypen siehe Hinweis unten; Standard: `VARCHAR`).
- Das Attribut `name` gibt einen frei wählbaren Namen für ein Parameter an.
- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.

Für jedes Element `<rowset/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n$ -mal).
- Es enthält beliebig viele Elemente `<row/>`, die die Statement-Ausführung steuern.

Für jedes Element `<row/>` gilt:

- Es darf beliebig viele Elemente enthalten, die den Namen der Tabellenspalte oder den im SQL-Statement gesetzten Spalten-Alias tragen.
- Die darin liegenden Elemente enthalten die jeweiligen Werte.

Beachten Sie Folgendes

- Die Attribute `mode` und `format` sind in *X4 Server* Revision 31469 Pflichtparameter; ab neueren Versionen sind diese optional.
- Zulässige Datentypen für das Attribut `type` zum Setzen (Input-Parameter) und Lesen (*ResultSet*) von Werten sind: *BIGINT*, *BINARY*, *BIT*, *BOOLEAN*, *BLOB*, *CHAR*, *CLOB*, *DATE*, *DECIMAL*, *DOUBLE*, *FLOAT*, *INTEGER*, *LONGNVARCHAR*, *LONGVARBINARY*, *LONGVARCHAR*, *NCHAR*, *NCLOB*, *NULL*, *NUMERIC*, *NVARCHAR*, *REAL*, *SMALLINT*, *SQLXML*, *TIME*, *TIMESTAMP*, *TINYINT*, *VARBINARY* und *VARCHAR* (Standard), *XML* (alias für *SQLXML*).

Beispiel

```
<Rootelement>
  <callablestatement resulttag="abc" timezone="MEZ"
    format="dd.MM.yyyy">
    <sql>
      <![CDATA[ {call getData(?)} ]]>
    </sql>
    <columns>
      <column mode="out" name="id" index="1"
        type="INTEGER">
      </column>
    </columns>
    <rowset>
      <row>
        <id>TestUser</id>
      </row>
    </rowset>
  </callablestatement>
</Rootelement>
```

Output

Wurde die SQL-Anweisung erfolgreich ausgeführt, gibt der Datenbankverbindungs-Baustein ein XML-Dokument mit folgender Struktur zurück, das die Antwort der Datenbank als Zeichenkette (String) enthält:

```
<Data>
  <Ergebniselementname>
    <Feld1><!--Wert--></Feld1>
    <Feldn><!--Wert--></Feldn>
  </Ergebniselementname>
</Data>
```

6.6.1.6 CallableStatementBatch: Stored Procedures im Batch-Modus

Der *JDBC Connector* führt Stored Procedures über *JDBC Callablestatements* mit dynamischen Input-Daten aus, wenn sie in folgender Input-XML-Struktur abgelegt werden. Dabei sind lediglich SQL-Anweisungen möglich, die kein ResultSet zurückliefern, also z. B. keine Abfragen mit SELECT.

Input

```
<Beliebiger Elementname>
  <callablestatementbatch resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      <!--SQL-Anweisungen -->
    </sql>
    <columns>
      <column mode="Modus" name="frei wählbarer Name" index="Platzhalter"
        type="Datentyp" timezone="Zeitzone" format="Datumsformat">
      </column>
    </columns>
    <rowset>
      <row>
        <Name><!--Wert--></Name>
      </row>
    </rowset>
  </callablestatementbatch>
</Beliebiger Elementname>
```

Für jedes Element `<callablestatementbatch/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.
- Es enthält mindestens ein Element `<sql/>`.

i Falls Sie innerhalb der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden *CDATA*-Block, z. B. `<![CDATA[SELECT * FROM TABLE WHERE ID<1]]`

Für jedes Element `<column/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n$ -mal)

- Das optionale Attribut `mode` gibt die "Richtung" des Datenstroms an (*in*: Input-Parameter für die Datenbank; *out*: Ergebnisparameter aus der Datenbank; *inout*: sowohl Input-Parameter als auch Ergebnisparameter; siehe <http://download.oracle.com/javase/1.5.0/docs/guide/jdbc/getstart/callablestatement.html>); Standard: *in*.
- Das Attribut `name` gibt einen frei wählbaren Namen für ein Parameter an.
- Das Attribut `index` gibt einen ganzzahligen Platzhalterindex für den in Attribut `name` definierten Parameter an.
- Das optionale Attribut `type` gibt den Datentyp an (zulässige Datentypen siehe Hinweis unten; Standard: *VARCHAR*).
- Das Attribut `name` gibt einen frei wählbaren Namen für ein Parameter an.
- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. *GMT+2*.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.

Für jedes Element `<rowset/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n$ -mal).
- Es enthält beliebig viele Elemente `<row/>`, die die Statement-Ausführung steuern.

Für jedes Element `<row/>` gilt:

- Es darf beliebig viele Elemente enthalten, die den Namen der Tabellenspalte oder den im SQL-Statement gesetzten Spalten-Alias tragen.
- Die darin liegenden Elemente enthalten die jeweiligen Werte.

Beachten Sie Folgendes

- Die Attribute `mode` und `format` sind in *X4 Server* Revision 31469 Pflichtparameter; ab neueren Versionen sind diese optional.
- Zulässige Datentypen für das Attribut `type` zum Setzen (Input-Parameter) und Lesen (*ResultSet*) von Werten sind: *BIGINT*, *BINARY*, *BIT*, *BOOLEAN*, *BLOB*, *CHAR*, *CLOB*, *DATE*, *DECIMAL*, *DOUBLE*, *FLOAT*, *INTEGER*, *LONGNVARCHAR*, *LONGVARBINARY*, *LONGVARCHAR*, *NCHAR*, *NCLOB*, *NULL*, *NUMERIC*, *NVARCHAR*, *REAL*, *SMALLINT*, *SQLXML*, *TIME*, *TIMESTAMP*, *TINYINT*, *VARBINARY* und *VARCHAR* (Standard), *XML* (alias für *SQLXML*).

Beispiel

```

<Rootelement>
  <callablestatementbatch resulttag="abc" timezone="MEZ"
    format="dd.MM.yyyy">
    <sql>
    <![CDATA[ {call getData(?)} ]]>
    </sql>
    <columns>
      <column mode="out" name="id" index="1"
        type="INTEGER">
      </column>
    </columns>
    <rowset>
      <row>
        <id>TestUser</id>
      </row>
    </rowset>
  </callablestatementbatch>
  <callablestatementbatch resulttag="abc" timezone="MEZ"
    format="dd.MM.yyyy">
    <sql>
    <![CDATA[ {call getData(?)} ]]>
    </sql>
    <columns>
      <column mode="out" name="id" index="1"
        type="INTEGER">
      </column>
    </columns>
    <rowset>
      <row>
        <id>TestUser</id>
      </row>
    </rowset>
  </callablestatementbatch>
</Rootelement>

```

Output

Wurde die Batch-Anweisung erfolgreich ausgeführt, gibt der Datenbankverbindungs-Baustein folgende XML-Struktur zurück:

```

<Data>
  <Ergebniselementname>
    <!--Status der Batch-Operation-->
  </Ergebniselementname>
</Data>

```

Als Status der Batch-Operation sind folgende Ausgaben möglich:

- (Anzahl der Zeilen) rows affected. oder
- SUCCESS_NO_INFO oder
- EXECUTE_FAILED

6.6.1.7 NamedPreparedStatement: SQL mit benannten Parametern

Der *JDBC Connector* führt SQL-Anweisungen mit benannten Parametern aus, wenn sie in folgender Input-XML-Struktur abgelegt werden.

Input

```
<Beliebiger Elementname>
  <namedpreparedstatement resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      <!--SQL-Anweisungen (optional innerhalb eines CDATA-Blocks) -->
      <!--Je Parameter ein XML-Element, gleichnamig zum entsprechenden Kindelement
in <row> -->
    </sql>
    <rowset>
      <row>
        <Name><!--Wert--></Name>
      </row>
    </rowset>
  </namedpreparedstatement>
</Beliebiger Elementname>
```

Für jedes Element `<namedpreparedstatement/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an und bestimmt das Format der zurückgegebenen Datumswerten.
- Es enthält ein Element `<sql/>`.

Für jedes XML-Element eines benannten Parameters innerhalb der SQL-Anweisung in Element `<sql/>` gilt:

- Der Elementname gibt den Parameter-Namen an, der mit entsprechenden Element innerhalb von `<row/>` übereinstimmt.
- Das optionale Attribut `mode` gibt die "Richtung" des Datenstroms an (*in*: Input-Parameter für die Datenbank; *out*: Ergebnisparameter aus der Datenbank; *inout*: sowohl Input-Parameter als auch Ergebnisparameter; Standard: *in*).
- Das optionale Attribut `type` gibt den Datentyp an, z. B. `INTEGER`, `BLOB` oder `VARCHAR` (Standard).
- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an und gibt das erwartete Format für den Wert innerhalb des entsprechenden `<row>`-Elements vor.

- ❗ Falls Sie in der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden *CDATA*-Block, z. B. `<![CDATA[SELECT * FROM TABLE WHERE ID<]]><id type="INTEGER" />.`

Für jedes Element `<rowset/>` gilt:

- Es darf beliebig häufig verwendet werden (*0 – n-mal*)
- Es enthält beliebig viele Elemente `<row/>`, die die Statement-Ausführung steuern.

Für jedes Element `<row/>` gilt:

- Es muss mindestens ein Element enthalten, das dem Elementnamen im SQL-Statement entspricht.
- Die darin liegenden Elemente enthalten die jeweiligen Werte.

Beachten Sie für jedes Datenelement innerhalb von `<row/>`, dass im Elementinhalt keine unbeabsichtigten Zeilenumbrüche definiert wurden (z.B. durch Editor-Formatierung). Andernfalls würden diese Zeilenumbrüche in der erzeugten SQL-Anweisung als `\n` angefügt.

❗ **Beachten Sie Folgendes**

Zulässige Datentypen für das Attribut `type` zum Setzen (Input-Parameter) und Lesen (*ResultSet*) von Werten sind: *BIGINT*, *BINARY*, *BIT*, *BOOLEAN*, *BLOB*, *CHAR*, *CLOB*, *DATE*, *DECIMAL*, *DOUBLE*, *FLOAT*, *INTEGER*, *LONGNVARCHAR*, *LONGVARBINARY*, *LONGVARCHAR*, *NCHAR*, *NCLOB*, *NULL*, *NUMERIC*, *NVARCHAR*, *REAL*, *SMALLINT*, *SQLXML*, *TIME*, *TIMESTAMP*, *TINYINT*, *VARBINARY* und *VARCHAR* (Standard), *XML* (alias für *SQLXML*).

Beispiel

```
<RootElement>
  <namedpreparedstatement resulttag="namedPreparedStatementWithCDATA"
    timezone="UTC" format="dd.MM.yyyy">
    <sql>
      SELECT * FROM PROCESS WHERE X4_DURATION= <duration type="INTEGER"
        mode="in" /> AND X4_PROCESSNAME= <processname
          type="VARCHAR" mode="in" />
    </sql>
    <rowset>
      <row>
        <duration>101</duration>
        <processname>Project/Process/Path/Process.wrf</processname>
      </row>
    </rowset>
  </namedpreparedstatement>
</RootElement>
```

Output

Wurde die SQL-Anweisung erfolgreich ausgeführt, gibt der Adapter ein XML-Dokument mit folgender Struktur zurück, das die Antwort der Datenbank als Zeichenkette (String) enthält:

```
<Data>
  <Ergebniselementname>
    <Feld1><!--Wert--></Feld1>
    <Feldn><!--Wert--></Feldn>
  </Ergebniselementname>
</Data>
```

6.6.1.8 NamedPreparedStatementBatch: SQL mit benannten Parametern im Batch-Modus

Der *JDBC Connector* führt mehrere SQL-Anweisungen auf einmal im Batch-Modus aus, wenn sie in folgender Input-XML-Struktur jeweils in einem `<namedpreparedstatementbatch>`-Block abgelegt werden. Dabei sind lediglich SQL-Anweisungen möglich, die kein `ResultSet` zurückliefern, also z. B. keine Abfragen mit `SELECT`.

Input

```
<Beliebiger Elementname>
  <namedpreparedstatementbatch resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      <!--SQL-Anweisungen (optional innerhalb eines CDATA-Blocks) -->
      <!--Je Parameter ein XML-Element, gleichnamig zum entsprechenden Kindelement
in <row> -->
    </sql>
    <rowset>
      <row>
        <Name><!--Wert--></Name>
      </row>
      <!-- Weitere <row>-Elemente für Batch-Modus -->
    </rowset>
  </namedpreparedstatementbatch>
</Beliebiger Elementname>
```

Für jedes Element `<namedpreparedstatementbatch/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an und bestimmt das Format der zurückgegebenen Datumswerten.
- Es enthält mindestens ein Element `<sql/>`.

Für jedes XML-Element eines benannten Parameters innerhalb der SQL-Anweisung im Element `<sql/>` gilt:

- Der Elementname gibt den Parameter-Namen an, der mit entsprechenden Element innerhalb von `<row/>` übereinstimmt.
- Das optionale Attribut `mode` gibt die "Richtung" des Datenstroms an (*in*: Input-Parameter für die Datenbank; *out*: Ergebnisparameter aus der Datenbank; *inout*: sowohl Input-Parameter als auch Ergebnisparameter; Standard: *in*).
- Das optionale Attribut `type` gibt den Datentyp an, z. B. `INTEGER`, `BLOB` oder `VARCHAR`(Standard).

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an und gibt das erwartete Format für den Wert innerhalb des entsprechenden `<row>`-Elements vor.

i Falls Sie in der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden `CDATA`-Block, z. B. `<![CDATA[SELECT * FROM TABLE WHERE ID<]]><id type="INTEGER" />`.

Für jedes Element `<rowset/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n$ -mal)
- Es enthält beliebig viele Elemente `<row/>`, die die Statement-Ausführung steuern.

Für jedes Element `<row/>` gilt:

- Es muss mindestens ein Element enthalten, das dem Elementnamen im SQL-Statement entspricht.
- Die darin liegenden Elemente enthalten die jeweiligen Werte.

Beachten Sie für jedes Datenelement innerhalb von `<row/>`, dass im Elementinhalt keine unbeabsichtigten Zeilenumbrüche definiert wurden (z.B. durch Editor-Formatierung). Andernfalls würden diese Zeilenumbrüche in der erzeugten SQL-Anweisung als `\n` angefügt.

i Beachten Sie Folgendes

Zulässige Datentypen für das Attribut `type` zum Setzen (Input-Parameter) und Lesen (*ResultSet*) von Werten sind: *BIGINT*, *BINARY*, *BIT*, *BOOLEAN*, *BLOB*, *CHAR*, *CLOB*, *DATE*, *DECIMAL*, *DOUBLE*, *FLOAT*, *INTEGER*, *LONGNVARCHAR*, *LONGVARBINARY*, *LONGVARCHAR*, *NCHAR*, *NCLOB*, *NULL*, *NUMERIC*, *NVARCHAR*, *REAL*, *SMALLINT*, *SQLXML*, *TIME*, *TIMESTAMP*, *TINYINT*, *VARBINARY* und *VARCHAR* (Standard), *XML* (alias für *SQLXML*).

Beispiel

```

<Rootelement>
  <namedpreparedstatementbatch resulttag="namedPreparedStatementBatchWithCData"
    timezone="UTC" format="dd.MM.yyyy">
    <sql>
      <![CDATA[UPDATE PROCESS SET X4_VERSION=4 WHERE X4_DURATION=]]>
      <duration type="INTEGER" mode="inout" />
    </sql>
    <rowset>
      <row>
        <duration>101</duration>
      </row>
      <row>
        <duration>9</duration>
      </row>
    </rowset>
  </namedpreparedstatementbatch>
</Rootelement>

```

Output

Wurde die SQL-Anweisung erfolgreich ausgeführt, gibt der Adapter ein XML-Dokument mit folgender Struktur zurück:

```

<Data>
  <Ergebniselementname count="Anzahl der betroffenen Datenbankeinträge">
    <!--Status der Batch-Operation-->
  </Ergebniselementname>
</Data>

```

Als Status der Batch-Operation sind folgende Ausgaben möglich:

- (Anzahl der Zeilen) rows affected. oder
- SUCCESS_NO_INFO oder
- EXECUTE_FAILED

6.6.1.9 Durchleiten von XML-Strukturen

Der *JDBC Connector* ermöglicht das Durchleiten von XML-Strukturen. In einem Input-XML-Dokument können Sie dabei zusätzlich zu den SQL-Statement-Strukturen eigene XML-Strukturen definieren, die der Adapter direkt wieder ausgibt. Damit lässt sich in vielen Fällen die Prozessmodellierung vereinfachen und die XML-Strukturen können gemeinsam mit den Ergebnissen der SQL-Anweisung(en) weiterverarbeitet werden.

Input

Im Input-XML-Dokument können Sie mit folgendem Statement, das als Kind des Wurzel-Elements definiert wird, eine oder mehrere XML-Struktur(en) durchleiten:

```

<copy resulttag="Name des Elternelements">
  Beliebige XML-Struktur
</copy>

```

Output

Die XML-Struktur, die innerhalb des `<copy>`-Elements definiert wurde, wird vom *JDBC Connector* ausgegeben, jedoch ohne Processing Instructions und Kommentare. Dabei wird die XML-Struktur von einem Element umschlossen, das in der Input-Struktur im Attribut `resulttag` angegeben wurde.

```
<Name des Elternelements>
  Input-XML-Struktur
</Name des Elternelements>
```

Beispiel

Im folgenden Input-XML-Dokument wird neben einem SQL-Statement eine XML-Struktur zur Durchleitung definiert:

```
<Root>
  <statement resulttag="result" timezone="UTC" format="dd.MM.yyyy">
    <sql> <![CDATA[SELECT * FROM JMS_USERS]]> </sql>
  </statement>
  <copy resulttag="Content">
    <Data>Text</Data>
  </copy>
</Root>
```

Der *JDBC Connector* gibt daraufhin z. B. folgendes Ergebnis-XML-Dokument aus, das die XML-Struktur mit dem angegebenen Elternelement enthält:

```
<Data>
  <result>
    <USERID>user</USERID>
    <PASSWD>password</PASSWD>
    <CLIENTID isNull="true" />
  </result>
  <Content>
    <Data>Text</Data>
  </Content>
</Data>
```

6.6.1.10 Abrufen von Datenbank-Metadaten

Mit dem Adapter-Parameter `AddMetaDataForQueries` (ab *X4 Server Revision 32408*) können Sie steuern, ob Datenbank-Metadaten zu den entsprechenden *Result Sets* im Ergebnis-XML-Dokument innerhalb eines XML-Elements ermittelt und gemeinsam mit den Ergebnissen der SQL-Abfrage ausgegeben werden sollen.

Der Name jedes XML-Elements, das die entsprechenden Metadaten enthält, setzt sich aus dem Wert des Attributs `resulttag` im Input-Dokument und dem Zusatz `-meta` zusammen, z. B. `<result-meta>`.

❗ Bei SQL-Statements, die kein *Result Set* ausgeben (z. B. bei UPDATE oder DELETE), können keine Metadaten ausgegeben werden. Im Batch-Modus des Adapters (StatementBatch, PreparedStatementBatch und CallableStatementBatch) ist dies ebenfalls nicht möglich.

Beispiel

Im folgenden Input-XML-Dokument wird eine SQL-Abfrage und als Ergebniselement der Name `result` definiert:

```
<Root>
  <statement resulttag="result">
    <sql>
      SELECT X4USER, STARTTIME FROM HISTORY WHERE PID='1279017294476'
    </sql>
  </statement>
</Root>
```


Der *JDBC Connector* gibt daraufhin z. B. folgendes Ergebnis-XML-Dokument aus, welches das Ergebnis der SQL-Abfrage sowie die Metadaten der *Result Sets* enthält:

```
<Data>
  <result-meta columns="2">
    <Column name="X4USER" label="X4USER" type="VARCHAR"
      nullable="true" auto-increment="false" read-only="false"
      writable="false" display-size="255" case-sensitive="true"
      searchable="true" signed="false" />
    <Column name="STARTTIME" label="STARTTIME" type="TIMESTAMP"
      nullable="true" auto-increment="false" read-only="false"
      writable="false" display-size="26" case-sensitive="false"
      searchable="true" signed="false" />
  </result-meta>
  <result>
    <X4USER>1</X4USER>
    <STARTTIME>2010-07-13 12:34:54.553</STARTTIME>
  </result>
</Data>
```

6.6.2 JDBC Iterator





JDBC Iterator: Stellt eine Verbindung zu einer JDBC-kompatiblen Datenbank her und führt ein oder mehrere Datenbank-Statements aus, die als Input-XML-Dokument mit adapterspezifischen Strukturen übergeben wurden. Der Iterator ruft die Daten blockweise ab und stellt diese auch blockweise zur Verarbeitung bereit. Dabei bleibt die Datenbankverbindung solange offen, bis alle Daten abgerufen wurden.


-  Voraussetzung ist eine im Applikations-Server bereitgestellte JDBC-DataSource, für welche die Verbindungs-URL, Zugangsdaten, Schema sowie Transaktionsverhaltensweisen etc. definiert wurden. Nähere Informationen zum Bereitstellen von JDBC-DataSources entnehmen Sie der Dokumentation Ihres Applikations-Servers.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Iterate</i>: Datenmenge abhängig von dem im Parameter <i>fetchSize</i> angegebenen Wert aus der Datenbank auslesen und in einem Ergebnis-Dokument ausgeben • <i>Clean</i>: Iterator eines <i>JDBC Iterators</i> zurücksetzen, der dieselbe ID besitzt wie der gewünschte <i>JDBC Iterator</i> mit der Operation <i>Iterate</i> (siehe Parameter <i>id</i>)
------------------	---

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.jdbc.JdbcIterator</code>: Hauptklasse (Standard)</p>
<i>JndiName</i>	<p>JNDI-Name der DataSource</p> <p> Wird die Verbindung über den JNDI-Namen hergestellt, darf der Parameter <code>connectionURL</code> nicht ausgefüllt werden.</p> <p>Mögliche Werte: Gültiger JNDI-Name (z. B. <code>java:/DefaultDS</code>)</p>
<i>ColumnCase</i>	<p>Steuert wie die Spaltennamen ins Ergebnis übernommen werden. Da Datenbanken unterschiedliches Verhalten betreffend der Groß-/Kleinschreibung von Spaltennamen haben, kann man hiermit die Weiterverarbeitung des Ergebnisses unabhängig vom Datenbankverhalten machen.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>NO_CHANGE</i>: Ändert die Spaltennamen nicht, d.h. die Schreibweise im Ergebnis ist exakt wie von der Datenbank geliefert (Standard) • <i>TO_LOWER</i>: Ändert die Spaltennamen auf Kleinbuchstaben • <i>TO_UPPER</i>: Ändert die Spaltennamen auf Großbuchstaben <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> Spaltenname, die keine gültigen Elementnamen (NCNAME) darstellen, werden durch <code>_<number></code> (e. g. <code>_2</code>) ersetzt, wo <code><number></code> der Index der Spalte ist.</p> </div>

<i>connectionURL</i>	<p>Verbindungs-URL für die JDBC-Datenbank</p> <p> Wird die Verbindung über die Verbindungs-URL hergestellt, darf der Parameter <code>JndiName</code> nicht ausgefüllt werden.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> Die Art des Aufbaus hängt von der Datenbank ab, zu der eine Verbindung aufgebaut werden soll. Beispiel für <code>sqlite</code>: <code>jdbc:sqlite:C:/Temp/mydatabase.db</code>
<i>user</i>	Benutzername zur Authentifizierung bei Verwendung des Parameters <code>connectionURL</code>
<i>password</i>	Passwort zur Authentifizierung bei Verwendung des Parameters <code>connectionURL</code>
<i>CommitMode</i>	<p>Steuert, welche Variante für Commits verwendet werden soll</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> <i>NONE</i>: Keinen Commit durchführen <i>AUTO</i>: Automatischer Commit gemäß JDBC-API (d. h. nach jedem Batch)(Standard) <i>STATEMENT</i>: Explizites Commit nach jedem Statement (d. h. nach allen Batches für dieses Statement) <i>DOCUMENT</i>: Am Ende des Dokuments (d. h. nach Ausführung aller Statements) <i>MANAGED</i>: Adapter in Transaktionen nutzen (für extern verwaltete Transaktionen)
<i>AddMetaDataForQueries</i>	<p>Steuert, ob Datenbank-Metadaten, siehe JDBC Metadata Explorer, zu jedem <i>Result Set</i> im Ergebnis-XML-Dokument des Adapters ausgegeben werden sollen, wenn der Adapter eine SQL-Abfrage ausführt (nicht der Fall bei UPDATE, DELETE etc.). Im Batch-Modus (StatementBatch, PreparedStatementBatch und CallableStatementBatch) ist dies ebenfalls nicht möglich.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> <i>ja</i>: Dem Ergebnisdokument Datenbank-Metadaten hinzufügen <i>nein</i>: Ergebnisdokument ohne Datenbank-Metadaten ausgeben (Standard)

<i>fetchSize</i>	<p>Durch die Eingabe einer positiven ganzen Zahl lässt sich die Anzahl der Ergebnisse pro Abfrage beschränken. Sind Daten vorhanden, wird der Status 1 geliefert, ist das Ergebnis leer, wird der Status 0 ausgegeben.</p> <p>Wenn die eingegebene Zahl kleiner 1 ist, werden alle Daten ausgegeben.</p>
<i>id</i>	<p>Eindeutige ID des <i>JDBC Iterator</i>-Bausteins innerhalb des Prozess-Kontextes bzw. der Prozess-Instanz (wenn Sie mehrere voneinander unabhängig iterierende <i>JDBC Iteratoren</i> in demselben Prozess-Kontext verwenden möchten);</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • 0 : Aktuelle Action-ID (Nummer des Prozess-Schritts des <i>JDBC Iterators</i> im aktuellen Prozess) als ID verwenden; Beachten Sie hierbei, dass ein <i>JDBC Iterator</i> in einem Subprozess die gleiche Action-ID besitzen könnte (Standard) • Beliebige ganze Zahl außer 0, um den jeweiligen <i>JDBC Iterator</i>-Baustein innerhalb des Prozess-Kontextes eindeutig zu kennzeichnen; Empfehlung: einen Wert >1000 verwenden, um nicht zufällig eine bereits bestehende Action-ID zu verwenden.

Statuswerte

1	<p>Die Abfrage hat Datensätze geliefert bzw. die Anweisung hat Datensätze geändert</p> <p>Der Status 1 wird geliefert, wenn mehrere Abfragen bzw. Anweisungen im Dokument enthalten sind und mindestens eine Abfrage ein Ergebnis geliefert oder verändert hat. Der Adapter gibt diesen Status solange zurück, solange noch Datenblöcke zur Verarbeitung vorhanden sind.</p>
0	<p>Die Abfrage hat keine Ergebnisse geliefert oder die Anweisung hat keine Datensätze geändert</p> <p>Der Status 0 wird geliefert, wenn mehrere Abfragen bzw. Anweisungen im Dokument enthalten sind und keine Abfrage ein Ergebnis geliefert oder verändert hat.</p>
-1	<p>Technischer Fehler: genauere Informationen zur Fehlerursache entnehmen Sie dem Server-Log</p>

Input

Der *JDBC Iterator* erwartet spezifische Input-XML-Strukturen, die Datenbank-Statements enthalten. Diese Input-XML-Strukturen können über XSL-Mappings in Abhängigkeit von den verarbeiteten XML-Daten dynamisch generiert werden.

6.6.2.1 Statement: Direktes SQL

Input

```
<Beliebiger Elementname>
  <statement resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      SQL-Anweisungen
    </sql>
  </statement>
</Beliebiger Elementname>
```

Für jedes Element `<statement/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.
- Es enthält mindestens ein Element `<sql/>`, das jeweils eine SQL-Anweisung enthält.

i Falls Sie innerhalb der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden `CDATA`-Block, z. B. `<![CDATA[SELECT * FROM TABLE WHERE ID<1]]>`

Beispiel

```
<select>
  <statement resulttag="result" timezone="UTC" format="dd.MM.yyyy">
    <sql>
      <![CDATA[SELECT * FROM JMS_USERS]]>
    </sql>
  </statement>
</select>
```

Output

Wurde die SQL-Anweisung erfolgreich ausgeführt, gibt der Datenbankverbindungs-Baustein ein XML-Dokument mit folgender Struktur zurück, das die Antwort der Datenbank als Zeichenkette (String) enthält:

```
<Data>
  <Ergebniselementname>
    <Feld1><!--Wert--></Feld1>
    <Feldn><!--Wert--></Feldn>
  </Ergebniselementname>
</Data>
```

i Bei `SELECT`-Statements zum Zugriff auf Datenbanktabellen mit Spaltennamen, die keine gültigen XML-Namen darstellen, werden im Ergebnis-XML-Dokument synthetische Elementnamen erzeugt, um die Wohlgeformtheit des XML-Dokuments sicherzustellen.

6.6.2.2 StatementBatch: Direktes SQL im Batch-Modus

Input

Der *JDBC Iterator* führt mehrere SQL-Anweisungen **auf einmal im Batch-Modus aus**, wenn sie in folgender Input-XML-Struktur abgelegt werden. Dabei sind lediglich SQL-Anweisungen möglich, die kein *ResultSet* zurückliefern, also z. B. keine Abfragen mit *SELECT*:

```
<Beliebiger Elementname>
  <statementbatch resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      SQL-Anweisung 1
    </sql>
    <sql>
      SQL-Anweisung n
    </sql>
  </statementbatch>
</Beliebiger Elementname>
```

Für jedes Element `<statementbatch/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. *GMT+2*.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.
- Es enthält mindestens ein Element `<sql/>`, das jeweils eine SQL-Anweisung enthält.

i Falls Sie innerhalb der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden *CDATA*-Block, z. B. `<![CDATA[UPDATE PROCESS SET X4_VERSION=4 WHERE X4_DURATION=100]]>`

Beispiel

```
<select>
  <statementbatch resulttag="Ergebniselementname" timezone="UTC" format="dd.MM.yyyy">
    <sql>
      <![CDATA[UPDATE PROCESS SET X4_VERSION=4 WHERE X4_DURATION=100]]>
    </sql>
    <sql>
      <![CDATA[UPDATE PROCESS SET X4_VERSION=4 WHERE X4_DURATION=200]]>
    </sql>
  </statementbatch>
</select>
```

Output

Wurde die Batch-Anweisung erfolgreich ausgeführt, gibt der Datenbankverbindungs-Baustein folgende XML-Struktur zurück:


```
<Data>
  <Ergebniselementname>
    <!--Status der Batch-Operation-->
  </Ergebniselementname>
</Data>
```

Als Status der Batch-Operation sind folgende Ausgaben möglich:

- (Anzahl der Zeilen) rows affected. oder
- SUCCESS_NO_INFO oder
- EXECUTE_FAILED

6.6.2.3 PreparedStatement: SQL mit Werten

Der *JDBC Iterator* führt SQL-Anweisungen mit dynamischen Input-Daten aus, wenn sie in folgender Input-XML-Struktur abgelegt werden.

 Alternativ können Sie ab Release 5.0 der X4 Suite auch SQL-Anweisungen mit benannten Parametern ausführen, siehe [NamedPreparedStatement: SQL mit benannten Parametern](#).

Input

```
<Beliebiger Elementname>
  <preparedstatement resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      <!--SQL-Anweisungen-->
    </sql>
    <columns>
      <column mode="in" name="frei wählbarer NCName"
        index="Platzhalter-Index" type="Datentyp"
        timezone="Zeitzone" format="Datumsformat">
      </column>
    </columns>
    <rowset>
      <row>
        <Name><!--Wert--></Name>
      </row>
    </rowset>
  </preparedstatement>
</Beliebiger Elementname>
```

Für jedes Element `<preparedstatement/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.
- Es enthält mindestens ein Element `<sql/>`.

i Falls Sie innerhalb der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden *CDATA*-Block, z. B. `<![CDATA[SELECT * FROM TABLE WHERE ID<1]]>`

Für jedes Element `<column/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n$ -mal)
- Das optionale Attribut `mode` gibt die "Richtung" des Datenstroms an (*in*: Input-Parameter für die Datenbank; *out*: Ergebnisparameter aus der Datenbank; *inout*: sowohl Input-Parameter als auch Ergebnisparameter; Standard: *in*).
- Das Attribut `name` gibt einen frei wählbaren Namen (*NCName*) für ein Parameter an.
- Das Attribut `index` gibt einen ganzzahligen Platzhalterindex für den in Attribut `name` definierten Parameter an.
- Das optionale Attribut `type` gibt den Datentyp an (z. B. *INTEGER* oder *BLOB*; Standard: *VARCHAR*).
- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. *GMT+2*.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.

Für jedes Element `<rowset/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n$ -mal)
- Es enthält mindestens ein Element `<row/>`, das die Statement-Ausführung steuert.

Für jedes Element `<row/>` gilt:

- Es darf beliebig viele Elemente enthalten, die den Namen der Tabellenspalte oder den im SQL-Statement gesetzten Spalten-Alias tragen.
- Die darin liegenden Elemente enthalten die jeweiligen Werte.

i Beachten Sie Folgendes

- Die Attribute `mode` und `format` sind in *X4 Server* Revision 31469 Pflichtparameter; ab neueren Versionen sind diese optional.
- Zulässige Datentypen für das Attribut `type` zum Setzen (Input-Parameter) und Lesen (*ResultSet*) von Werten sind: *BIGINT*, *BINARY*, *BIT*, *BOOLEAN*, *BLOB*, *CHAR*, *CLOB*, *DATE*, *DECIMAL*, *DOUBLE*, *FLOAT*, *INTEGER*, *LONGNVARCHAR*, *LONGVARBINARY*, *LONGVARCHAR*, *NCHAR*, *NCLOB*, *NULL*, *NUMERIC*, *NVARCHAR*, *REAL*, *SMALLINT*, *SQLXML*, *TIME*, *TIMESTAMP*, *TINYINT*, *VARBINARY* und *VARCHAR* (Standard), *XML* (alias für *SQLXML*).

Beispiel


```

<Rootelement>
  <preparedstatement resulttag="abc" timezone="MEZ"
    format="dd.MM.yyyy">
    <sql>
    <![CDATA[SELECT * FROM JMS_USERS WHERE USERID=?]]>
    </sql>
    <columns>
      <column mode="in" name="id" index="1"
        type="INTEGER">
      </column>
    </columns>
    <rowset>
      <row>
        <id>23</id>
      </row>
    </rowset>
  </preparedstatement>
</Rootelement>

```

Output

Wurde die SQL-Anweisung erfolgreich ausgeführt, gibt der Adapter ein XML-Dokument mit folgender Struktur zurück, das die Antwort der Datenbank als Zeichenkette (String) enthält:


```

<Data>
  <Ergebniselementname>
    <Feld1><!--Wert--></Feld1>
    <Feldn><!--Wert--></Feldn>
  </Ergebniselementname>
</Data>

```

6.6.2.4 PreparedStatementBatch: SQL mit Werten im Batch-Modus

Der *JDBC Iterator* führt mehrere SQL-Anweisungen auf einmal im Batch-Modus aus, wenn sie in folgender Input-XML-Struktur abgelegt werden. Dabei sind lediglich SQL-Anweisungen möglich, die kein Result Set zurückliefern, also z.B. keine Abfragen mit SELECT.

-  Alternativ können Sie ab Release 5.0 der X4 Suite auch SQL-Anweisungen mit benannten Parametern mit oder ohne Batch-Modus ausführen, siehe [NamedPreparedStatementBatch: SQL mit benannten Parametern im Batch-Modus](#).

Input

```

<Beliebiger Elementname>
  <preparedstatementbatch resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      <!-- SQL-Anweisung -->
    </sql>
    <columns>
      <column mode="in" name="frei wählbarer NCName"
        index="Platzhalter-Index" type="Datentyp"
        timezone="Zeitzone" format="Datumsformat">
      </column>
    </columns>
    <rowset>
      <row>
        <Name><!--Wert--></Name>
      </row>
      <!-- Weitere <row>-Elemente für Batch-Modus -->
    </rowset>
  </preparedstatementbatch>
</Beliebiger Elementname>

```

Für jedes Element `<preparedstatementbatch/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.
- Es enthält mindestens ein Element `<sql/>`.

i Falls Sie innerhalb der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden `CDATA`-Block, z. B. `<![CDATA[UPDATE PROCESS SET X4_VERSION=4 WHERE X4_DURATION=?]]>`

Für jedes Element `<column/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n$ -mal)
- Das optionale Attribut `mode` gibt die "Richtung" des Datenstroms an (*in*: Input-Parameter für die Datenbank; *out*: Ergebnisparameter aus der Datenbank; *inout*: sowohl Input-Parameter als auch Ergebnisparameter; Standard: *in*).
- Das Attribut `name` gibt einen frei wählbaren Namen (*NCName*) für ein Parameter an.
- Das Attribut `index` gibt einen ganzzahligen Platzhalterindex für den in Attribut `name` definierten Parameter an.
- Das optionale Attribut `type` gibt den Datentyp an (z. B. *INTEGER* oder *BLOB*; Standard: *VARCHAR*).
- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.

Für jedes Element `<rowset/>` gilt:

- Es darf beliebig häufig verwendet werden ($0-n$ -mal)
- Es enthält beliebig viele Elemente `<row/>`, welche die Statement-Ausführung steuern.

Für jedes Element `<row/>` gilt:

- Es darf beliebig viele Elemente enthalten, die den Namen der Tabellenspalte oder den im SQL-Statement gesetzten Spalten-Alias tragen.
- Die darin liegenden Elemente enthalten die jeweiligen Werte.

Beachten Sie Folgendes

- Die Attribute `mode` und `format` sind in *X4 Server* Revision 31469 Pflichtparameter; ab neueren Versionen sind diese optional.
- Zulässige Datentypen für das Attribut `type` zum Setzen (Input-Parameter) und Lesen (*ResultSet*) von Werten sind: *BIGINT*, *BINARY*, *BIT*, *BOOLEAN*, *BLOB*, *CHAR*, *CLOB*, *DATE*, *DECIMAL*, *DOUBLE*, *FLOAT*, *INTEGER*, *LONGNVARCHAR*, *LONGVARBINARY*, *LONGVARCHAR*, *NCHAR*, *NCLOB*, *NULL*, *NUMERIC*, *NVARCHAR*, *REAL*, *SMALLINT*, *SQLXML*, *TIME*, *TIMESTAMP*, *TINYINT*, *VARBINARY* und *VARCHAR*(Standard), *XML* (alias für *SQLXML*).

Beispiel

```
<Rootelement>
  <preparedstatementbatch resulttag="Ergebniselementname" timezone="MEZ"
    format="dd.MM.yyyy">
    <sql>
      <![CDATA[ UPDATE PROCESS SET X4_VERSION=4 WHERE X4_DURATION=? ]]>
    </sql>
    <columns>
      <column mode="in" name="id" index="1"
        type="INTEGER">
      </column>
    </columns>
    <rowset>
      <row>
        <id>23</id>
      </row>
      <row>
        <id>25</id>
      </row>
    </rowset>
  </preparedstatementbatch>
</Rootelement>
```

Output

Wurde die Batch-Anweisung erfolgreich ausgeführt, gibt der Datenbankverbindungs-Baustein folgende XML-Struktur zurück:

```
<Data>
  <Ergebniselementname>
    <!--Status der Batch-Operation-->
  </Ergebniselementname>
</Data>
```

Als Status der Batch-Operation sind folgende Ausgaben möglich:

- *(Anzahl der Zeilen)* rows affected. oder
- SUCCESS_NO_INFO oder
- EXECUTE_FAILED

6.6.2.5 CallableStatement: Stored Procedures

Der *JDBC Iterator* führt Stored Procedures über *JDBC Callablestatements* mit dynamischen Input-Daten aus, wenn sie in folgender Input-XML-Struktur abgelegt werden.

Input

```
<Beliebiger Elementname>
  <callablestatement resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      <!--SQL-Anweisungen -->
    </sql>
    <columns>
      <column mode="Modus" name="frei wählbarer Name" index="Platzhalter"
        type="Datentyp" timezone="Zeitzone" format="Datumsformat">
      </column>
    </columns>
    <rowset>
      <row>
        <Name><!--Wert--></Name>
      </row>
    </rowset>
  </callablestatement>
</Beliebiger Elementname>
```

Für jedes Element `<callablestatement/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.
- Es enthält mindestens ein Element `<sql/>`.

i Falls Sie innerhalb der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden *CDATA*-Block, z. B. `<![CDATA[SELECT * FROM TABLE WHERE ID<1]]>`

Für jedes Element `<column/>` gilt:

- Es darf beliebig häufig verwendet werden ($0-n$ -mal)
- Das optionale Attribut `mode` gibt die "Richtung" des Datenstroms an (`in`: Input-Parameter für die Datenbank; `out`: Ergebnisparameter aus der Datenbank; `inout`: sowohl Input-Parameter als auch Ergebnisparameter; siehe <http://download.oracle.com/javase/1.5.0/docs/guide/jdbc/getstart/callablestatement.html>); Standard: `in`.
- Das Attribut `name` gibt einen frei wählbaren Namen für ein Parameter an.
- Das Attribut `index` gibt einen ganzzahligen Platzhalterindex für den in Attribut `name` definierten Parameter an.
- Das optionale Attribut `type` gibt den Datentyp an (zulässige Datentypen siehe Hinweis unten; Standard: `VARCHAR`).
- Das Attribut `name` gibt einen frei wählbaren Namen für ein Parameter an.
- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.

Für jedes Element `<rowset/>` gilt:

- Es darf beliebig häufig verwendet werden ($0-n$ -mal).
- Es enthält beliebig viele Elemente `<row/>`, die die Statement-Ausführung steuern.

Für jedes Element `<row/>` gilt:

- Es darf beliebig viele Elemente enthalten, die den Namen der Tabellenspalte oder den im SQL-Statement gesetzten Spalten-Alias tragen.
- Die darin liegenden Elemente enthalten die jeweiligen Werte.

Beachten Sie Folgendes

- Die Attribute `mode` und `format` sind in *X4 Server* Revision 31469 Pflichtparameter; ab neueren Versionen sind diese optional.
- Zulässige Datentypen für das Attribut `type` zum Setzen (Input-Parameter) und Lesen (*ResultSet*) von Werten sind: `BIGINT`, `BINARY`, `BIT`, `BOOLEAN`, `BLOB`, `CHAR`, `CLOB`, `DATE`, `DECIMAL`, `DOUBLE`, `FLOAT`, `INTEGER`, `LONGNVARCHAR`, `LONGVARBINARY`, `LONGVARCHAR`, `NCHAR`, `NCLOB`, `NULL`, `NUMERIC`, `NVARCHAR`, `REAL`, `SMALLINT`, `SQLXML`, `TIME`, `TIMESTAMP`, `TINYINT`, `VARBINARY` und `VARCHAR` (Standard), `XML` (alias für `SQLXML`).

Beispiel

```

<Rootelement>
  <callablestatement resulttag="abc" timezone="MEZ"
    format="dd.MM.yyyy">
    <sql>
    <![CDATA[ {call getData(?)} ]]>
    </sql>
    <columns>
      <column mode="out" name="id" index="1"
        type="INTEGER">
      </column>
    </columns>
    <rowset>
      <row>
        <id>TestUser</id>
      </row>
    </rowset>
  </callablestatement>
</Rootelement>

```

Output

Wurde die SQL-Anweisung erfolgreich ausgeführt, gibt der Datenbankverbindungs-Baustein ein XML-Dokument mit folgender Struktur zurück, das die Antwort der Datenbank als Zeichenkette (String) enthält:

```

<Data>
  <Ergebniselementname>
    <Feld1><!--Wert--></Feld1>
    <Feldn><!--Wert--></Feldn>
  </Ergebniselementname>
</Data>

```

6.6.2.6 CallableStatementBatch: Stored Procedures im Batch-Modus

Der *JDBC Iterator* führt Stored Procedures über *JDBC Callablestatements* mit dynamischen Input-Daten aus, wenn sie in folgender Input-XML-Struktur abgelegt werden. Dabei sind lediglich SQL-Anweisungen möglich, die kein ResultSet zurückliefern, also z. B. keine Abfragen mit SELECT.

Input

```

<Beliebiger Elementname>
  <callablestatementbatch resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      <!--SQL-Anweisungen -->
    </sql>
    <columns>
      <column mode="Modus" name="frei wählbarer Name" index="Platzhalter"
        type="Datentyp" timezone="Zeitzone" format="Datumsformat">
      </column>
    </columns>
    <rowset>
      <row>
        <Name><!--Wert--></Name>
      </row>
    </rowset>
  </callablestatementbatch>
</Beliebiger Elementname>

```

Für jedes Element `<callablestatementbatch/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.
- Es enthält mindestens ein Element `<sql/>`.

i Falls Sie innerhalb der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden `CDATA`-Block, z. B. `<![CDATA[SELECT * FROM TABLE WHERE ID<1]]`

Für jedes Element `<column/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n$ -mal)
- Das optionale Attribut `mode` gibt die "Richtung" des Datenstroms an (*in*: Input-Parameter für die Datenbank; *out*: Ergebnisparameter aus der Datenbank; *inout*: sowohl Input-Parameter als auch Ergebnisparameter; siehe <http://download.oracle.com/javase/1.5.0/docs/guide/jdbc/getstart/callablestatement.html>); Standard: *in*.
- Das Attribut `name` gibt einen frei wählbaren Namen für ein Parameter an.
- Das Attribut `index` gibt einen ganzzahligen Platzhalterindex für den in Attribut `name` definierten Parameter an.
- Das optionale Attribut `type` gibt den Datentyp an (zulässige Datentypen siehe Hinweis unten; Standard: `VARCHAR`).
- Das Attribut `name` gibt einen frei wählbaren Namen für ein Parameter an.
- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an.

Für jedes Element `<rowset/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n$ -mal).
- Es enthält beliebig viele Elemente `<row/>`, die die Statement-Ausführung steuern.

Für jedes Element `<row/>` gilt:

- Es darf beliebig viele Elemente enthalten, die den Namen der Tabellenspalte oder den im SQL-Statement gesetzten Spalten-Alias tragen.
- Die darin liegenden Elemente enthalten die jeweiligen Werte.

ⓘ Beachten Sie Folgendes

- Die Attribute `mode` und `format` sind in *X4 Server* Revision 31469 Pflichtparameter; ab neueren Versionen sind diese optional.
- Zulässige Datentypen für das Attribut `type` zum Setzen (Input-Parameter) und Lesen (*ResultSet*) von Werten sind: *BIGINT*, *BINARY*, *BIT*, *BOOLEAN*, *BLOB*, *CHAR*, *CLOB*, *DATE*, *DECIMAL*, *DOUBLE*, *FLOAT*, *INTEGER*, *LONGNVARCHAR*, *LONGVARBINARY*, *LONGVARCHAR*, *NCHAR*, *NCLOB*, *NULL*, *NUMERIC*, *NVARCHAR*, *REAL*, *SMALLINT*, *SQLXML*, *TIME*, *TIMESTAMP*, *TINYINT*, *VARBINARY* und *VARCHAR*(Standard), *XML* (alias für *SQLXML*).

Beispiel


```

<Rootelement>
  <callablestatementbatch resulttag="abc" timezone="MEZ"
    format="dd.MM.yyyy">
    <sql>
    <![CDATA[ {call getData(?) } ]]>
    </sql>
    <columns>
      <column mode="out" name="id" index="1"
        type="INTEGER">
      </column>
    </columns>
    <rowset>
      <row>
        <id>TestUser</id>
      </row>
    </rowset>
  </callablestatementbatch>
  <callablestatementbatch resulttag="abc" timezone="MEZ"
    format="dd.MM.yyyy">
    <sql>
    <![CDATA[ {call getData(?) } ]]>
    </sql>
    <columns>
      <column mode="out" name="id" index="1"
        type="INTEGER">
      </column>
    </columns>
    <rowset>
      <row>
        <id>TestUser</id>
      </row>
    </rowset>
  </callablestatementbatch>
</Rootelement>

```

Output

Wurde die Batch-Anweisung erfolgreich ausgeführt, gibt der Datenbankverbindungs-Baustein folgende XML-Struktur zurück:

```

<Data>
  <Ergebniselementname>
    <!--Status der Batch-Operation-->
  </Ergebniselementname>
</Data>

```

Als Status der Batch-Operation sind folgende Ausgaben möglich:

- (Anzahl der Zeilen) rows affected. oder
- SUCCESS_NO_INFO oder
- EXECUTE_FAILED

6.6.2.7 NamedPreparedStatement: SQL mit benannten Parametern

Der *JDBC Iterator* führt SQL-Anweisungen mit benannten Parametern aus, wenn sie in folgender Input-XML-Struktur abgelegt werden.

Input

```
<Beliebiger Elementname>
  <namedpreparedstatement resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      <!--SQL-Anweisungen (optional innerhalb eines CDATA-Blocks) -->
      <!--Je Parameter ein XML-Element, gleichnamig zum entsprechenden Kindelement
in <row> -->
    </sql>
    <rowset>
      <row>
        <Name><!--Wert--></Name>
      </row>
    </rowset>
  </namedpreparedstatement>
</Beliebiger Elementname>
```

Für jedes Element `<namedpreparedstatement/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an und bestimmt das Format der zurückgegebenen Datumswerten.
- Es enthält ein Element `<sql/>`.

Für jedes XML-Element eines benannten Parameters innerhalb der SQL-Anweisung in Element `<sql/>` gilt:

- Der Elementname gibt den Parameter-Namen an, der mit entsprechenden Element innerhalb von `<row/>` übereinstimmt.
- Das optionale Attribut `mode` gibt die "Richtung" des Datenstroms an (*in*: Input-Parameter für die Datenbank; *out*: Ergebnisparameter aus der Datenbank; *inout*: sowohl Input-Parameter als auch Ergebnisparameter; Standard: *in*).
- Das optionale Attribut `type` gibt den Datentyp an, z. B. `INTEGER`, `BLOB` oder `VARCHAR` (Standard).
- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an und gibt das erwartete Format für den Wert innerhalb des entsprechenden `<row>`-Elements vor.

- ❗ Falls Sie in der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden *CDATA*-Block, z. B. `<![CDATA[SELECT * FROM TABLE WHERE ID<]]><id type="INTEGER" />.`

Für jedes Element `<rowset/>` gilt:

- Es darf beliebig häufig verwendet werden (*0 – n-mal*)
- Es enthält beliebig viele Elemente `<row/>`, die die Statement-Ausführung steuern.

Für jedes Element `<row/>` gilt:

- Es muss mindestens ein Element enthalten, das dem Elementnamen im SQL-Statement entspricht.
- Die darin liegenden Elemente enthalten die jeweiligen Werte.

Beachten Sie für jedes Datenelement innerhalb von `<row/>`, dass im Elementinhalt keine unbeabsichtigten Zeilenumbrüche definiert wurden (z.B. durch Editor-Formatierung). Andernfalls würden diese Zeilenumbrüche in der erzeugten SQL-Anweisung als `\n` angefügt.

❗ **Beachten Sie Folgendes**

Zulässige Datentypen für das Attribut `type` zum Setzen (Input-Parameter) und Lesen (*ResultSet*) von Werten sind: *BIGINT*, *BINARY*, *BIT*, *BOOLEAN*, *BLOB*, *CHAR*, *CLOB*, *DATE*, *DECIMAL*, *DOUBLE*, *FLOAT*, *INTEGER*, *LONGNVARCHAR*, *LONGVARBINARY*, *LONGVARCHAR*, *NCHAR*, *NCLOB*, *NULL*, *NUMERIC*, *NVARCHAR*, *REAL*, *SMALLINT*, *SQLXML*, *TIME*, *TIMESTAMP*, *TINYINT*, *VARBINARY* und *VARCHAR* (Standard), *XML* (alias für *SQLXML*).

Beispiel

```
<RootElement>
  <namedpreparedstatement resulttag="namedPreparedStatementWithCDATA"
    timezone="UTC" format="dd.MM.yyyy">
    <sql>
      SELECT * FROM PROCESS WHERE X4_DURATION= <duration type="INTEGER"
        mode="in" /> AND X4_PROCESSNAME= <processname
          type="VARCHAR" mode="in" />
    </sql>
    <rowset>
      <row>
        <duration>101</duration>
        <processname>Project/Process/Path/Process.wrf</processname>
      </row>
    </rowset>
  </namedpreparedstatement>
</RootElement>
```

Output

Wurde die SQL-Anweisung erfolgreich ausgeführt, gibt der Adapter ein XML-Dokument mit folgender Struktur zurück, das die Antwort der Datenbank als Zeichenkette (String) enthält:

```
<Data>
  <Ergebniselementname>
    <Feld1><!--Wert--></Feld1>
    <Feldn><!--Wert--></Feldn>
  </Ergebniselementname>
</Data>
```

6.6.2.8 NamedPreparedStatementBatch: SQL mit benannten Parametern im Batch-Modus

Der *JDBC Iterator* führt mehrere SQL-Anweisungen auf einmal im Batch-Modus aus, wenn sie in folgender Input-XML-Struktur jeweils in einem `<namedpreparedstatementbatch>`-Block abgelegt werden. Dabei sind lediglich SQL-Anweisungen möglich, die kein `ResultSet` zurückliefern, also z. B. keine Abfragen mit `SELECT`.

Input

```
<Beliebiger Elementname>
  <namedpreparedstatementbatch resulttag="Ergebniselementname" timezone="Zeitzone"
    format="Datumsformat">
    <sql>
      <!--SQL-Anweisungen (optional innerhalb eines CDATA-Blocks) -->
      <!--Je Parameter ein XML-Element, gleichnamig zum entsprechenden Kindelement
in <row> -->
    </sql>
    <rowset>
      <row>
        <Name><!--Wert--></Name>
      </row>
      <!-- Weitere <row>-Elemente für Batch-Modus -->
    </rowset>
  </namedpreparedstatementbatch>
</Beliebiger Elementname>
```

Für jedes Element `<namedpreparedstatementbatch/>` gilt:

- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. `GMT+2`.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an und bestimmt das Format der zurückgegebenen Datumswerten.
- Es enthält mindestens ein Element `<sql/>`.

Für jedes XML-Element eines benannten Parameters innerhalb der SQL-Anweisung im Element `<sql/>` gilt:

- Der Elementname gibt den Parameter-Namen an, der mit entsprechenden Element innerhalb von `<row/>` übereinstimmt.
- Das optionale Attribut `mode` gibt die "Richtung" des Datenstroms an (*in*: Input-Parameter für die Datenbank; *out*: Ergebnisparameter aus der Datenbank; *inout*: sowohl Input-Parameter als auch Ergebnisparameter; Standard: *in*).

- Das optionale Attribut `type` gibt den Datentyp an, z. B. *INTEGER*, *BLOB* oder *VARCHAR*(Standard).
- Das optionale Attribut `timezone` gibt eine Zeitzone gemäß `java.util.TimeZone`, siehe <http://docs.oracle.com/javase/1.5.0/docs/api/java/util/TimeZone.html>, an, z. B. *GMT+2*.
- Das optionale Attribut `format` gibt ein Datumsformat gemäß `java.text.SimpleDateFormat` an und gibt das erwartete Format für den Wert innerhalb des entsprechenden `<row>`-Elements vor.

i Falls Sie in der SQL-Anweisung Zeichen verwenden, die in XML eine Bedeutung besitzen, maskieren Sie die SQL-Anweisung mit einem umgebenden *CDATA*-Block, z. B. `<![CDATA[SELECT * FROM TABLE WHERE ID<]]><id type="INTEGER" />`.

Für jedes Element `<rowset/>` gilt:

- Es darf beliebig häufig verwendet werden ($0 - n$ -mal)
- Es enthält beliebig viele Elemente `<row/>`, die die Statement-Ausführung steuern.

Für jedes Element `<row/>` gilt:

- Es muss mindestens ein Element enthalten, das dem Elementnamen im SQL-Statement entspricht.
- Die darin liegenden Elemente enthalten die jeweiligen Werte.

Beachten Sie für jedes Datenelement innerhalb von `<row/>`, dass im Elementinhalt keine unbeabsichtigten Zeilenumbrüche definiert wurden (z.B. durch Editor-Formatierung). Andernfalls würden diese Zeilenumbrüche in der erzeugten SQL-Anweisung als `\n` angefügt.

i Beachten Sie Folgendes

Zulässige Datentypen für das Attribut `type` zum Setzen (Input-Parameter) und Lesen (*ResultSet*) von Werten sind: *BIGINT*, *BINARY*, *BIT*, *BOOLEAN*, *BLOB*, *CHAR*, *CLOB*, *DATE*, *DECIMAL*, *DOUBLE*, *FLOAT*, *INTEGER*, *LONGNVARCHAR*, *LONGVARBINARY*, *LONGVARCHAR*, *NCHAR*, *NCLOB*, *NULL*, *NUMERIC*, *NVARCHAR*, *REAL*, *SMALLINT*, *SQLXML*, *TIME*, *TIMESTAMP*, *TINYINT*, *VARBINARY* und *VARCHAR*(Standard), *XML* (alias für *SQLXML*).

Beispiel

```

<Rootelement>
  <namedpreparedstatementbatch resulttag="namedPreparedStatementBatchWithCData"
    timezone="UTC" format="dd.MM.yyyy">
    <sql>
      <![CDATA[UPDATE PROCESS SET X4_VERSION=4 WHERE X4_DURATION=]]>
      <duration type="INTEGER" mode="inout" />
    </sql>
    <rowset>
      <row>
        <duration>101</duration>
      </row>
      <row>
        <duration>9</duration>
      </row>
    </rowset>
  </namedpreparedstatementbatch>
</Rootelement>

```

Output

Wurde die SQL-Anweisung erfolgreich ausgeführt, gibt der Adapter ein XML-Dokument mit folgender Struktur zurück:

```

<Data>
  <Ergebniselementname count="Anzahl der betroffenen Datenbankeinträge">
    <!--Status der Batch-Operation-->
  </Ergebniselementname>
</Data>

```


Als Status der Batch-Operation sind folgende Ausgaben möglich:

- (Anzahl der Zeilen) rows affected. oder
- SUCCESS_NO_INFO oder
- EXECUTE_FAILED

6.6.2.9 Abrufen von Datenbank-Metadaten

Mit dem Adapter-Parameter `AddMetaDataForQueries` (ab *X4 Server Revision 32408*) können Sie steuern, ob Datenbank-Metadaten zu den entsprechenden *Result Sets* im Ergebnis-XML-Dokument innerhalb eines XML-Elements ermittelt und gemeinsam mit den Ergebnissen der SQL-Abfrage ausgegeben werden sollen.

Der Name jedes XML-Elements, das die entsprechenden Metadaten enthält, setzt sich aus dem Wert des Attributs `resulttag` im Input-Dokument und dem Zusatz `-meta` zusammen, z. B. `<result-meta>`.

-  Bei SQL-Statements, die kein *Result Set* ausgeben (z. B. bei UPDATE oder DELETE), können keine Metadaten ausgegeben werden. Im Batch-Modus des Adapters (StatementBatch, PreparedStatementBatch und CallableStatementBatch) ist dies ebenfalls nicht möglich.

Beispiel

Im folgenden Input-XML-Dokument wird eine SQL-Abfrage und als Ergebniselement der Name `result` definiert:

```
<Root>
  <statement resulttag="result">
    <sql>
      SELECT X4USER, STARTTIME FROM HISTORY WHERE PID='1279017294476'
    </sql>
  </statement>
</Root>
```

Der *JDBC Iterator* gibt daraufhin z. B. folgendes Ergebnis-XML-Dokument aus, welches das Ergebnis der SQL-Abfrage sowie die Metadaten der *Result Sets* enthält:

```
<Data>
  <result-meta columns="2">
    <Column name="X4USER" label="X4USER" type="VARCHAR"
      nullable="true" auto-increment="false" read-only="false"
      writable="false" display-size="255" case-sensitive="true"
      searchable="true" signed="false" />
    <Column name="STARTTIME" label="STARTTIME" type="TIMESTAMP"
      nullable="true" auto-increment="false" read-only="false"
      writable="false" display-size="26" case-sensitive="false"
      searchable="true" signed="false" />
  </result-meta>
  <result>
    <X4USER>1</X4USER>
    <STARTTIME>2010-07-13 12:34:54.553</STARTTIME>
  </result>
</Data>
```

6.6.3 JDBC Metadata Explorer






JDBC Metadata Explorer: Stellt eine Verbindung zu einer JDBC-kompatiblen Datenbank her und ermittelt Metadaten zur Datenbank, zu Tabellen und Spalten sowie zu hinterlegten Primär- und Fremdschlüsseln. Der Adapter erwartet ein Input-XML-Dokument mit adapterspezifischen Strukturen, um die gewünschten Metadaten abzufragen.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>execute</i> : Adapter ausführen
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.jdbc.meta.MetadataExplorer</code>: Hauptklasse (Standard)</p>
<i>JndiName</i>	<p>JNDI-Name der DataSource</p> <p> Wird die Verbindung über den JNDI-Namen hergestellt, darf der Parameter <code>connectionURL</code> nicht ausgefüllt werden.</p> <p>Mögliche Werte: Gültiger JNDI-Name (z.B. <code>java:/DefaultDS</code>)</p>
<i>ColumnCase</i>	<p>Steuert wie die Spaltennamen ins Ergebnis übernommen werden. Da Datenbanken unterschiedliches Verhalten betreffend der Groß-/ Kleinschreibung von Spaltennamen haben, kann man hiermit die Weiterverarbeitung des Ergebnis unabhängig vom Datenbankverhalten machen.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> <code>NO_CHANGE</code>: Ändert die Spaltennamen nicht, d.h. die Schreibweise im Ergebnis ist exakt wie von der Datenbank geliefert (Standard) <code>TO_LOWER</code>: Ändert die Spaltennamen auf Kleinbuchstaben <code>TO_UPPER</code>: Ändert die Spaltennamen auf Großbuchstaben <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> Spaltenname, die keine gültigen Elementnamen (NCNAME) darstellen, werden durch <code>_<number></code> (z.B. <code>_2</code>) ersetzt, wobei <code><number></code> der Index der Spalte ist.</p> </div>
<i>connectionURL</i>	<p>Verbindungs-URL für die JDBC-Datenbank</p> <p> Wird die Verbindung über die Verbindungs-URL hergestellt, darf der Parameter <code>JndiName</code> nicht ausgefüllt werden.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> Die Art des Aufbaus hängt von der Datenbank ab, zu der eine Verbindung aufgebaut werden soll. Beispiel für sqlite: <code>jdbc:sqlite:C:/Temp/mydatabase.db</code>
<i>user</i>	Benutzername zur Authentifizierung bei Verwendung des Parameters <code>connectionURL</code>
<i>password</i>	Passwort zur Authentifizierung bei Verwendung des Parameters <code>connectionURL</code>

Input

Der *JDBC Metadata Explorer* erwartet spezifische Input-XML-Strukturen, die je nach Bedarf eine oder mehrere der im Folgenden beschriebenen Metadaten-Abfragen enthalten können. Diese Input-XML-

Strukturen können über XSL-Mappings in Abhängigkeit von den verarbeiteten XML-Daten dynamisch generiert werden.

6.6.3.1 Datenbank-Metadaten abrufen

Mit dem *JDBC Metadata Explorer* können Sie verschiedene Metadaten (Hersteller-/Produktname, unterstützte Funktionen etc.) und die vorhandenen Schemas von der via JDBC verbundenen Datenbank abrufen.

Input

Einem Input-XML-Dokument fügen Sie dazu innerhalb eines beliebigen Wurzel-Elements das Element `<datasemeta resulttag="Ergebniselementname"/>` hinzu, in dessen Attribut `resulttag` Sie einen gültigen XML-Namen angeben.

```
<Root>
  <datasemeta resulttag="Ergebniselementname"/>
</Root>
```

Ausgabe

Die XML-Struktur, die vom Adapter ausgegeben wird, besitzt innerhalb des Wurzel-Elements `<Data>` ein Element, dessen Name in der Input-Struktur im Attribut `resulttag` angegeben wurde und das die Schema-Informationen und Datenbank-Metadaten enthält:

```
<Data>
  <Ergebniselementname eigenschaft="wert" ...>
    <Schema>
      <TABLE_SCHEM>DB-Schema</TABLE_SCHEM>
      <TABLE_CATALOG isNull="(true/false)" />
    </Schema>
    ...
  </Ergebniselementname>
</Data>
```

Beispiel

In einem Input-XML-Dokument wird folgende XML-Struktur definiert, um Metadaten zur verbundenen Datenbank auszugeben:

```
<RootElement>
  <datasemeta resulttag="DB"/>
</RootElement>
```

Der *JDBC Metadata Explorer* gibt daraufhin z. B. folgendes Ergebnis-XML-Dokument aus:

```
<Data>
  <DB product-name="Apache Derby" readOnly="false"
    supports-stored-procedures="true" supports-batch-updates="true"
    all-tables-are-selectable="true" supports-outer-joins="true"
    supports-convert="true" supports-group-by="true">
    <Schema>
      <TABLE_SCHEM>APP</TABLE_SCHEM>
      <TABLE_CATALOG isNull="true" />
    </Schema>
    ...
  </DB>
```

6.6.3.2 Tabellen-Metadaten abrufen

Mit dem *JDBC Metadata Explorer* können Sie Tabellen-Metadaten (Name, Typ etc.) zu einer oder mehreren Datenbank-Tabellen abrufen; hierbei können Sie die Suche nach Katalog, Schema und Tabellename eingrenzen.

Input


Einem Input-XML-Dokument fügen Sie dazu innerhalb eines beliebigen Wurzel-Elements das Element `<tablemetadata resulttag="Ergebniselementname" />` hinzu, in dessen Attribut `resulttag` Sie einen gültigen XML-Namen angeben. Die Angabe von Katalog, Schema und Tabelle in den Unterelementen `<catalog>`, `<schema>` und `<table>` sind hierbei optional; wenn diese Such-Einschränkungen fehlen, werden sämtliche verfügbaren Metadaten abgerufen.

```
<Root>
  <tablemetadata resulttag="Ergebniselementname">
    <catalog>Katalog</catalog>
    <schema>DB-Schema</schema>
    <table>DB-Tabelle</table>
  </tablemetadata>
</Root>
```

Ausgabe

Die XML-Struktur, die vom Adapter ausgegeben wird, besitzt innerhalb des Wurzel-Elements `<Data>` ein XML-Element, dessen Elementname in der Input-Struktur im Attribut `resulttag` angegeben wurde und das die Tabellen-Metadaten enthält. Wenn es zu einem Abruf mehrere Ergebnisse gibt (d. h. mehrere Tabellen oder Sichten), so werden diese jeweils in einem separaten Element ausgegeben.

```
<Data>
  <Ergebniselementname>
    <Tabellen-Eigenschaft>Wert</Tabellen-Eigenschaft>
    ...
  </Ergebniselementname>
</Data>
```

 Erläuterungen (englischsprachig) zu den einzelnen Tabellen-Eigenschaften finden Sie [hier](#).

Beispiel

In einem Input-XML-Dokument wird folgende XML-Struktur definiert, um Metadaten zu einer Tabelle HISTORY im Datenbankschema APP auszugeben:

```
<Root>
  <tablemetadata resulttag="DBTable">
    <catalog></catalog>
    <schema>APP</schema>
    <table>HISTORY</table>
  </tablemetadata>
</Root>
```

Der *JDBC Metadata Explorer* gibt daraufhin z. B. folgendes Ergebnis-XML-Dokument aus:

```
<Data>
  <DBTable>
    <TABLE_CAT />
    <TABLE_SCHEM>APP</TABLE_SCHEM>
    <TABLE_NAME>HISTORY</TABLE_NAME>
    <TABLE_TYPE>TABLE</TABLE_TYPE>
    <REMARKS />
    <TYPE_CAT isNull="true" />
    <TYPE_SCHEM isNull="true" />
    <TYPE_NAME isNull="true" />
    <SELF_REFERENCING_COL_NAME isNull="true" />
    <REF_GENERATION isNull="true" />
  </DBTable>
</Data>
```

6.6.3.3 Spalten-Metadaten abrufen

Mit dem *JDBC Metadata Explorer* können Sie Spalten-Metadaten (Datentyp, Nullable, Autoincrement etc.) zu einer oder mehreren Tabellenspalten abrufen; hierbei können Sie die Suche nach Katalog, Schema, Tabelle und Spalte eingrenzen.

Input


Einem Input-XML-Dokument fügen Sie dazu innerhalb eines beliebigen Wurzel-Elements das Element `<tablecolumns resulttag="Ergebniselementname"/>` hinzu, in dessen Attribut `resulttag` Sie einen gültigen XML-Namen angeben. Die Angabe von Katalog, Schema, Tabelle und Spalte in den Unterelementen `<catalog>`, `<schema>`, `<table>` und `<column>` sind hierbei optional; wenn diese Such-Einschränkungen fehlen, werden sämtliche verfügbaren Metadaten abgerufen.

```
<Root>
  <tablemetadata resulttag="Ergebniselementname">
    <catalog>Katalog</catalog>
    <schema>DB-Schema</schema>
    <table>DB-Tabelle</table>
    <column>Spalte</column>
  </tablemetadata>
</Root>
```

Ausgabe

Die XML-Struktur, die vom Adapter ausgegeben wird, besitzt innerhalb des Wurzel-Elements <Data> ein Element, dessen Name in der Input-Struktur im Attribut resulttag angegeben wurde und das die Spalten-Metadaten enthält. Wenn es mehrere Ergebnisse gibt (d. h. mehrere Spalten), so werden diese jeweils in einem separaten Element ausgegeben.

```
<Data>
  <Ergebniselementname>
    <Spalten-Eigenschaft>Wert</Spalten-Eigenschaft>
    ...
  </Ergebniselementname>
</Data>
```

 Erläuterungen (englischsprachig) zu den einzelnen Spalten-Eigenschaften finden Sie [hier](#).

Beispiel 1

In einem Input-XML-Dokument wird folgende XML-Struktur definiert, um Metadaten zu einer Spalte X4_PID in der Tabelle HISTORY im Datenbankschema APP auszugeben:

```
<Root>
  <tablecolumns resulttag="DBColumn">
    <catalog></catalog>
    <schema>APP</schema>
    <table>HISTORY</table>
    <column>X4_PID</column>
  </tablecolumns>
</Root>
```

Der *JDBC Metadata Explorer* gibt daraufhin z. B. folgendes Ergebnis-XML-Dokument aus:

```

<Data>
  <DBColumn>
    <TABLE_CAT />
    <TABLE_SCHEM>APP</TABLE_SCHEM>
    <TABLE_NAME>HISTORY</TABLE_NAME>
    <COLUMN_NAME>X4_PID</COLUMN_NAME>
    <DATA_TYPE>12</DATA_TYPE>
    <TYPE_NAME>VARCHAR</TYPE_NAME>
    <COLUMN_SIZE>20</COLUMN_SIZE>
    <BUFFER_LENGTH isNull="true" />
    <DECIMAL_DIGITS isNull="true" />
    <NUM_PREC_RADIX isNull="true" />
    <NULLABLE>0</NULLABLE>
    <REMARKS/>
    <COLUMN_DEF isNull="true" />
    <SQL_DATA_TYPE isNull="true" />
    <SQL_DATETIME_SUB isNull="true" />
    <CHAR_OCTET_LENGTH>40</CHAR_OCTET_LENGTH>
    <ORDINAL_POSITION>1</ORDINAL_POSITION>
    <IS_NULLABLE>NO</IS_NULLABLE>
    <SCOPE_CATALOG isNull="true" />
    <SCOPE_SCHEMA isNull="true" />
    <SCOPE_TABLE isNull="true" />
    <SOURCE_DATA_TYPE isNull="true" />
    <IS_AUTOINCREMENT>NO</IS_AUTOINCREMENT>
  </DBColumn>
</Data>

```

Beispiel 2

In einem Input-XML-Dokument wird folgende XML-Struktur definiert, um Metadaten zu einer Spalte X4_PID in der Tabelle HISTORY im Datenbankschema APP auszugeben. Das Element `<column>` mit dem Attribut `label` innerhalb von `<resultsetRestriction>` schränkt dabei die ausgegebenen Daten ein:

```

<Root>
  <tablecolumns resulttag="DBColumn">
    <catalog></catalog>
    <schema>APP</schema>
    <table>HISTORY</table>
    <column>X4_PID</column>
    <resultsetRestriction>
      <column label="TABLE_NAME" />
      <column label="DATA_TYPE" />
      <column label="NULLABLE" />
      <column label="TYPE_NAME" />
    </resultsetRestriction>
  </tablecolumns>
</Root>

```

Der *JDBC Metadata Explorer* gibt daraufhin z. B. folgendes Ergebnis-XML-Dokument aus:

```
<Data>
  <DBColumn>
    <TABLE_NAME>HISTORY</TABLE_NAME>
    <COLUMN_NAME>X4_PID</COLUMN_NAME>
    <DATA_TYPE>12</DATA_TYPE>
    <TYPE_NAME>VARCHAR</TYPE_NAME>
  </DBColumn>
</Data>
```

6.6.3.4 Primärschlüssel einer Tabelle abrufen

Mit dem *JDBC Metadata Explorer* können Sie die Primärschlüssel (Name und Spalteninformationen) einer bestimmten Datenbank-Tabelle abrufen.

Input


Einem Input-XML-Dokument fügen Sie dazu innerhalb eines beliebigen Wurzel-Elements das Element `<primarykeys resulttag="Ergebniselementname"/>` hinzu, in dessen Attribut `resulttag` Sie einen gültigen XML-Namen angeben. Die Angabe einer Tabelle in Element `<table>` ist dabei erforderlich; die Elemente `<catalog>` und `<schema>` sind jedoch optional.

```
<Root>
  <primarykeys resulttag="Ergebniselementname">
    <catalog>Katalog</catalog>
    <schema>DB-Schema</schema>
    <table>DB-Tabelle (Pflichtfeld, darf nicht leer sein)</table>
  </primarykeys>
</Root>
```

Ausgabe

Die XML-Struktur, die vom Adapter ausgegeben wird, besitzt innerhalb des Wurzel-Elements `<Data>` ein XML-Element, dessen Elementname in der Input-Struktur im Attribut `resulttag` angegeben wurde und das die Metadaten zu jedem Primärschlüssel in der Tabelle enthält. Wenn es zu einem Abruf mehrere Ergebnisse gibt (mehrere Primärschlüsselspalten), so werden diese jeweils in einem separaten Element ausgegeben.

```
<Data>
  <Ergebniselementname>
    <Primärschlüssel-Eigenschaft>Wert</Primärschlüssel-Eigenschaft>
    ...
  </Ergebniselementname>
</Data>
```

 Erläuterungen (englischsprachig) zu den einzelnen Spalten-Eigenschaften finden Sie [hier](#).

Beispiel

In einem Input-XML-Dokument wird folgende XML-Struktur definiert, um die Primärschlüssel der Tabelle HISTORY auszugeben:

```
<Root>
  <primarykeys resulttag="historykeys">
    <schema></schema>
    <table>HISTORY</table>
  </primarykeys>
</Root>
```

Der *JDBC Metadata Explorer* gibt daraufhin z. B. folgendes Ergebnis-XML-Dokument aus:

```
<Data>
  <historykeys>
    <TABLE_CAT />
    <TABLE_SCHEM>APP</TABLE_SCHEM>
    <TABLE_NAME>HISTORY</TABLE_NAME>
    <COLUMN_NAME>PID</COLUMN_NAME>
    <KEY_SEQ>1</KEY_SEQ>
    <PK_NAME>SQL090505161403130</PK_NAME>
  </historykeys>
</Data>
```

6.6.3.5 Fremdschlüssel einer Tabelle abrufen

Mit dem *JDBC Metadata Explorer* können Sie die Fremdschlüssel (Name und Spalteninformationen) einer bestimmten Tabelle abrufen.

Input

Einem Input-XML-Dokument fügen Sie dazu innerhalb eines beliebigen Wurzel-Elements das Element `<foreignkeys resulttag="Ergebniselementname" />` hinzu, in dessen Attribut `resulttag` Sie einen gültigen XML-Namen angeben. Die Angabe einer Tabelle in Element `<table>` ist dabei erforderlich; die Elemente `<catalog>` und `<schema>` sind jedoch optional.


```
<Root>
  <foreignkeys resulttag="Ergebniselementname">
    <catalog>Katalog</catalog>
    <schema>DB-Schema</schema>
    <table>DB-Tabelle (Pflichtfeld, darf nicht leer sein)</table>
  </foreignkeys>
</Root>
```

Ausgabe

Die XML-Struktur, die vom Adapter ausgegeben wird, enthält innerhalb des Wurzel-Elements `<Data>` ein Element, dessen Name in der Input-Struktur im Attribut `resulttag` angegeben wurde und das die Metadaten zu jedem Fremdschlüssel in der Tabelle enthält. Wenn es zu einem Abruf mehrere

Ergebnisse gibt (d. h. mehrere Fremdschlüsselspalten), so werden diese jeweils in einem separaten Element ausgegeben.

```
<Data>
  <Ergebniselementname>
    <Fremdschlüssel-Eigenschaft>Wert</Fremdschlüssel-Eigenschaft>
    ...
  </Ergebniselementname>
</Data>
```

 Erläuterungen (englischsprachig) zu den einzelnen Spalten-Eigenschaften finden Sie [hier](#).

Beispiel

In einem Input-XML-Dokument wird folgende XML-Struktur definiert, um die Fremdschlüssel der Tabelle Provider auszugeben:

```
<Root>
  <foreignkeys resulttag="historykeys">
    <table>Provider</table>
  </foreignkeys>
</Root>
```

Der *JDBC Metadata Explorer* gibt daraufhin z. B. folgendes Ergebnis-XML-Dokument aus:

```
<Data>
  <historykeys>
    <PKTABLE_CAT>port</PKTABLE_CAT>
    <PKTABLE_SCHEM>dbo</PKTABLE_SCHEM>
    <PKTABLE_NAME>NETWORK</PKTABLE_NAME>
    <PKCOLUMN_NAME>NETWORK</PKCOLUMN_NAME>
    <FKTABLE_CAT>port</FKTABLE_CAT>
    <FKTABLE_SCHEM>dbo</FKTABLE_SCHEM>
    <FKTABLE_NAME>PROVIDER</FKTABLE_NAME>
    <FKCOLUMN_NAME>NETWORK</FKCOLUMN_NAME>
    <KEY_SEQ>1</KEY_SEQ>
    <UPDATE_RULE>1</UPDATE_RULE>
    <DELETE_RULE>1</DELETE_RULE>
    <FK_NAME>FK_PROVIDER_NETWORK1</FK_NAME>
    <PK_NAME>PK_NETWORK</PK_NAME>
    <DEFERRABILITY>7</DEFERRABILITY>
  </historykeys>
</Data>
```

6.6.3.6 Fremdschlüssel-Referenzen abrufen

Mit dem *JDBC Metadata Explorer* können Sie die Beschreibung der Fremdschlüssel abrufen, die sich in anderen Tabellen befinden und die auf den Primärschlüssel der ausgewählten Tabelle verweisen.

Input


Einem Input-XML-Dokument fügen Sie dazu innerhalb eines beliebigen Wurzel-Elements das Element `<exportedkeys resulttag="Ergebniselementname"/>` hinzu, in dessen Attribut `resulttag` Sie einen gültigen XML-Namen angeben. Die Angabe einer Tabelle in Element `<table>` ist dabei erforderlich; die Elemente `<catalog>` und `<schema>` sind jedoch optional.

```
<Root>
  <exportedkeys resulttag="Ergebniselementname">
    <catalog>Katalog</catalog>
    <schema>DB-Schema</schema>
    <table>DB-Tabelle (Pflichtfeld, darf nicht leer sein)</table>
  </exportedkeys>
</Root>
```

Ausgabe

Die XML-Struktur, die vom Adapter ausgegeben wird, enthält innerhalb des Wurzel-Elements `<Data>` ein Element, dessen Name in der Input-Struktur im Attribut `resulttag` angegeben wurde und das die Metadaten zu jedem Fremdschlüssel in anderen Tabellen enthält, der auf den oder die Primärschlüssel der ausgewählten Tabelle verweist. Wenn es mehrere Ergebnisse gibt (d. h. mehrere Fremdschlüssel), so werden diese jeweils in einem separaten Element ausgegeben.

```
<Data>
  <Ergebniselementname>
    <Fremdschlüssel-Eigenschaft>Wert</Fremdschlüssel-Eigenschaft>
    ...
  </Ergebniselementname>
</Data>
```

 Erläuterungen (englischsprachig) zu den einzelnen Spalten-Eigenschaften finden Sie [hier](#).

Beispiel

In einem Input-XML-Dokument wird folgende XML-Struktur definiert, um die Fremdschlüssel-Referenzen zum Primärschlüssel der Tabelle `Provider` auszugeben:

```
<Root>
  <exportedkeys resulttag="keys">
    <table>Operator</table>
  </exportedkeys>
</Root>
```

Der *JDBC Metadata Explorer* gibt daraufhin z. B. folgendes Ergebnis-XML-Dokument aus:

```

<Data>
  <keys>
    <PKTABLE_CAT>port</PKTABLE_CAT>
    <PKTABLE_SCHEM>dbo</PKTABLE_SCHEM>
    <PKTABLE_NAME>OPERATOR</PKTABLE_NAME>
    <PKCOLUMN_NAME>OPERATOR_ID</PKCOLUMN_NAME>
    <FKTABLE_CAT>port</FKTABLE_CAT>
    <FKTABLE_SCHEM>dbo</FKTABLE_SCHEM>
    <FKTABLE_NAME>ROLEMAP</FKTABLE_NAME>
    <FKCOLUMN_NAME>OPERATOR_ID</FKCOLUMN_NAME>
    <KEY_SEQ>1</KEY_SEQ>
    <UPDATE_RULE>1</UPDATE_RULE>
    <DELETE_RULE>1</DELETE_RULE>
    <FK_NAME>FK_ROLEMAP_OPERATOR</FK_NAME>
    <PK_NAME>PK_OPERATOR</PK_NAME>
    <DEFERRABILITY>7</DEFERRABILITY>
  </keys>
</Data>

```

6.6.4 MongoDB Connector


6.6.4.1 Über den Adapter











MongoDB Connector: Stellt eine Verbindung zu einer MongoDB her und führt ein oder mehrere Datenbank-Statements aus, die als Input-XML-Dokument mit adapterspezifischen Strukturen übergeben wurden.

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.mongodb.MongoAdapter</i> : Hauptklasse (Standard)
<i>ConnectionName</i>	Alias/Referenzname für die Verbindung; Entspricht dem <i>Connection name</i> aus der Verbindungskonfiguration, siehe Verbindung konfigurieren Mögliche Werte: Name der Verbindung (z. B. <i>#default</i> (Standard))

<i>ReadPreference</i>	<p>Read-Preference-Modus; Dieser Parameter definiert, wie Reads bearbeitet werden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>PRIMARY</i>: Alle Operationen lesen aus dem primary-Knoten des aktuellen Replica sets• <i>PRIMARY_PREFERRED</i>: In den meisten Fällen lesen alle Operationen aus dem primary-Knoten, ist dieser jedoch nicht verfügbar, lesen Sie aus den secondary-Knoten (Standard)• <i>SECONDARY</i>: Alle Operationen lesen aus den secondary-Knoten des aktuellen Replica sets• <i>SECONDARY_PREFERRED</i>: In den meisten Fällen lesen alle Operationen aus den secondary-Knoten, sind diese jedoch nicht verfügbar, lesen Sie aus dem primary-Knoten• <i>NEAREST</i>: Die Operationen lesen aus dem Knoten des Replica sets mit der niedrigsten Netzwerk-Latenzzeit, unabhängig vom Knotentyp <div data-bbox="598 981 1436 1393"><p> Zusätzliche Informationen zu den Read-Preference-Modi erhalten Sie auch unter:</p><ul style="list-style-type: none">• http://docs.mongodb.org/manual/core/read-preference/#read-preference-modes• http://api.mongodb.org/java/current/com/mongodb/ReadPreference.html#method-summary</div>
-----------------------	--

<i>WriteConcern</i>	<p>Gibt an, ob eine <code>write</code>-Operation erfolgreich ausgeführt wurde. Write Concerns erlauben es, Eingabefehler oder ungültige mongod-Instanzen zu ermitteln.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• ERRORS_IGNORED: Es werden keine Exceptions ausgeworfen, auch nicht bei Netzwerk-Problemen.  Dieser Write Concern ist veraltet und verfügt über keinen Ersatz. Am ehesten vergleichbar ist der Write Concern UNACKNOWLEDGED. Dabei müssen jedoch alle Exceptions des Typs <i>MongoSocketException</i> über <i>catch and ignore</i> abgefangen werden.• ACKNOWLEDGED: Die Operation wartet auf die Bestätigung des primären Servers, bevor sie ein Ergebnis ausgibt; Exceptions werden sowohl bei Netzwerk-Problemen als auch bei Server-Fehlern ausgeworfen.• UNACKNOWLEDGED: Die Operation gibt das Ergebnis aus, sobald die Nachricht in den Socket geschrieben wird; Exceptions werden lediglich bei Netzwerk-Problemen, jedoch nicht bei Server-Fehlern ausgeworfen.• FSYNCED: Die Operation wartet bis der Server sichergestellt hat, dass die Daten auf der Festplatte angekommen sind; Exceptions werden sowohl bei Netzwerk-Problemen als auch bei Server-Fehlern ausgeworfen.• JOURNALED: Die Operation wartet bis der Server die Daten zu den Journal-Files committet; Exceptions werden sowohl bei Netzwerk-Problemen als auch bei Server-Fehlern ausgeworfen. <div data-bbox="646 1339 1433 1657"><p> Beachten Sie:</p><ul style="list-style-type: none">• Das Journaling muss aktiviert sein, um diesen Write Concern zu verwenden.• Bei diesem Write Concern müssen die Operationen bis zum nächsten Journal-Commit warten. Um die Wartezeit für diese Operationen zu verkürzen, erhöht <i>MongoDB</i> die Commit-Häufigkeit, siehe auch commitIntervalMs.</div> <ul style="list-style-type: none">• REPLICA_ACKNOWLEDGED: Die Operation wartet auf mindestens zwei Server, bevor sie ausgeführt wird; Exceptions werden sowohl bei Netzwerk-Problemen als auch bei Server-Fehlern ausgeworfen.• NONE: Es werden keine Exceptions ausgeworfen, auch nicht bei Netzwerk-Problemen.  Dieser Write Concern ist veraltet und wurde durch den Write Concern ERRORS_IGNORED ersetzt. Dieser kann in kommenden Releases ggf. veraltet sein.
---------------------	---

- **NORMAL:** Die Operation gibt das Ergebnis aus, sobald die Nachricht in den Socket geschrieben wird; Exceptions werden lediglich bei Netzwerk-Problemen, jedoch nicht bei Server-Fehlern ausgeworfen.
 Dieser Write Concern wurde durch den Write Concern UNACKNOWLEDGED ersetzt und kann in kommenden Releases ggf. veraltet sein .
- **SAFE:** Die Operation wartet auf die Bestätigung des primären Servers, bevor sie ein Ergebnis ausgibt; Exceptions werden sowohl bei Netzwerk-Problemen als auch bei Server-Fehlern ausgeworfen.
 Dieser Write Concern wurde durch den Write Concern ACKNOWLEDGED ersetzt und kann in kommenden Releases ggf. veraltet sein .
- **MAJORITY:** Die Operation wartet auf die Mehrheit der Server, bevor sie ausgeführt wird; Exceptions werden sowohl bei Netzwerk-Problemen als auch bei Server-Fehlern ausgeworfen.
- **FSYNC_SAFE:** Die Operation wartet bis der Server sichergestellt hat, dass die Daten auf der Festplatte angekommen sind; Exceptions werden sowohl bei Netzwerk-Problemen als auch bei Server-Fehlern ausgeworfen.
 Dieser Write Concern wurde durch den Write Concern
FSYNCED
ersetzt und kann in
kommenden Releases ggf. veraltet sein .
- **JOURNAL_SAFE:** Die Operation wartet bis der Server die Daten zu den Journal-Files committet; Exceptions werden sowohl bei Netzwerk-Problemen als auch bei Server-Fehlern ausgeworfen.
 Dieser Write Concern wurde durch den Write Concern JOURNALED ersetzt und kann in kommenden Releases ggf. veraltet sein .
- **REPLICAS_SAFE:** Die Operation wartet auf mindestens zwei Server, bevor sie ausgeführt wird; Exceptions werden sowohl bei Netzwerk-Problemen als auch bei Server-Fehlern ausgeworfen.
 Dieser Write Concern wurde durch den Write Concern REPLICA_ACKNOWLEDGED ersetzt und kann in kommenden Releases ggf. veraltet sein .



Zusätzliche Informationen zu den Write Concerns erhalten Sie auch unter:

- <http://docs.mongodb.org/manual/core/write-concern/#write-operations-write-concern>
- <http://docs.mongodb.org/ecosystem/drivers/java-replica-set-semantics/>

- http://api.mongodb.org/java/current/com/mongodb/WriteConcern.html#field_summary

6.6.4.2 Verbindung konfigurieren

Die Verbindung zur *MongoDB* wird in der `mongodb-connections.xml` innerhalb des X4-Installationsordners konfiguriert, z. B. `X4\Server\X4DB\0\mongodb-connections.xml`.

```
<MongoDB>
  <!-- connects to localhost on the default port without authentication -->
  <Connection name="#default" db="test"/>

  <Connection name="localhost" db="test">
    <Server host="localhost" port="27017"/>
  </Connection>
</MongoDB>
```

Erläuterung:

- /MongoDB/Connection: Stellt eine benannte Verbindung dar; Kann mehrfach existieren
- /MongoDB/Connection/@name: Alias/Referenzname für die Verbindung
- /MongoDB/Connection/@db: Name der Datenbank zu der die Verbindung hergestellt werden soll, d.h. Datenbanken müssen immer über externe Werkzeuge (z.B. *MongoShell*) angelegt werden
- /MongoDB/Connection/Server: Stellt einen Servereintrag für eine Verbindung dar; Kann mehrfach pro Connection existieren, um einen Cluster zu adressieren
- /MongoDB/Connection/Server/@host: IP/DNS-Name des Servers zu dem sich verbunden werden soll (default: 127.0.0.1)
- /MongoDB/Connection/Server/@port: Port des Servers zu dem sich verbunden werden soll (default: 27017)
- /MongoDB/Connection/Credential: Anmeldedaten für eine Verbindung; Kann mehrfach pro Verbindung existieren
- /MongoDB/Connection/Credential/@user: Benutzername der Anmeldung
- /MongoDB/Connection/Credential/@password: Passwort der Anmeldung
- /MongoDB/Connection/Credential/@source: Ursprung der Anmeldedaten (Nutzung zZt. unklar)



Beachten Sie:

Besitzt eine Verbindung keine Server-Elemente, wird ohne Authentifizierungsdaten gegen `localhost:27017` verbunden, unabhängig davon, ob Credential-Elemente existieren oder nicht.

6.6.4.3 Kodierung einer Dokumentenstruktur

Die *JSON/Mongo*-Dokumente werden nach einfachen Regeln in ihr XML-Äquivalent übersetzt:


- Dokumente und Objekte/Substrukturen werden durch `<Object>` dargestellt. Die Schlüssel-Wert-Einträge eines Objekts werden als Kindelemente des `Object`-Elements abgebildet.
- Arrays werden durch `<Array>` dargestellt. Die Array-Einträge werden als Kindelemente des `Array`-Elements abgebildet.
- Wenn es kein Array oder Object ist, wird es durch `<Value>` dargestellt.
- Der Datentyp eines Values wird in dem Attribut `type` auf dem `Value`-Element festgehalten (siehe [Unterstützte Datentypen](#)). Wenn kein Datentyp angegeben ist, wird `String` angenommen.
- Die Schlüssel-Wert-Einträge eines Objektes oder eine Substruktur bekommen den Schlüssel als `name`-Attribute auf das Element des Wert-Elements gesetzt.

Invarianten:

- Für jede `{` im JSON wird ein `Object`-Element im XML angelegt. (Ausnahme: die Dokumentenwurzel wird in Kommandos durch andere Elemente abgebildet)
- Für jede `[` im JSON wird ein `Array`-Element im XML angelegt.
- Für jeden `:` im JSON (der nicht Bestandteil eines String Literals ist) wird ein `name`-Attribut angelegt das den Wert vor dem `:` als Inhalt enthält.
- Alle Kindelemente eines `Object`-Elements haben ein `name`-Attribut.
- Kindelemente eines `Array`-Elements haben kein `name`-Attribut
- `Value`-Elemente haben kein Kindelement.
- Text-Knoten existieren nur als Kinder von `Value`-Elementen (Ausnahme: formatierender Whitespace).
- `type`-Attribute existieren nur auf `Value`-Elementen (aber nicht jedes `Value`-Element muss ein `type`-Attribute haben).

Beispiel

```
<Object>
  <Value name="some" type="String">thing</Value>
</Object>
```

 Die Dokumenten-Wurzel kann auch durch andere Elemente ersetzt werden, um die Erzeugung eines `Object`-Elements zu vermeiden. In diesem Fall wäre dort immer ein `Object`-Element notwendig.

6.6.4.4 Unterstützte Datentypen

Der Adapter unterstützt folgende Datentypen:

- `Null`
- `ObjectId`
- `Boolean`
- `Long`
- `Int`
- `Double`
- `String`

- `DateTime`
- `Integer`: String wird als `BigInteger` gelesen und anschließend als `Long` gespeichert
- `Decimal`: String wird als `BigDecimal` gelesen und anschließend als `Double` gespeichert
- `Numeric`: wenn der String einen `.` enthält, wird er als `Decimal` gelesen, sonst als `Integer`
- `Regex`: Regulärer Ausdruck gemäß <http://docs.oracle.com/javase/7/docs/api/java/util/regex/Pattern.html>

❗ Dieses `Value`-Element kennt zudem das Attribut `flags`, welches einen String aus folgenden Zeichen ist und die entsprechende Option auf dem Muster setzt:

- `c`: `CANON_EQ`
- `i`: `CASE_INSENSITIVE`
- `m`: `MULTILINE`
- `s`: `DOTALL`
- `u`: `UNICODE_CASE`
- `L`: `LITERAL`
- `N`: `UNIX_LINES`
- `U`: `UNICODE_CHARACTER_CLASS`
- `Z`: `COMMENTS`

6.6.4.5 Input

Der *MongoDB Adapter* erwartet spezifische Input-XML-Strukturen. Diese Input-XML-Strukturen enthalten eine Wurzel mit mehreren Kommandos darunter. Ebenso wird auch das `resulttag`-Attribut unterstützt.

Input-Struktur

```
<!-- Ein Root-Element mit beliebigem Namen. -->
<root>
  <insert collection="products" resulttag="insertOutput">
    <Value name="item" type="String">card</Value>
    <Value name="qty" type="Integer">15</Value>
  </insert>
  <find collection="products" resulttag="findOutput">
    <query>
      <Value name="item" type="String">card</Value>
    </query>
  </find>
  <!-- Hier können weitere Kommandos angegeben werden. -->
</root>
```

❗ Dies ist ein vollständiges und ausführbares Beispiel. Alle nachfolgenden Beispiele sind lediglich Ausschnitte, die zur Ausführung in eine vollständige Input-Struktur mit einem umschließenden Root-Element eingefügt werden müssen.

Unterstützte Kommandos

- `copy` (kein MongoDB Kommando)

- find
- insert
- remove
- save
- update

6.6.4.5.1 Kommando copy

Mit copy werden alle Kindelemente des copy-Kommandos in die Ausgabe kopiert.

Input

Das copy-Kommando hat folgende Syntax:

```
<copy resulttag="output">
  <Child1/>
  <Child2>
    <Grandchild1/>
    <Grandchild2/>
  </Child2>
</copy>
```

Output

Dieses Kommando gibt folgende XML-Struktur aus:

```
<output>
  <Child1/>
  <Child2>
    <Grandchild1/>
    <Grandchild2/>
  </Child2>
</output>
```

Kommando insert

Mit insert wird ein Dokument in die Collection eingefügt. Das insert-Element ist dabei gleichzeitig die Dokumentenwurzel.



- Bitte beachten Sie, dass Batchinsert mit insert nicht unterstützt wird.
- Weitere Informationen zu diesem Kommando finden Sie auch unter <http://docs.mongodb.org/manual/reference/method/db.collection.insert/#db.collection.insert>.

Input

Das insert-Kommando hat folgende Syntax:

```
<insert collection="foo">
  <Value name="bar" type="String">baz</Value>
</insert>
```

Output

Dieses Kommando gibt ein `WriteResult`-Objekt aus, das den Status der Operation enthält.

6.6.4.5.2 Kommando `save`

Mit `save` wird gemäß mitgegebener Daten ein neues Dokument angelegt oder ein bestehendes Dokument aktualisiert. Das `save`-Element ist dabei gleichzeitig die Dokumentenwurzel.

ⓘ Weitere Informationen zu diesem Kommando finden Sie auch unter <http://docs.mongodb.org/manual/reference/method/db.collection.save/#db.collection.save>.

Input

Das

`save`
-Kommando hat folgende Syntax:

```
<save collection="foo">
  <Value name="bar" type="String">baz</Value>
</save>
```

Output

Dieses Kommando gibt ein `WriteResult`-Objekt aus, das den Status der Operation enthält.

6.6.4.5.3 Kommando `update`

Mit `update` wird ein Dokumente in einer Collection modifiziert. Je nach `update`-Parameter kann die Methode spezielle Felder eines bereits bestehenden Dokuments oder ein komplettes Dokument modifizieren.

ⓘ Weitere Informationen zu diesem Kommando finden Sie auch unter <http://docs.mongodb.org/manual/reference/method/db.collection.update/#db.collection.update>.

Input

Das `update`-Kommando hat folgende Syntax:

```
<update collection="foo">
  <query/>
  <update upsert="false" multi="false"/>
</update>
```

Erläuterung:

- Das `query`-Element enthält den Abfrage/Selektionsteil des `updates` und ist auch Dokumentenwurzel für die Abfrage.
- Das `update`-Element enthält den Veränderungsteil des `Updates`, und ist auch Dokumentenwurzel für die Veränderung.
- Die Optionen `upsert` und `multi` werden auf dem inneren `update`-Element angegeben.

Output

Dieses Kommando gibt ein `WriteResult`-Objekt aus, das den Status der Operation enthält.

6.6.4.5.4 Kommando `remove`

Mit `remove` wird ein Dokumente aus einer Collection entfernt.

① Weitere Informationen zu diesem Kommando finden Sie auch unter <http://docs.mongodb.org/manual/reference/method/db.collection.remove/#db.collection.remove>.

Input

Das `remove`-Kommando hat folgende Syntax:

```
<remove collection="foo">  
  <query/>  
</remove>
```

Erläuterung:

- Das `query`-Element enthält den Abfrage/Selektionsteil des `remove` und ist auch Dokumentenwurzel für die Abfrage.

① Beachten Sie, dass die Option `justOne` durch den Java-Treiber nicht unterstützt wird.

Output

Dieses Kommando gibt ein `WriteResult`-Objekt aus, das den Status der Operation enthält.

6.6.4.5.5 Kommando `find`

Mit `find` wird ein Dokumente aus einer Collection selektiert.

① Weitere Informationen zu diesem Kommando finden Sie auch unter <http://docs.mongodb.org/manual/reference/method/db.collection.find/#db.collection.find>.

Input

Das `find`-Kommando hat folgende Syntax:

```

<find collection="foo">
  <!--! Optional elements -->
  <query/>
  <projection/>
  <sort/>
  <hint/>
  <min/>
  <max/>
  <special/>
  <cursor skip="0" limit="0" batchSize="0"/>
</find>

```

Erläuterung:

- Das query-Element enthält den Abfrage/Selektionsteil des find und ist auch Dokumentenwurzel für die Abfrage.
- Das projection-Element enthält die Feldauswahl des find und ist auch Dokumentenwurzel für die Auswahl.

Die weiteren optionalen Elemente steuern das Verhalten des Ergebnis-Cursors:

- sort (Dokumentenwurzel), siehe auch <http://docs.mongodb.org/manual/reference/method/cursor.sort/#cursor.sort>
- hint (Dokumentenwurzel), siehe auch <http://docs.mongodb.org/manual/reference/method/cursor.hint/#cursor.hint>
- min (Dokumentenwurzel), siehe auch <http://docs.mongodb.org/manual/reference/method/cursor.min/#cursor.min>
- max (Dokumentenwurzel), siehe auch <http://docs.mongodb.org/manual/reference/method/cursor.max/#cursor.max>
- special (quasi-Dokumentenwurzel), siehe auch [http://api.mongodb.org/java/current/com/mongodb/DBCursor.html#addSpecial\(java.lang.String,%20java.lang.Object\)](http://api.mongodb.org/java/current/com/mongodb/DBCursor.html#addSpecial(java.lang.String,%20java.lang.Object))
-  Für jeden Kindknoten in der Wurzel des special-Dokuments wird die addSpecial-Methode aufgerufen.
- cursor (no body) ist lediglich der Träger für folgende drei Attribute:
 - skip, siehe auch <http://docs.mongodb.org/manual/reference/method/cursor.skip/#cursor-skip>
 - limit, siehe auch <http://docs.mongodb.org/manual/reference/method/cursor.limit/#cursor.limit>
 - batchSize, siehe auch <http://docs.mongodb.org/manual/reference/method/cursor.batchSize/#cursor.batchSize>

Output

Dieses Kommando gibt einen Cursor aus, der auf die Dokumente zeigt, welche die query-Kriterien erfüllen.

6.6.4.6 Beispiele

Folgende Beispiele zeigen Mongo-Ausdrücke und ihre jeweiligen Entsprechungen in XML.

Beispiele für die Dokumentenstruktur:

JSON/Mongo-Struktur	Entsprechende XML-Struktur
<pre>{}</pre>	<pre><Object/></pre>
<pre>{ type: "snacks" }</pre>	<pre><Object> <Value name="type" type="String">snacks</Value> </Object></pre>
<pre>{ type: { \$in: ['food', 'snacks'] } }</pre>	<pre><Object> <Object name="type"> <Array name="type"> <Value type="String">food</Value> <Value type="String">snacks</Value> </Array> </Object> </Object></pre>
<pre>{ type: 'food', price: { \$lt: 9.95 } }</pre>	<pre><Object> <Value name="type" type="String">food</Value> <Object name="price"> <Value name="\$lt" type="Double">9.95</Value> </Object> </Object></pre>
<pre>{ \$or: [{ qty: { \$gt: 100 } }, { price: { \$lt: 9.95 } }]</pre>	<pre><Object> <Array name="\$or"> <Object> <Object name="qty"> <Value name="\$gt" type="Integer">100</Value> </Object> </Object> <Object> <Object name="price"> <Value name="\$lt" type="Double">9.95</Value> </Object> </Object> </Array> </Object></pre>

Beispiele für Aufrufe:

Mongo-Aufruf	Entsprechende XML-Struktur
<pre>db.products.insert({ item: "card", qty: 15 })</pre>	<pre><insert collection="products"> <Value name="item" type="String">card</ Value> <Value name="qty" type="Integer">15</ Value> </insert></pre>
<pre>db.inventory.save({ type: "book", item: "notebook", qty: 40 })</pre>	<pre><save collection="inventory"> <Value name="type" type="String">book</ Value> <Value name="item" type="String">noteboo k</Value> <Value name="qty" type="Integer">40</ Value> </save></pre>
<pre>db.books.update({ item: "The Banquet" }, { item: "The Banquet", price: 19 , stock: 3 })</pre>	<pre><update collection="books"> <query> <Value name="item" type="String">The Banquet</Value> </query> <update upsert="true"> <Value name="item" type="String">The Banquet</Value> <Value name="price" type="Double">19< /Value> <Value name="stock" type="Integer">3< /Value> </update> </update></pre>
<pre>db.books.update({ item: "Divine Comedy" }, { \$set: { price: 18 }, \$inc: { stock: 5 } })</pre>	<pre><update collection="books"> <query> <Value name="item" type="String">Div ine Comedy</Value> </query> <update upsert="true"> <Object name="\$set"> <Value name="price" type="Double">19</Value> </Object> <Object name="\$inc"> <Value name="stock" type="Integer">5</Value> </Object> </update> </update></pre>

Mongo-Aufruf	Entsprechende XML-Struktur
<pre>db.inventory.update({ type: "book", item : "journal" }, { \$set : { qty: 10 } }, { upsert : true })</pre>	<pre><update collection="inventory"> <query> <Value name="type" type="String">book</Value> <Value name="item" type="String">journal</Value> </query> <update upsert="true"> <Object name="\$set"> <Value name="qty" type="Integer">10</Value> </Object> </update> </update></pre>
<pre>db.inventory.update({ type : "book" }, { \$inc : { qty : -1 } }, { multi: true })</pre>	<pre><update collection="inventory"> <query> <Value name="type" type="String">book</Value> </query> <update multi="true"> <Object name="\$inc"> <Value name="qty" type="Integer">-1</Value> </Object> </update> </update></pre>
<pre>db.inventory.find()</pre>	<pre><find collection="inventory" /></pre>
<pre>db.inventory.find({})</pre>	<pre><find collection="inventory"> <query/> </find></pre>
<pre>db.inventory.find({ type: "snacks" })</pre>	<pre><find collection="inventory"> <query> <Value name="type" type="String">snacks</Value> </query> </find></pre>

Mongo-Aufruf	Entsprechende XML-Struktur
<pre>db.inventory.find({ type: 'food' }, { _id: 0, item: 1, qty: 1 })</pre>	<pre><find collection="inventory"> <query> <Value name="type" type="String">food</Value> </query> <projection> <Value name="_id" type="Integer">0</Value> <Value name="item" type="Integer">1</Value> <Value name="qty" type="Integer">1</Value> </projection> </find></pre> <p>oder</p> <pre><find collection="inventory"> <query> <Value name="type" type="String">food</Value> </query> <projection> <exclude name="_id" /> <include name="item" /> <include name="qty" /> <!-- excludes und includes können mit Object, Array und Value im projection- Element gemischt werden --> </projection> </find></pre>

Mongo-Aufruf	Entsprechende XML-Struktur
<pre>db.stocks.find().sort({ ticker: 1, date: -1 })</pre>	<pre><find collection="stocks"> <sort> <Value name="ticker" type="Integer">1< /Value> <Value name="date" type="Integer">-1< /Value> </sort> </find></pre> <p>oder</p> <pre><find collection="stocks"> <sort> <ascending name="ticker" /> <descending name="date" /> <!-- ascending und descending können mit Object, Array und Value im sort-Element gemischt werden --> </sort> </find></pre>
<pre>db.stocks.find().sort({ ticker: 1 }).limit(100)</pre>	<pre><find collection="stocks"> <sort> <ascending name="ticker" /> </sort> <cursor limit="100"/> </find></pre>
<pre>db.stocks.find().sort({ ticker: 1 }).skip(200).limit(100)</pre>	<pre><find collection="stocks"> <sort> <ascending name="ticker" /> </sort> <cursor skip="200" limit="100"/> </find></pre>

6.7 EDI

6.7.1 EDIFACT Adapter



EDIFACT Adapter: Liest eine EDIFACT-Nachricht (Rohdaten und Stage 1-3, *DIN ISO 9735*-konform) und gibt sie als UTF-8-kodiertes XML-Dokument aus. Außerdem kann der Adapter eine EDIFACT-Nachricht ausgeben.

Ehemalige Bezeichnung bis Version 4.5.1: *EDIFACT Message Adapter*

Ehemalige Bezeichnung bis Version 5.5.4: *EDIFACT Converter*

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>ReadRaw</i>: EDIFACT-Nachricht 1:1 als XML ausgeben • <i>ReadStage1</i>: EDIFACT-Nachricht als XML mit konkreten Elementnamen einlesen (<i>Stage 1</i>) • <i>ReadStage2</i>: EDIFACT-Nachricht als XML einlesen, in dem die Segmente anhand der Nachrichten organisiert sind (<i>Stage 2</i>) • <i>ReadStage3</i>: EDIFACT-Nachricht als XML einlesen, in dem die Segmente mit der korrekten Hierarchie inklusive verschachtelten Segmentgruppen im Baum gegliedert sind, sofern der Adapter den Nachrichtentyp unterstützt (<i>Stage 3</i>) • <i>Write</i>: EDIFACT-Nachricht schreiben
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.edifact.EdifactAdapter</i> : Hauptklasse (Standard)
----------------	---

Input

Eine EDIFACT-Nachricht vom Typ INVOIC (Rechnung) könnte wie folgt aufgebaut sein:

```
UNA:+,? '
UNB+UNOC:2+FHPELAL+HUBERGMBH+990802:1557+9908021557'
UNH+INVOIC0001+INVOIC:D:93A:UN'
BGM+380+9908001+9'
DTM+3:19990802:102'
RFF+ON:00010001'
DTM+4:19999715:102'
NAD+SE++Fahrradhandel Pedal++Wagingerstr. 5+München++81549'
NAD+BY++Huber GmbH++Obstgasse 2+München++81549'
LIN+1++4711.001'
IMD+F+++::Fahrrad, Damen'
QTY+47:1:PCE'
MOA+66:750'
PRI+AAA:750'
LIN+2++4711.002'
IMD+F+++::Luftpumpe, Stand-'
QTY+47:1:PCE'
MOA+66:19,9'
PRI+AAA:19,9'
LIN+3++4711.003'
IMD+F+++::Ersatzventil'
QTY+47:3:PCE'
MOA+66:7,5'
PRI+AAA:2,5'
UNS+S'
MOA+79:777,4'
MOA+124:124,38'
MOA+128:901,78'
TAX+7+VAT+++::16+S'
UNT+28+INVOIC0001'
UNZ+1+9908021557'
```

Ausgabe

Mit Operation ReadRaw werden EDIFACT-Nachrichten 1:1 als XML ausgegeben, z. B.:

```

<EDIFACT decimalSeparator="," segmentTerminator=""
  releaseIndicator="?" dataSeparator="+" componentSeparator=":">
  <Segment>
    <Component>
      <Data>UNB</Data>
    </Component>
    <Component>
      <Data>UNOC</Data>
      <Data>2</Data>
    </Component>
    <Component>
      <Data>FHPEDAL</Data>
    </Component>
    <Component>
      <Data>HUBERGMBH</Data>
    </Component>
    <Component>
      <Data>990802</Data>
      <Data>1557</Data>
    </Component>
    <Component>
      <Data>9908021557</Data>
    </Component>
  </Segment>
  <!-- weitere Segmente -->
</EDIFACT>

```

Mit Operation ReadStage1 werden EDIFACT-Nachrichten als XML-Dokumente ausgegeben, die bereits statt generischen Elementnamen konkrete Namen verwenden, z. B.:

```

<EDIFACT decimalSeparator="," segmentTerminator=""
  releaseIndicator="?" dataSeparator="+" componentSeparator=":">
  <UNB>
    <Component>
      <Data>UNOC</Data>
      <Data>2</Data>
    </Component>
    <Component>
      <Data>FHPEDAL</Data>
    </Component>
    <Component>
      <Data>HUBERGMBH</Data>
    </Component>
    <Component>
      <Data>990802</Data>
      <Data>1557</Data>
    </Component>
    <Component>
      <Data>9908021557</Data>
    </Component>
  </UNB>
  <!-- weitere Segmente -->
</EDIFACT>

```

Mit Operation ReadStage2 werden EDIFACT-Nachrichten als XML-Dokumente ausgegeben, in denen die Segmente anhand der Nachrichten organisiert sind, z. B.:

```
<EDIFACT decimalSeparator="," segmentTerminator="'"
  releaseIndicator="?" dataSeparator="+" componentSeparator=":">
  <UNB>
    <Component>
      <Data>UNOC</Data>
      <Data>2</Data>
    </Component>
    <!--weitere Komponenten-->
    <UNH>
      <Component>
        <Data>INVOIC0001</Data>
      </Component>
      (weitere Komponenten)
      <BGM>
        <Component>
          <Data>380</Data>
        </Component>
        <Component>
          <Data>9908001</Data>
        </Component>
        <Component>
          <Data>9</Data>
        </Component>
      </BGM>
      <!--weitere Segmente-->
      <UNT>
        <Component>
          <Data>28</Data>
        </Component>
        <Component>
          <Data>INVOIC0001</Data>
        </Component>
      </UNT>
    </UNH>
    <UNZ>
      <Component>
        <Data>1</Data>
      </Component>
      <Component>
        <Data>9908021557</Data>
      </Component>
    </UNZ>
  </UNB>
</EDIFACT>
```

Mit Operation ReadStage3 werden EDIFACT-Nachrichten als XML-Dokumente ausgegeben, in denen die Segmente mit der korrekten Hierarchie inklusive verschachtelten Segmentgruppen im Baum gegliedert sind, sofern der Adapter den Nachrichtentyp unterstützt, z. B.:

```
<EDIFACT decimalSeparator="," segmentTerminator=""
  releaseIndicator="?" dataSeparator="+" componentSeparator=":">
  <UNB>
    <Component>
      <Data>UNOC</Data>
      <Data>2</Data>
    </Component>
    <!--weitere Komponenten-->
    <UNH>
      <Component>
        <Data>INV0IC0001</Data>
      </Component>
      <Component>
        <Data>INV0IC</Data>
        <Data>D</Data>
        <Data>93A</Data>
        <Data>UN</Data>
      </Component>
      <BGM>
        <Component>
          <Data>380</Data>
        </Component>
        <Component>
          <Data>9908001</Data>
        </Component>
        <Component>
          <Data>9</Data>
        </Component>
      </BGM>
      <DTM>
        <Component>
          <Data>3</Data>
          <Data>19990802</Data>
          <Data>102</Data>
        </Component>
      </DTM>
      <!--weitere Segmente-->
      <SegmentGroup22>
        <LIN>
          <Component>
            <Data>1</Data>
          </Component>
          <Component>
            <Data />
          </Component>
          <Component>
            <Data>4711.001</Data>
          </Component>
        </LIN>
        <IMD>
          <Component>
            <Data>F</Data>
          </Component>
          <Component>
            <Data />
          </Component>
```

```
<Component>
  <Data />
  <Data />
  <Data />
  <Data>Fahrrad, Damen</Data>
</Component>
</IMD>
<QTY>
  <Component>
    <Data>47</Data>
    <Data>1</Data>
    <Data>PCE</Data>
  </Component>
</QTY>
<SegmentGroup23>
  <MOA>
    <Component>
      <Data>66</Data>
      <Data>750</Data>
    </Component>
  </MOA>
</SegmentGroup23>
<SegmentGroup25>
  <PRI>
    <Component>
      <Data>AAA</Data>
      <Data>750</Data>
    </Component>
  </PRI>
</SegmentGroup25>
</SegmentGroup22>
<!--weitere Segmentgruppen-->
<UNT>
  <Component>
    <Data>28</Data>
  </Component>
  <Component>
    <Data>INV0IC0001</Data>
  </Component>
</UNT>
</UNH>
<UNZ>
  <Component>
    <Data>1</Data>
  </Component>
  <Component>
    <Data>9908021557</Data>
  </Component>
</UNZ>
</UNB>
</EDIFACT>
```

Siehe auch:

- [Adapter](#)
- [Unterstützte EDIFACT-Nachrichten-Typen](#)

6.7.2 Unterstützte EDIFACT-Nachrichten-Typen

In der X4 Suite können mithilfe des [EDIFACT Adapter](#)-Adapters eine Reihe von EDIFACT-Nachrichten bearbeitet werden. Die unterstützten Nachrichtentypen werden stetig erweitert.

Derzeit werden folgende Nachrichtentypen und Revisionen unterstützt:

APERAK	<ul style="list-style-type: none"> • APERAK Rev 1: APERAK*D*95A*UN - APERAK*D*96A*UN • APERAK Rev 2: APERAK*D*96B*UN - APERAK*D*97A*UN • APERAK Rev 3: APERAK*D*97B*UN - APERAK*D*98B*UN • APERAK Rev 4: APERAK*D*99A*UN - APERAK*D*13B*UN
UTILMD	<ul style="list-style-type: none"> • UTILMD Rev 1: UTILMD*D*01A*UN - UTILMD*D*01B*UN • UTILMD Rev 2: UTILMD*D*01C*UN - UTILMD*D*02A*UN • UTILMD Rev 3: UTILMD*D*02B*UN - UTILMD*D*03B*UN • UTILMD Rev 4: UTILMD*D*04A*UN - UTILMD*D*05B*UN • UTILMD Rev 5: UTILMD*D*06A*UN - UTILMD*D*07A*UN • UTILMD Rev 6: UTILMD*D*07B*UN - UTILMD*D*13B*UN • UTILMD custom: UTILMD*D*02B*ZZ
IFCSUM	<ul style="list-style-type: none"> • IFCSUM Rev ?? (jedoch ohne IFCSUM*D*93A*UN) • IFCSUM Rev 3 (jedoch ohne IFCSUM*D*95A*UN - IFCSUM*D*95B*UN) • IFCSUM Rev 4 (jedoch ohne IFCSUM*D*96A*UN) • IFCSUM Rev 5: IFCSUM*D*96B*UN • IFCSUM Rev 6: IFCSUM*D*97A*UN • IFCSUM Rev 7: IFCSUM*D*97B*UN • IFCSUM Rev 8: IFCSUM*D*98A*UN • IFCSUM Rev 9: IFCSUM*D*98B*UN - IFCSUM*D*99A*UN • IFCSUM Rev 10: IFCSUM*D*99B*UN • IFCSUM Rev 11: IFCSUM*D*00A*UN • IFCSUM Rev 12: IFCSUM*D*00B*UN - IFCSUM*D*01B*UN • IFCSUM Rev 13: IFCSUM*D*01C*UN - IFCSUM*D*02B*UN • IFCSUM Rev 14: IFCSUM*D*03A*UN - IFCSUM*D*08A*UN • IFCSUM Rev 15: IFCSUM*D*08B*UN - IFCSUM*D*10A*UN • IFCSUM Rev 16: IFCSUM*D*10B*UN - IFCSUM*D*13B*UN

IFTMBF	<ul style="list-style-type: none"> • IFTMBF Rev ??: IFTMBF*D*93A*UN • IFTMBF Rev 1(jedoch ohne IFTMBF*D*94A*UN) • IFTMBF Rev 2(jedoch ohne IFTMBF*D*94B*UN) • IFTMBF Rev 3: IFTMBF*D*95A*UN • IFTMBF Rev 4(jedoch ohne IFTMBF*D*95B*UN) • IFTMBF Rev 5: IFTMBF*D*96A*UN - IFTMBF*D*96B*UN • IFTMBF Rev 6: IFTMBF*D*97A*UN • IFTMBF Rev 7: IFTMBF*D*97B*UN • IFTMBF Rev 8: IFTMBF*D*98A*UN - IFTMBF*D*99A*UN • IFTMBF Rev 9: IFTMBF*D*99B*UN • IFTMBF Rev 10: IFTMBF*D*00A*UN - IFTMBF*D*08A*UN • IFTMBF Rev 11: IFTMBF*D*08B*UN - IFTMBF*D*10A*UN • IFTMBF Rev 12: IFTMBF*D*10B*UN - IFTMBF*D*13B*UN
IFTMCS	<ul style="list-style-type: none"> • IFTMCS Rev ??: IFTMCS*D*93A*UN • IFTMCS Rev 1(jedoch ohne IFTMCS*D*94A*UN) • IFTMCS Rev 2(jedoch ohne IFTMCS*D*94B*UN) • IFTMCS Rev 3: IFTMCS*D*95A*UN • IFTMCS Rev 4(jedoch ohne IFTMCS*D*95B*UN) • IFTMCS Rev 5: IFTMCS*D*96A*UN - IFTMCS*D*96B*UN • IFTMCS Rev 6: IFTMCS*D*97A*UN • IFTMCS Rev 7: IFTMCS*D*97B*UN • IFTMCS Rev 8: IFTMCS*D*98A*UN - IFTMCS*D*99A*UN • IFTMCS Rev 9: IFTMCS*D*99B*UN • IFTMCS Rev 10: IFTMCS*D*00A*UN - IFTMCS*D*08A*UN • IFTMCS Rev 11: IFTMCS*D*08B*UN - IFTMCS*D*10A*UN • IFTMCS Rev 12: IFTMCS*D*10B*UN - IFTMCS*D*13B*UN

IFTSTA	<ul style="list-style-type: none"> • IFTSTA Rev ??: IFTSTA*D*93A*UN • IFTSTA Rev 1: IFTSTA*D*94A*UN • IFTSTA Rev 2 (jedoch ohne IFTSTA*D*94B*UN) • IFTSTA Rev 3: IFTSTA*D*95A*UN • IFTSTA Rev 4: IFTSTA*D*95B*UN • IFTSTA Rev 5: IFTSTA*D*96A*UN • IFTSTA Rev 6: IFTSTA*D*96B*UN - IFTSTA*D*97A*UN • IFTSTA Rev 7: IFTSTA*D*97B*UN • IFTSTA Rev 8: IFTSTA*D*98A*UN - IFTSTA*D*99A*UN • IFTSTA Rev 9: IFTSTA*D*99B*UN • IFTSTA Rev 10: IFTSTA*D*00A*UN - IFTSTA*D*00B*UN • IFTSTA Rev 11: IFTSTA*D*01A*UN - IFTSTA*D*01B*UN • IFTSTA Rev 12: IFTSTA*D*01C*UN - IFTSTA*D*02B*UN • IFTSTA Rev 13: IFTSTA*D*03A*UN • IFTSTA Rev 14: IFTSTA*D*03B*UN - IFTSTA*D*08A*UN • IFTSTA Rev 15: IFTSTA*D*08B*UN - IFTSTA*D*12A*UN • IFTSTA Rev 16: IFTSTA*D*12B*UN • IFTSTA Rev 17: IFTSTA*D*13A*UN - IFTSTA*D*13B*UN
INVOIC	<ul style="list-style-type: none"> • INVOIC Rev 1: INVOIC*D*93A*UN - INVOIC*D*94A*UN • INVOIC Rev 2: INVOIC*D*94B*UN • INVOIC Rev 3: INVOIC*D*95A*UN - INVOIC*D*95B*UN • INVOIC Rev 4: INVOIC*D*96A*UN • INVOIC Rev 5: INVOIC*D*96B*UN • INVOIC Rev 6: INVOIC*D*97A*UN • INVOIC Rev 7: INVOIC*D*97B*UN • INVOIC Rev 8: INVOIC*D*98A*UN • INVOIC Rev 9: INVOIC*D*98B*UN • INVOIC Rev 10: INVOIC*D*99A*UN • INVOIC Rev 11: INVOIC*D*99B*UN - INVOIC*D*00A*UN • INVOIC Rev 12: INVOIC*D*00B*UN • INVOIC Rev 13: INVOIC*D*01A*UN - INVOIC*D*02A*UN • INVOIC Rev 14: INVOIC*D*02B*UN - INVOIC*D*07A*UN • INVOIC Rev 15: INVOIC*D*07B*UN - INVOIC*D*12B*UN • INVOIC Rev 16: INVOIC*D*13A*UN - INVOIC*D*13B*UN
MSCONS	<ul style="list-style-type: none"> • MSCONS Rev 1: MSCONS*D*97A*UN - MSCONS*D*99A*UN • MSCONS Rev 2: MSCONS*D*99B*UN - MSCONS*D*13B*UN

ORDERS	<ul style="list-style-type: none"> • ORDERS Rev ??: ORDERS*D*93A*UN • ORDERS Rev 1: ORDERS*D*94A*UN • ORDERS Rev 2: ORDERS*D*94B*UN • ORDERS Rev 3: ORDERS*D*95A*UN • ORDERS Rev 4: ORDERS*D*95B*UN • ORDERS Rev 5: ORDERS*D*96A*UN • ORDERS Rev 6: ORDERS*D*96B*UN • ORDERS Rev 7: ORDERS*D*97A*UN • ORDERS Rev 8: ORDERS*D*97B*UN - ORDERS*D*98A*UN • ORDERS Rev 9: ORDERS*D*98B*UN • ORDERS Rev 10: ORDERS*D*99A*UN • ORDERS Rev 11: ORDERS*D*99B*UN - ORDERS*D*00A*UN • ORDERS Rev 12: ORDERS*D*00B*UN - ORDERS*D*01C*UN • ORDERS Rev 13: ORDERS*D*02A*UN • ORDERS Rev 14: ORDERS*D*02B*UN - ORDERS*D*05B*UN • ORDERS Rev 15: ORDERS*D*06A*UN - ORDERS*D*06B*UN • ORDERS Rev 16: ORDERS*D*07A*UN - ORDERS*D*13B*UN
NOMINT	NOMINT*4*0*EG
ORDRSP	<ul style="list-style-type: none"> • ORDRSP Rev ??: ORDRSP*D*93A*UN • ORDRSP Rev 1: ORDRSP*D*94A*UN • ORDRSP Rev 2: ORDRSP*D*94B*UN • ORDRSP Rev 3: ORDRSP*D*95A*UN - ORDRSP*D*95B*UN • ORDRSP Rev 4: ORDRSP*D*96A*UN • ORDRSP Rev 5: ORDRSP*D*96B*UN • ORDRSP Rev 6: ORDRSP*D*97A*UN • ORDRSP Rev 7: ORDRSP*D*97B*UN • ORDRSP Rev 8: ORDRSP*D*98A*UN • ORDRSP Rev 9: ORDRSP*D*98B*UN • ORDRSP Rev 10: ORDRSP*D*99A*UN • ORDRSP Rev 11: ORDRSP*D*99B*UN - ORDRSP*D*00A*UN • ORDRSP Rev 12: ORDRSP*D*00B*UN - ORDRSP*D*01C*UN • ORDRSP Rev 13: ORDRSP*D*02A*UN • ORDRSP Rev 14: ORDRSP*D*02B*UN - ORDRSP*D*05B*UN • ORDRSP Rev 15: ORDRSP*D*06A*UN • ORDRSP Rev 16: ORDRSP*D*06B*UN - ORDRSP*D*07B*UN • ORDRSP Rev 17: ORDRSP*D*08A*UN - ORDRSP*D*13B*UN

DESADV

- DESADV Rev ?? : DESADV*D*93A*UN
- DESADV Rev 1: DESADV*D*94A*UN
- DESADV Rev 2: DESADV*D*94B*UN
- DESADV Rev 3: DESADV*D*95A*UN
- DESADV Rev 4: DESADV*D*95B*UN
- DESADV Rev 5: DESADV*D*96A*UN
- DESADV Rev 6: DESADV*D*96B*UN - DESADV*D*97B*UN
- DESADV Rev 7: DESADV*D*98A*UN
- DESADV Rev 8: DESADV*D*98B*UN
- DESADV Rev 9: DESADV*D*99A*UN
- DESADV Rev 10: DESADV*D*99B*UN - DESADV*D*01C*UN
- DESADV Rev 11: DESADV*D*02A*UN
- DESADV Rev 12: DESADV*D*02B*UN - DESADV*D*10B*UN
- DESADV Rev 13: DESADV*D*11A*UN
- DESADV Rev 14: DESADV*D*11B*UN - DESADV*D*13B*UN

6.8 E-Mail

6.8.1 IMAP



IMAP: Empfängt, kopiert/verschiebt oder löscht eine E-Mail-Nachricht in einem IMAP-Postfach (Internet Message Access Protocol), setzt ein Flag oder listet Nachrichten-Header in einem Ordner eines IMAP-Postfachs.

Der Adapter unterstützt mit S/MIME verschlüsselte/signierte Nachrichten sowie die SSL-Verschlüsselung der Verbindung zum IMAP-Server (*IMAPS*). Aus einer Properties-Datei können beliebige *Java Mail Properties* verarbeitet werden.

Ehemalige Bezeichnung bis Version 4.5.1: *IMAP Connection Adapter*

Ehemalige Bezeichnung bis Version 5.5.4: *IMAP Connector*

- ❗ Die Standard-Blockgröße (16 KB) bei der Nachrichtenübertragung kann über die JavaMail-System-Eigenschaft `mail.imap.fetchsize` in der Laufzeitumgebung des X4 Servers (i.d.R. Java-VM mit Applikations-Server) gesetzt werden, um das Herunterladen der E-Mails via IMAP optimiert.

Beispiel für das Startskript von *JBoss*, die Blockgröße wird auf 5 MB gesetzt:

```
@set JAVA_OPTS=%JAVA_OPTS% -Dmail.imap.fetchsize=5242880
```

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> <p><i>Receive Message</i> : Nachricht mit der in Parameter <code>messageID</code> angegeben ID als UTF-8-kodiertes XML-Dokument empfangen</p> <p><i>Receive EML Message</i> : Nachricht mit der in Parameter <code>messageID</code> angegeben ID empfangen und als <code>.eml</code>-Datei (MS Outlook) speichern</p> <p><i>List Messages</i> : Nachrichten-Header eines IMAP-Ordners, der in Parameter <code>folder</code> angegeben wird, in einem UTF-8-kodierten XML-Dokument auflisten</p> <p><i>Copy Message</i> : Nachricht, deren Nachrichten-ID in Parameter <code>messageID</code> angegeben wird, in den IMAP-Ordner kopieren, der in Parameter <code>folder</code> angegeben wird</p> <p><i>Move Message</i> : Nachricht, deren Nachrichten-ID in Parameter <code>messageID</code> angegeben wird, in den IMAP-Ordner verschieben, der in Parameter <code>folder</code> angegeben wird</p> <p><i>Delete Message</i> : Nachricht, deren Nachrichten-ID in Parameter <code>messageID</code> angegeben wird, vom IMAP-Server löschen</p> <p><i>Set Flag</i> : Nachricht, deren Nachrichten-ID in Parameter <code>messageID</code> angegeben wird, mit einem oder mehreren Flags kennzeichnen, die in Parameter <code>flags</code> angegeben werden</p>
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.imap.Adapter</i>: Hauptklasse (Standard)</p>
<i>imapServer</i>	<p>IP-Adresse oder Host-Name des IMAP-Servers</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette (z. B. <i>imap.1und1.de</i>)</p>
<i>login</i>	<p>Benutzername für das verwendete E-Mail-Konto</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>password</i>	<p>Passwort für das verwendete E-Mail-Konto</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>folder</i>	<p>Zielordner auf dem IMAP-Server (wenn die Eigenschaft <i>Operation</i> auf <i>Copy Message</i> oder <i>Move Message</i> gesetzt ist) oder Ordner, dessen Nachrichten aufgelistet werden soll (wenn die Eigenschaft <i>Operation</i> auf <i>List Messages</i> gesetzt ist)</p> <p>Mögliche Werte: Vorhandener Ordnername (Groß-/Kleinschreibung beachten)(z. B. <i>INBOX</i>)</p>
<i>srcFolder</i>	<p>QuellOrdner für die Aktionen <i>Copy Message</i>, <i>Move Message</i>, <i>Receive Message</i>, <i>Set Flag</i> und <i>Delete Message</i>.</p> <p>Mögliche Werte: Vorhandener Ordnername (Groß-/Kleinschreibung beachten)(z. B. <i>INBOX</i>)</p>
<i>flags</i>	<p>IMAP-Flag(s) zur Kennzeichnung einer E-Mail-Nachricht; Mehrere Flags werden durch Semikolon (;) getrennt (wenn die Eigenschaft <i>Operation</i> auf <i>Set Flag</i> gesetzt ist)</p> <p>Mögliche Werte: Beliebiges Flag (z. B. <i>Answered</i>)</p>
<i>messageID</i>	<p>Nachrichten-ID der E-Mail inklusive Ordnerangabe</p> <p>Mögliche Werte: Ordnername und Nachrichten-ID (mit spitzen Klammern)(z. B. <i>INBOX/<831921101.31255094008434.JavaMail.javamailuser@localhost></i>)</p>
<i>uid</i>	<p>Nachrichten-ID auf dem IMAP-Server</p> <p>Mögliche Werte: Beliebige ganze Zahl</p>

<i>maxMessages</i>	<p>Anzahl der aufzulistenden Nachrichten begrenzen (wenn die Eigenschaft Operation auf <i>List Messages</i> gesetzt ist)</p> <p>Mögliche Werte: Beliebige ganze Zahl</p>
<i>base64</i>	<p>Anhänge von Nachrichten (Multipart-Teile) immer Base64-kodieren (wenn die Eigenschaft Operation auf <i>Receive Message</i> gesetzt ist)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Alle Multipart-Teile Base64-kodieren • <i>nein</i>: Nicht Base64-kodieren (Standard)
<i>encrypt</i>	<p>Nachrichten sind mit S/MIME verschlüsselt (<i>Multipart/Encrypted</i>)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: S/MIME-Verschlüsselung aktivieren (erfordert die Angabe zur Schlüssel-/Zertifikatsdatei in Parameter Pkcs12 und der Passphrase in Parameter passphrase) • <i>nein</i>: Keine Verschlüsselung (Standard)
<i>sign</i>	<p>Nachrichten sind mit S/MIME signiert (<i>Multipart/Signed</i>)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: S/MIME-Signatur verarbeiten (erfordert die Angabe zur Schlüssel-/Zertifikatsdatei in Parameter Pkcs12 und der Passphrase in Parameter passphrase) • <i>nein</i>: Keine Signatur (Standard)
<i>Pkcs12</i>	<p>Pfad zur einer gültigen Schlüssel-/Zertifikatsdatei im <i>PKCS#12</i>-Format</p> <p>Mögliche Werte: Beliebige gültige Pfadangabe im lokalen Dateisystem des <i>X4 Servers</i> (z. B. <i>%USER_COLLECTION\Project\Folder\Key.p12</i>)</p>
<i>passphrase</i>	<p>Passwort/Passphrase zur Schlüssel-/Zertifikatsdatei im <i>PKCS#12</i>-Format, die Sie in Parameter Pkcs12 angegeben haben</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette</p>
<i>port</i>	<p>Port des IMAP-Servers</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl (z. B. <i>993</i>) • (<i>kein Wert</i>): Standard-Port <i>143</i> verwenden

<i>ssl</i>	<p>SSL-Verschlüsselung für die Verbindung zum IMAP-Server aktivieren</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Ja</i>: IMAP über SSL (IMAPS) verwenden • <i>Nein</i>: Verbindung nicht verschlüsseln (Standard)
<i>propertiesURL</i>	<p>URL zu einer Properties-Datei (XML oder Text), die beliebige <i>Java Mail Properties</i> enthalten kann (siehe http://javamail.kenai.com/nonav/javadocs/com/sun/mail/imap/package-summary.html)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige gültige URL zu einer Properties-Datei (z. B. <i>xstore://Project/Folder/mail.properties</i>) • <i>(leer)</i>: Keine <i>Java Mail Properties</i> verarbeiten
<i>propertiesType</i>	<p>Format der Properties-Datei, die in Parameter <i>propertiesURL</i> angegeben wurde</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>TEXT</i> : Als Text-Properties interpretieren (Standard) • <i>XML</i> : Als XML-Properties interpretieren

6.8.2 MIME Multipart Converter



MIME Multipart Converter: Ermöglicht das Auslesen und das komfortable Erzeugen von MIME-Multipart-Dokumenten – primär für den Einsatzbereich HTTP-Kommunikation (z. B. ReSTful Webservices). Der Adapter ist zudem für die E-Mail-Kommunikation bzw. für den Umgang mit E-Mail-Anhängen nutzbar.

Der Adapter unterstützt Sie bei der Arbeit mit MIME-Multipart-Dokumenten. Er konvertiert MIME-Dokumente in eine XML-Repräsentation und die XML-Repräsentation in MIME-Dokumente.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>FromBytesWithHeaders</i>: aus einem MIME-Dokument ein MIME-XML-Dokument erzeugen • <i>FromRestMetaXml</i>: aus einem ReST-Input-XML ein entsprechendes MIME-XML-Dokument erzeugen • <i>ModifyRestMetaXml</i>: aus einem ReST-Input-XML ein modifiziertes ReST-Input-XML erzeugen • <i>ToRestMetaXml</i>: aus einem MIME-XML-Dokument ein ReST-Output-XML mit HTTP-Statuscode 200 erzeugen • <i>ToBytesWithHeaders</i>: aus einem MIME-XML-Dokument ein MIME-Dokument erzeugen • <i>FromBytesWithBoundaryFromFirstLine</i>: aus einem MIME-Body ein MIME-XML-Dokument erzeugen • <i>ToBytesWithBoundaryInFirstLine</i>: aus einem MIME-XML-Dokument einen MIME-Body ohne Präambel und ohne Epilog erzeugen • <i>FlattenXml</i>: aus einem MIME-XML-Dokument ein "flaches" MIME-XML-Dokument erzeugen (ggf. wird Multipart-Baum als einzelner Part serialisiert) • <i>PartFromX4Document</i>: Verwandelt ein beliebiges eingehendes X4-Dokument in ein Part in der Mime-XML-Struktur
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.mimemultipart.Adapter</code>: Hauptklasse (Standard)</p>
----------------	--

6.8.2.1 Operation FromBytesWithHeaders

Mit der Adapter-Operation *FromBytesWithHeaders* lassen sich die Bestandteile eines MIME-Dokuments (z.B. eine E-Mail im EML-Format) verarbeiten, z.B. die Anhänge der E-Mail.

Input: Beispiel eines MIME(-Multipart)-Dokuments

```
From: John Doe <example@example.com>
MIME-Version: 1.0
Content-Type: multipart/mixed;
    boundary="XXXXboundary text"

This is a multipart message in MIME format.

--XXXXboundary text
Content-Type: text/plain

this is the body text

--XXXXboundary text
Content-Type: text/plain;
Content-Disposition: attachment;
    filename="test.txt"

this is the attachment text

--XXXXboundary text--
```

Output: Beispiel eines erzeugten MIME(-Multipart)-XML-Dokument

```

<Multipart>
  <Header>
    <Name>from</Name>
    <Value>John Doe &lt;example@example.com&gt;</Value>
  </Header>
  <Header>
    <Name>mime-version</Name>
    <Value>1.0</Value>
  </Header>
  <Header>
    <Name>content-type</Name>
    <Value>multipart/mixed; boundary="XXXXboundary text"</Value>
    <Type>multipart/mixed</Type>
    <Parameter>
      <Name>boundary</Name>
      <Value>XXXXboundary text</Value>
    </Parameter>
  </Header>
  <Prolog>VGhpcyBpcyBhIG11bHRpcGFydCBtZXNzYWdlIGluIE1JTUUgZm9ybWF0Lg0K</Prolog>
  <Part>
    <Header>
      <Name>content-type</Name>
      <Value>text/plain</Value>
      <Type>text/plain</Type>
    </Header>
    <Content encoding="base64">dGhpcyBpcyB0aGUgYm9keSB0ZXh0DQo=</Content>
  </Part>
  <Part>
    <Header>
      <Name>content-type</Name>
      <Value>text/plain;</Value>
      <Type>text/plain</Type>
    </Header>
    <Header>
      <Name>content-disposition</Name>
      <Value>attachment; filename="test.txt"</Value>
      <Disposition>attachment</Disposition>
      <Parameter>
        <Name>filename</Name>
        <Value>test.txt</Value>
      </Parameter>
    </Header>
    <Content encoding="base64">dGhpcyBpcyB0aGUgYXR0YWNobWVudCB0ZXh0DQo=</Content>
  </Part>
  <Epilog />
</Multipart>

```

Wenn das Input-Dokument lediglich ein MIME-Dokument enthält, wird im Output-XML-Dokument nur der entsprechende Part in einem Wurzel-Element <Part/> ausgegeben.

6.8.2.2 Operation FromRestMetaXml

Mit der Adapter-Operation FromRestMetaXml lassen sich XML-Dokumente im *MetaXml*-Format verarbeiten und als MIME-XML ausgeben. Dieses Input-XML-Format wird erzeugt, wenn ein X4-Prozess über den *X4 ReST Starter* aufgerufen wurde und dieser den Input-Handler *HTTP-MetaXml* verwendet.

Einsatzbeispiele:

- Datei-Upload mit Hilfe eines HTML-Formulars: wenn Daten mit dem Content-Type `multipart/form-data` verarbeitet werden sollen.
- Webservices-Kommunikation, bei der sämtliche (Meta-)Daten des HTTP-Requests für die Verarbeitung relevant sind.

Input: Beispiel eines X4 ReST-Input-Dokuments

```

<Request>
  <Path>
    <Method>POST</Method>
    <FullPath>http://localhost:8080/X4/httpstarter/ReST/httpService/simple</
  FullPath>
    <Server>http://localhost:8080</Server>
    <RequestPath>/X4/httpstarter/ReST/httpService/simple</RequestPath>
    <ContextPrefix>/X4/httpstarter/ReST</ContextPrefix>
    <PathInfo>/httpService/simple</PathInfo>
    <Parts>
      <Constant name="httpService" />
      <Constant name="simple" />
    </Parts>
  </Path>
  <Connection>
    <Scheme>http</Scheme>
    <Remote>
      <Address>0:0:0:0:0:0:0:1</Address>
      <Port>51181</Port>
    </Remote>
    <Local>
      <Address>0:0:0:0:0:0:0:1</Address>
      <Port>8080</Port>
    </Local>
    <Server>
      <Name>localhost</Name>
      <Port>8080</Port>
    </Server>
  </Connection>
  <Headers>
    <Header>
      <Name>accept</Name>
      <Value>text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8</
  Value>
    </Header>
    <Header>
      <Name>user-agent</Name>
      <Value>Mozilla/5.0 (Windows NT 6.1; WOW64; rv:43.0) Gecko/20100101
  Firefox/43.0
    </Value>
    </Header>
    <Header>
      <Name>connection</Name>
      <Value>keep-alive</Value>
    </Header>
    <Header>
      <Name>host</Name>
      <Value>localhost:8080</Value>
    </Header>
    <Header>
      <Name>accept-language</Name>
      <Value>de,en-US;q=0.7,en;q=0.3</Value>
    </Header>
  </Headers>

```

```
<Header>  
    <Name>accept-encoding</Name>  
    <Value>gzip, deflate</Value>  
</Header>  
<Header>  
    <Name>content-length</Name>  
    <Value>546</Value>  
</Header>  
<Header>  
    <Name>content-type</Name>  
    <Value>multipart/form-data;  
boundary=-----2109532091963  
    </Value>  
</Header>  
</Headers>  
<Query />  
<Security />  
<Body encoding="base64">  
LS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0yMTA5NTMyMDkxOTYzDQpDb250ZW50LURpc3Bv  
c2l0aW9uOiBmb3JtLWRhdGE7IG5hbWU9IkZlbGQxIg0KDQplcnN0ZXRMgRmVsZA0KLS0tLS0tLS0t  
LS0tLS0tLS0tLS0tLS0tLS0tLS0yMTA5NTMyMDkxOTYzDQpDb250ZW50LURpc3Bvc2l0aW9uOiBm  
b3JtLWRhdGE7IG5hbWU9IkZlbGQxIg0KDQp6d2VpdGVzIEZlbGQNCi0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tMjEwOTUzMjA5MTk2Mw0KQ29udGVudC1EaXNwb3NpdGlvbjoZm9ybS1kYXRh  
OyBuYW1lPSJhcmlldGE7IG5hbWU9IkZlbGQxIg0KDQp6d2VpdGVzIEZlbGQNCi0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLTExMDk1MzIwOTE5NjMNCKNvbnRlbnQtRGZlcG9zaXRpb246IGZvcmt0ZGF0YTsgbmFtZT0iRGF0ZWkiOyBmaWxlbnFtZT0iIG0K  
Q29udGVudC1UeXBloIbhChBSaWNhdGlvi9vY3RldC1zdHJlYW0NCg0KDQotLS0tLS0tLS0tLS0tLS0tLS0tLS0tLS0tLTExMDk1MzIwOTE5NjMtLQ0K  
  
</Body>  
</Request>
```

Output: Beispiel eines erzeugten MIME(-Multipart)-XML-Dokuments

```

<Multipart>
  <Header>
    <Name>accept</Name>
    <Value>text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8</Value>
  </Header>
  <Header>
    <Name>user-agent</Name>
    <Value>Mozilla/5.0 (Windows NT 6.1; WOW64; rv:43.0) Gecko/20100101 Firefox/
43.0</Value>
  </Header>
  <Header>
    <Name>connection</Name>
    <Value>keep-alive</Value>
  </Header>
  <Header>
    <Name>host</Name>
    <Value>localhost:8080</Value>
  </Header>
  <Header>
    <Name>accept-language</Name>
    <Value>de,en-US;q=0.7,en;q=0.3</Value>
  </Header>
  <Header>
    <Name>accept-encoding</Name>
    <Value>gzip, deflate</Value>
  </Header>
  <Header>
    <Name>content-length</Name>
    <Value>546</Value>
  </Header>
  <Header>
    <Name>content-type</Name>
    <Value>multipart/form-data;
boundary=-----2109532091963</Value>
    <Type>multipart/form-data</Type>
    <Parameter>
      <Name>boundary</Name>
      <Value>-----2109532091963</Value>
    </Parameter>
  </Header>
  <Prolog />
  <Part>
    <Header>
      <Name>content-disposition</Name>
      <Value>form-data; name="Feld1"</Value>
      <Disposition>form-data</Disposition>
      <Parameter>
        <Name>name</Name>
        <Value>Feld1</Value>
      </Parameter>
    </Header>
    <Content encoding="base64">ZXJzdGVzIEZlbGQ=</Content>
  </Part>

```

```

<Part>
  <Header>
    <Name>content-disposition</Name>
    <Value>form-data; name="Feld2"</Value>
    <Disposition>form-data</Disposition>
    <Parameter>
      <Name>name</Name>
      <Value>Feld2</Value>
    </Parameter>
  </Header>
  <Content encoding="base64">endlaxRlcyBGZWxk</Content>
</Part>
<Part>
  <Header>
    <Name>content-disposition</Name>
    <Value>form-data; name="area"</Value>
    <Disposition>form-data</Disposition>
    <Parameter>
      <Name>name</Name>
      <Value>area</Value>
    </Parameter>
  </Header>
  <Content encoding="base64">
    RGlIEdyb99lIEZs5GN0ZQ0KDQpibHViDQoNCi0tLS0NCg0Kbw9yZSBibHViDQoNCg==
  </Content>
</Part>
<Part>
  <Header>
    <Name>content-disposition</Name>
    <Value>form-data; name="Datei"; filename=""</Value>
    <Disposition>form-data</Disposition>
    <Parameter>
      <Name>name</Name>
      <Value>Datei</Value>
    </Parameter>
    <Parameter>
      <Name>filename</Name>
      <Value />
    </Parameter>
  </Header>
  <Header>
    <Name>content-type</Name>
    <Value>application/octet-stream</Value>
    <Type>application/octet-stream</Type>
  </Header>
  <Content encoding="base64" />
</Part>
<Epilog />
</Multipart>

```

6.8.2.3 Operation ModifyRestMetaXml

Mit der Adapter-Operation `ModifyRestMetaXml` lassen sich analog zu Operation `FromRestMetaXml` die vom *X4 ReST Starter* erzeugte *MetaXml*-Dokumente verarbeiten, jedoch besitzt das ausgegebene Dokument eine veränderte Struktur:

- Der Header-Bereich (/Request/Headers) wird entfernt.
- Das Body-Element (/Request/Body) wird durch die MIME-XML-Struktur ersetzt.

Input: Rest-Input-XML-Dokument, siehe Input für Operation FromRestMetaXml)

Output: Beispiel eines modifizierten X4 ReST-Input-Dokuments

```

<Request>
  <Path>
    <Method>POST</Method>
    <FullPath>http://localhost:8080/X4/httpstarter/ReST/httpService/simple</
  FullPath>
    <Server>http://localhost:8080</Server>
    <RequestPath>/X4/httpstarter/ReST/httpService/simple</RequestPath>
    <ContextPrefix>/X4/httpstarter/ReST</ContextPrefix>
    <PathInfo>/httpService/simple</PathInfo>
    <Parts>
      <Constant name="httpService" />
      <Constant name="simple" />
    </Parts>
  </Path>
  <Connection>
    <Scheme>http</Scheme>
    <Remote>
      <Address>0:0:0:0:0:0:0:1</Address>
      <Port>51181</Port>
    </Remote>
    <Local>
      <Address>0:0:0:0:0:0:0:1</Address>
      <Port>8080</Port>
    </Local>
    <Server>
      <Name>localhost</Name>
      <Port>8080</Port>
    </Server>
  </Connection>
  <Query />
  <Security />
  <Multipart>
    <Header>
      <Name>accept</Name>
      <Value>text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8</
  Value>
    </Header>
    <Header>
      <Name>user-agent</Name>
      <Value>Mozilla/5.0 (Windows NT 6.1; WOW64; rv:43.0) Gecko/20100101
  Firefox/43.0
    </Value>
    </Header>
    <Header>
      <Name>connection</Name>
      <Value>keep-alive</Value>
    </Header>
    <Header>
      <Name>host</Name>
      <Value>localhost:8080</Value>
    </Header>
    <Header>
      <Name>accept-language</Name>

```

```

        <Value>de,en-US;q=0.7,en;q=0.3</Value>
    </Header>
    <Header>
        <Name>accept-encoding</Name>
        <Value>gzip, deflate</Value>
    </Header>
    <Header>
        <Name>content-length</Name>
        <Value>546</Value>
    </Header>
    <Header>
        <Name>content-type</Name>
        <Value>multipart/form-data;
boundary=-----2109532091963
    </Value>
    <Type>multipart/form-data</Type>
    <Parameter>
        <Name>boundary</Name>
        <Value>-----2109532091963</Value>
    </Parameter>
</Header>
<Prolog />
<Part>
    <Header>
        <Name>content-disposition</Name>
        <Value>form-data; name="Feld1"</Value>
        <Disposition>form-data</Disposition>
        <Parameter>
            <Name>name</Name>
            <Value>Feld1</Value>
        </Parameter>
    </Header>
    <Content encoding="base64">ZXJzdGVzIEZlbGQ= </Content>
</Part>
<Part>
    <Header>
        <Name>content-disposition</Name>
        <Value>form-data; name="Feld2"</Value>
        <Disposition>form-data</Disposition>
        <Parameter>
            <Name>name</Name>
            <Value>Feld2</Value>
        </Parameter>
    </Header>
    <Content encoding="base64">endlaxRlcyBGZWxk</Content>
</Part>
<Part>
    <Header>
        <Name>content-disposition</Name>
        <Value>form-data; name="area"</Value>
        <Disposition>form-data</Disposition>
        <Parameter>
            <Name>name</Name>
            <Value>area</Value>
        </Parameter>
    </Header>
    <Content encoding="base64">

```

```

        RGLlIEdyb99lIEZs5GNoZQ0KDQpibHVlDQoNCi0tLS0NCg0Kbw9yZSBibHVlDQoNCg==
    </Content>
</Part>
<Part>
    <Header>
        <Name>content-disposition</Name>
        <Value>form-data; name="Datei"; filename=""</Value>
        <Disposition>form-data</Disposition>
        <Parameter>
            <Name>name</Name>
            <Value>Datei</Value>
        </Parameter>
        <Parameter>
            <Name>filename</Name>
            <Value />
        </Parameter>
    </Header>
    <Header>
        <Name>content-type</Name>
        <Value>application/octet-stream</Value>
        <Type>application/octet-stream</Type>
    </Header>
    <Content encoding="base64" />
</Part>
<Epilog />
</Multipart>
</Request>

```

6.8.2.4 Operation ToRestMetaXml

Mit der Adapter-Operation `ToRestMetaXml` lässt sich ein MIME-XML-Dokument einlesen und im *MetaXML*-Format erzeugen. Dieses XML-Format wird typischerweise als Ergebnis eines X4-Prozesses ausgegeben, der über den *X4 ReST Starter* aufgerufen wird, wenn dieser den Output-Handler *HTTP-MetaXml* verwendet.

Input: Beispiel eines X4 ReST-Input-Dokuments

```

<Multipart>
  <Header>
    <Name>content-transfer-encoding</Name>
    <Value>binary</Value>
  </Header>
  <Header>
    <Name>content-type</Name>
    <Value>multipart/related; type="application/xop+xml";
      start="&lt;0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d&gt;";
      start-info="text/xml"</Value>
    <Type>multipart/related</Type>
    <Parameter>
      <Name>type</Name>
      <Value>application/xop+xml</Value>
    </Parameter>
    <Parameter>
      <Name>start</Name>
      <Value>&lt;0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d&gt;</Value>
    </Parameter>
    <Parameter>
      <Name>start-info</Name>
      <Value>text/xml</Value>
    </Parameter>
  </Header>
  <Part>
    <Header>
      <Name>content-type</Name>
      <Value>application/xop+xml; charset=UTF-8; type="text/xml"</Value>
      <Type>application/xop+xml</Type>
      <Parameter>
        <Name>charset</Name>
        <Value>UTF-8</Value>
      </Parameter>
      <Parameter>
        <Name>type</Name>
        <Value>text/xml</Value>
      </Parameter>
    </Header>
    <Header>
      <Name>content-transfer-encoding</Name>
      <Value>binary</Value>
    </Header>
    <Header>
      <Name>content-id</Name>
      <Value>&lt;0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d&gt;</Value>
      <Id asUrl="cid:0@4cd748bf-d5ea-4d2f ...</Id>
    </Header>
    <Content encoding="base64">PD94bWwg ...</Content>
  </Part>
</Multipart>

```

Output: Beispiel eines erzeugten X4 ReST-Output-Dokument

```

<Response>
  <Status code="200"/>
  <Headers>
    <Header>
      <Name>content-transfer-encoding</Name><Value>binary</Value>
    </Header>
    <Header>
      <Name>content-type</Name>
      <Value>multipart/related; type="application/xop+xml";
        start="&lt;0@4cd748bf-d5ea-4d2f ... &gt;"; start-info="text/
xml";
        boundary=37aa08a1-cbc3-4143-b565-aae4ff9007d8</Value>
    </Header>
  </Headers>
  <Content encoding="base64">LS0zN2FhM ... </Content>
</Response>

```

6.8.2.5 Operation ToBytesWithHeaders

Mit der Adapter-Operation `ToBytesWithHeaders` lässt sich ein MIME-Dokument mit allen benötigten Header-Informationen erzeugen, d.h. sämtliche Header müssen selbst erzeugt werden.

Anwendungsfall: Erzeugen einer HTML-E-Mail mit eingebetteten Bildern

Input: Beispiel eines MIME-XML-Dokuments

```

<Multipart>
  <Header>
    <Name>content-transfer-encoding</Name>
    <Value>binary</Value>
  </Header>
  <Header>
    <Name>content-type</Name>
    <Value>multipart/related; type="application/xop+xml";
      start="&lt;0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d&gt;";
      start-info="text/xml"</Value>
    <Type>multipart/related</Type>
    <Parameter>
      <Name>type</Name>
      <Value>application/xop+xml</Value>
    </Parameter>
    <Parameter>
      <Name>start</Name>
      <Value>&lt;0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d&gt;</Value>
    </Parameter>
    <Parameter>
      <Name>start-info</Name>
      <Value>text/xml</Value>
    </Parameter>
  </Header>
  <Part>
    <Header>
      <Name>content-type</Name>
      <Value>application/xop+xml; charset=UTF-8; type="text/xml"</Value>
      <Type>application/xop+xml</Type>
      <Parameter>
        <Name>charset</Name>
        <Value>UTF-8</Value>
      </Parameter>
      <Parameter>
        <Name>type</Name>
        <Value>text/xml</Value>
      </Parameter>
    </Header>
    <Header>
      <Name>content-transfer-encoding</Name>
      <Value>binary</Value>
    </Header>
    <Header>
      <Name>content-id</Name>
      <Value>&lt;0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d&gt;</Value>
      <Id asUrl="cid:0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d">
        0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d</Id>
    </Header>
    <Content encoding="base64">PD94bWwgd ... </Content>
  </Part>
</Multipart>

```

Output: Beispiel eines erzeugten MIME-Dokuments

```

content-transfer-encoding: binary
content-type: multipart/related; type="application/xop+xml"; start="<0@4cd748bf-
d5ea-4d2f-89cd-18eb54442e0d>"; start-info="text/xml"; boundary=1e66b8da-5b2c-425e-
b502-7ffdefbe4283

--1e66b8da-5b2c-425e-b502-7ffdefbe4283
content-type: application/xop+xml; charset=UTF-8; type="text/xml"
content-transfer-encoding: binary
content-id: <0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d>

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:stoc="http://example.com/stockquote.xsd">
  <soapenv:Header/>
  <soapenv:Body>
    <blub>SGFsbG8gSGFsbG8g ...</blub>
    <bla>SGFsbG8=</bla>
    <foo>SGFsbG8=</foo>
    <bar>SGFsbG8=</bar>
  </soapenv:Body>
</soapenv:Envelope>
--1e66b8da-5b2c-425e-b502-7ffdefbe4283--

```

6.8.2.6 Operation FromBytesWithBoundaryFromFirstLine

Mit der Adapter-Operation `FromBytesWithBoundaryFromFirstLine` lassen sich analog zu Adapter-Operation `FromBytesWithHeaders` Bestandteile eines MIME-Dokuments (z.B. eine E-Mail im EML-Format) in einem ausgegebenen XML-Dokument verarbeiten, jedoch wird hier kein Header-Block erwartet und es wird direkt der MIME-Body verarbeitet – vorausgesetzt es handelt sich um eine gültige Multipart-Boundary. Ansonsten wird der Input als Part ohne Header im erzeugten XML-Dokument ausgegeben.

Anwendungsfall: SOAP über JMS. Hier wird eine Byte-Message verwendet, die einen MIME-Body ohne MIME-Header enthält. Um MTOM auch hier verwenden zu können, wird ein MIME-Multipart-Body ohne Präambel verwendet.

Input: Beispiel eines MIME-Bodys

```
--XXXXboundary text
Content-Type: text/plain

this is the body text
--XXXXboundary text
Content-Type: text/plain;
Content-Disposition: attachment;
    filename="test.txt"

this is the attachment text
--XXXXboundary text--
```

Output: Beispiel eines erzeugten MIME-XML-Dokuments

```
<Multipart>
  <Prolog />
  <Part>
    <Header>
      <Name>content-type</Name>
      <Value>text/plain</Value>
      <Type>text/plain</Type>
    </Header>
    <Content encoding="base64">dGhpcyBpcyB0aGUgYm9keSB0ZXh0DQo=</Content>
  </Part>
  <Part>
    <Header>
      <Name>content-type</Name>
      <Value>text/plain;</Value>
      <Type>text/plain</Type>
    </Header>
    <Header>
      <Name>content-disposition</Name>
      <Value>attachment; filename="test.txt"</Value>
      <Disposition>attachment</Disposition>
      <Parameter>
        <Name>filename</Name>
        <Value>test.txt</Value>
      </Parameter>
    </Header>
    <Content encoding="base64">dGhpcyBpcyB0aGUgYXR0YWNobWVudCB0ZXh0DQo=</Content>
  </Part>
  <Epilog />
</Multipart>
```

6.8.2.7 Operation ToBytesWithBoundaryInFirstLine

Mit der Adapter-Operation ToBytesWithBoundaryInFirstLine lässt sich ein MIME-Dokument ohne Header-Block und bei Multipart ohne Präambel und ohne Epilog erzeugen.

Anwendungsfall: SOAP über JMS (mit MTOM)

Input: Beispiel eines MIME-XML-Dokuments

```

<Multipart>
  <Header>
    <Name>content-transfer-encoding</Name>
    <Value>binary</Value>
  </Header>
  <Header>
    <Name>content-type</Name>
    <Value>multipart/related; type="application/xop+xml";
      start="&lt;0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d&gt;";
      start-info="text/xml"</Value>
    <Type>multipart/related</Type>
    <Parameter>
      <Name>type</Name>
      <Value>application/xop+xml</Value>
    </Parameter>
    <Parameter>
      <Name>start</Name>
      <Value>&lt;0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d&gt;</Value>
    </Parameter>
    <Parameter>
      <Name>start-info</Name>
      <Value>text/xml</Value>
    </Parameter>
  </Header>
  <Part>
    <Header>
      <Name>content-type</Name>
      <Value>application/xop+xml; charset=UTF-8; type="text/xml"</Value>
      <Type>application/xop+xml</Type>
      <Parameter>
        <Name>charset</Name>
        <Value>UTF-8</Value>
      </Parameter>
      <Parameter>
        <Name>type</Name>
        <Value>text/xml</Value>
      </Parameter>
    </Header>
    <Header>
      <Name>content-transfer-encoding</Name>
      <Value>binary</Value>
    </Header>
    <Header>
      <Name>content-id</Name>
      <Value>&lt;0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d&gt;</Value>
      <Id asUrl="cid:0@4cd748bf-d5ea ..."0@4cd748bf-d5ea ...</Id>
    </Header>
    <Content encoding="base64">PD94bWwgdmVyc2 ...</Content>
  </Part>
</Multipart>

```

Output: Beispiel eines erzeugten MIME-Body-Dokuments

```
--7aed8c79-1491-41a3-9122-c82b15d88fd0
content-type: application/xop+xml; charset=UTF-8; type="text/xml"
content-transfer-encoding: binary
content-id: <0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d>

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:stoc="http://example.com/stockquote.xsd">
  <soapenv:Header/>
  <soapenv:Body>
    <blub>SGFsbG8gSGFsbG8g ... </blub>
    <bla>SGFsbG8=</bla>
    <foo>SGFsbG8=</foo>
    <bar>SGFsbG8=</bar>
  </soapenv:Body>
</soapenv:Envelope>
--7aed8c79-1491-41a3-9122-c82b15d88fd0--
```

6.8.2.8 Operation FlattenXml

Mit der Adapter-Operation `FlattenXml` lässt sich ein MIME-Multipart-XML-Dokument erzeugen, das für den Versand mit einem Managed File Transfer-Adapter der X4 Suite vorbereitet ist, z.B. mit dem *HTTP(S)Adapter*.

Das Ergebnis-XML-Dokument ist hierfür wie folgt aufgebaut:

- Es enthält einen MIME-Body als Base64-kodierte Zeichenkette.
- Die Header sind als XML-Struktur verfügbar, um einen X4 Transfer-Adapter einfach konfigurieren können.
- Das erzeugte Dokument liegt im MIME-XML-Format vom Typ *Part* vor. Sie können es daher verwenden, um weitere Multipart-Dokumente zu erzeugen.

Anwendungsfall: Webservice-Aufrufe, bei denen Multipart-Dokumente benötigt werden

Input: Beispiel eines MIME-XML-Dokuments

```

<Multipart>
  <Header>
    <Name>content-transfer-encoding</Name>
    <Value>binary</Value>
  </Header>
  <Header>
    <Name>content-type</Name>
    <Value>multipart/related; type="application/xop+xml";
      start="&lt;0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d&gt;";
      start-info="text/xml"</Value>
    <Type>multipart/related</Type>
    <Parameter>
      <Name>type</Name>
      <Value>application/xop+xml</Value>
    </Parameter>
    <Parameter>
      <Name>start</Name>
      <Value>&lt;0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d&gt;</Value>
    </Parameter>
    <Parameter>
      <Name>start-info</Name>
      <Value>text/xml</Value>
    </Parameter>
  </Header>
  <Part>
    <Header>
      <Name>content-type</Name>
      <Value>application/xop+xml; charset=UTF-8; type="text/xml"</Value>
      <Type>application/xop+xml</Type>
      <Parameter>
        <Name>charset</Name>
        <Value>UTF-8</Value>
      </Parameter>
      <Parameter>
        <Name>type</Name>
        <Value>text/xml</Value>
      </Parameter>
    </Header>
    <Header>
      <Name>content-transfer-encoding</Name>
      <Value>binary</Value>
    </Header>
    <Header>
      <Name>content-id</Name>
      <Value>&lt;0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d&gt;</Value>
      <Id asUrl="cid:0@4cd748bf-d5ea ...">0@4cd748bf-d5ea ...</Id>
    </Header>
    <Content encoding="base64">PD94bWwgdmVyc2 ...</Content>
  </Part>
</Multipart>

```

Output: Beispiel eines erzeugten MIME-XML-Dokuments

```

<Part>
  <Header>
    <Name>content-transfer-encoding</Name>
    <Value>binary</Value>
  </Header>
  <Header>
    <Name>content-type</Name>
    <Value>multipart/related; type="application/xop+xml";
      start="&lt;0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d&gt;";
      start-info="text/xml"; boundary=da9a6ade-4452-4546-851c-84da7b638c24</Value>
    <Type>multipart/related</Type>
    <Parameter>
      <Name>type</Name>
      <Value>application/xop+xml</Value>
    </Parameter>
    <Parameter>
      <Name>start</Name>
      <Value>&lt;0@4cd748bf-d5ea-4d2f-89cd-18eb54442e0d&gt;</Value>
    </Parameter>
    <Parameter>
      <Name>start-info</Name>
      <Value>text/xml</Value>
    </Parameter>
    <Parameter>
      <Name>boundary</Name>
      <Value>da9a6ade-4452-4546-851c-84da7b638c24</Value>
    </Parameter>
  </Header>
  <Content encoding="base64">LS1kYTlhNmFkZ ...</Content>
</Part>

```

6.8.2.9 Operation PartFromX4Document

Mit der Operation `PartFromX4Document` wird der Adapter-Input als Rumpf eines Mime-Dokuments konvertiert (im XML-Format dieses Adapters). Es wird ein Content-Type Header erzeugt mit den entsprechenden Werten aus dem X4-Dokument (diese Werte lassen sich mit dem [Document Properties Changer](#) beeinflussen).

X4Dokument-Eigenschaften umwandeln in den Content-Type Header

X4Document <i>mimeType</i>	X4Document <i>encoding</i>	resultierender Content-Type Header
leer	leer	kein Header
leer	someCharset	kein Header
some/type	leer	Content-Type: some/type
some/type	someCharset	Content-Type: some/type; charset=someCharset

Diese Operation ist vergleichbar mit folgender Adapterfolge:

1. Base64 (encode) – Input als Base64-String verpacken

2. ElementWrapper (wrap) – Base64-String in ein XML-Dokument verpacken
3. XSL – XML-Dokument in die passende Form bringen (und die Header erzeugen)

6.8.3 MSG Converter



MSG Converter: Liest ein *Microsoft Exchange Mail Document* (.msg) in beliebiger Zeichenkodierung, das z. B. von *Microsoft Outlook* erzeugt wurde, und gibt ein XML-Dokument mit den Sender- und Empfängerdaten, dem Betreff sowie den *Base64*-kodierte Inhalten des E-Mail-Bodys und von ggf. vorhandenen Anhängen aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt
	Mögliche Werte: <i>Convert</i> : MSG-Dokument in XML konvertieren

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!)
	Mögliche Werte: <i>de.softproject.integration.adapter.msg.MSGConverter</i> : Hauptklasse (Standard)

Ausgabe

Der Adapter gibt ein XML-Dokument nach folgendem Muster aus, in dem ggf. innerhalb von einem Element <attachments> sämtliche Anhänge in jeweils einem Element <attachment> enthalten sind:

```
<mail>
  <from><!--Absendername--><email><!--E-Mail-Adresse des Absenders--></email></from>
  <to><!--Empfängername--><email><!--E-Mail-Adresse des Empfängers--></email></to>
  <subject><!--Betreff--></subject>
  <body><!--Base64-kodierter E-Mail Body--></body>
  <bodyRTF><!--Base64-kodierter E-Mail Body (RTF)--></bodyRTF>
  <attachments>
    <attachment filename="Dateiname" mimetype="MIME-Typ"
      size="Dateigröße in Bytes"><!--Base64-kodierter Anhang--></attachment>
  </attachments>
</mail>
```

6.8.4 POP/SMTP

POP/SMTP: Sendet und empfängt E-Mail-Nachrichten über SMTP (Simple Mails Transfer Protocol) bzw. POP3 (Post Office Protocol v3).





Der Adapter unterstützt Dateianhänge und MIME-Typ-Angaben für den E-Mail-Body u.a. zum Versenden von HTML-E-Mails. Zudem wird das Senden und Empfangen von S/MIME-verschlüsselten und/oder -signierten E-Mail-Nachrichten unterstützt.

Eigenschaften (Properties)

I/O Mode (input_type)	<p>Art der Quelle der zu verarbeitenden Daten</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Actions</i>: Daten aus dem vorangegangenen Prozess-Schritt verarbeiten • <i>Directory</i>: Alle Dateien eines angegebenen Ordners als Anhang verarbeiten (oder Dateien gemäß Filterregel in Parameter filter)
File name (documentname)	<p>Name des zu sendenden Dateianhangs (wenn I/O Mode auf <i>Actions</i> gesetzt ist).</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p>ⓘ Wenn diese Eigenschaft dynamisch parametrisiert werden soll, muss im zugehörigen XSL-Mapping der technische Name verwendet werden. Dieser Name kann abweichen und ist in Klammern angegeben.</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>Beispiel</p> <pre><xsl:processing-instruction name="DynamicParameter"> documentname=" <xsl:value-of select="."/> " </xsl:processing-instruction></pre> </div> </div> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebiger Dateiname (z.B. <i>Data.xml</i>) • Mehrere Dateinamen, durch Semikolon getrennt (wenn Sie dem Transfer-Adapter über Fork/Join mehrere Input-Dokumente übergeben, dann können auch mehrere Dateien gemäß der Priorität der Verzweigungen nacheinander verarbeitet werden (z.B. <i>File1.xml;File2.xml</i>))
Delete (delete)	E-Mail nach erfolgreichem Empfang vom POP3-Server löschen (wenn Eigenschaft Operation auf <i>Receive</i> gesetzt ist)
Operation (action)	<p>Bestimmt, welche Operation mit den verarbeiteten Dateien durchgeführt wird. Je nach Wert in I/O Mode unterscheidet sich die Funktionsweise des Adapters.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Send</i>: E-Mail über den SMTP-Server senden • <i>Receive</i>: E-Mail vom POP3-Server empfangen

Parameter

from	<p>E-Mail-Adresse des Absenders; Bei Bedarf mit Angabe einer Anzeigenamens</p> <p>Mögliche Werte: Beliebige gültige E-Mail-Adresse (z.B. <i>name@domain.tld</i> (<i>SampleName</i>))</p>
sender	<p>Technischer Absender (falls abweichend von from), der die Nachricht verschickt; Bei Bedarf mit Angabe einer Anzeigenamens</p> <p>Mögliche Werte: Beliebige gültige E-Mail-Adresse (z.B. <i>name@domain.tld</i> (<i>SampleName</i>))</p>
to	<p>E-Mail-Adresse(n), an die die E-Mail gesendet werden soll; Bei Bedarf mit Angabe einer Anzeigenamens</p> <p>Mögliche Werte: Beliebige gültige E-Mail-Adresse, bei mehreren Adressen durch Semikolon getrennt (z.B. <i>name@domain.tld;email@host.tld</i> (<i>SampleName</i>))</p>
cc	<p>E-Mail-Adresse(n), an die die E-Mail als Kopie gesendet werden soll; Bei Bedarf mit Angabe einer Anzeigenamens</p> <p>Mögliche Werte: Beliebige gültige E-Mail-Adresse, bei mehreren Adressen durch Semikolon getrennt (z.B. <i>name@domain.tld</i> (<i>SampleName</i>);<i>email@host.tld</i>)</p>
bcc	<p>E-Mail-Adresse(n), an die die E-Mail als Blindkopie gesendet werden soll; Bei Bedarf mit Angabe einer Anzeigenamens</p> <p>Mögliche Werte: Beliebige gültige E-Mail-Adresse, bei mehreren Adressen durch Semikolon getrennt (z.B. <i>name@domain.tld;email@host.tld</i> (<i>SampleName</i>))</p>
replyTo	<p>E-Mail-Adresse, an die eine Antwort verschickt werden soll</p> <p>Mögliche Werte: Beliebige gültige E-Mail-Adresse (z.B. <i>name@domain.tld</i>)</p>
subject	Betreff der E-Mail
body	<p>Inhalt der E-Mail (wenn die Input-Daten des Adapters ein HTML-Dokument ist, so wird dieses als HTML-Nachrichten-Body verwendet)</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
contentTypeBody	<p>MIME-Typ des E-Mail-Inhalts (ermöglicht die Darstellung von HTML-E-Mails oder mehreren Anhängen)</p> <p>Mögliche Werte: Gültige MIME-Typen (z.B. <i>multipart/mixed</i>)</p>
charset	<p>Zu verwendender Zeichensatz des E-Mail-Inhalts</p> <p>Mögliche Werte: Gültige Zeichensatz-Angabe (z.B. <i>ISO-8859-15</i>)</p>

smtp	IP-Adresse oder Host-Name des SMTP-Servers zum Senden von E-Mails Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>smtp.domain.tld</i>)
smtpPort	Port des SMTP-Servers zum Senden von E-Mails Mögliche Werte: Beliebige positive ganze Zahl, z.B. <i>250</i>
smtpSecurityProtocol	Zu verwendendes SMTP-Sicherheitsprotokoll Mögliche Werte: <ul style="list-style-type: none"> • NONE: Kein Protokoll verwenden • TLS: TLS als Sicherheitsprotokoll verwenden • SSL: SSL als Sicherheitsprotokoll verwenden
smtpAllowAllHosts	Alle Hosts für SMTP erlauben <div>  Bitte beachten Sie, dass auch Hosts erlaubt werden, die kein gültiges Zertifikat haben. </div> <div>  Dieser Parameter ist nur gültig, wenn SSL als Sicherheitsprotokoll verwendet wird. </div>
pop3	IP-Adresse oder Host-Name des POP3-Servers zum Empfangen von E-Mails Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>pop3.domain.tld</i>)
pop3Port	Port des POP3-Servers zum Empfangen von E-Mails Mögliche Werte: Beliebige positive ganze Zahl
pop3SecurityProtocol	Zu verwendendes POP3-Sicherheitsprotokoll Mögliche Werte: <ul style="list-style-type: none"> • NONE: Kein Protokoll verwenden • TLS: TLS als Sicherheitsprotokoll verwenden • SSL: SSL als Sicherheitsprotokoll verwenden
pop3AllowAllHosts	Alle Hosts für POP3 erlauben <div>  Bitte beachten Sie, dass auch Hosts erlaubt werden, die kein gültiges Zertifikat haben. </div> <div>  Dieser Parameter ist nur gültig, wenn SSL als Sicherheitsprotokoll verwendet wird. </div>

login	<p>Benutzername für den POP3-Server (beim Empfangen einer E-Mail) oder für den SMTP-Server (beim Senden einer E-Mail)</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <i>name@domain.tld</i>)</p>
password	<p>Passwort für den POP3-Server (beim Empfangen einer E-Mail) oder für den SMTP-Server (beim Senden einer E-Mail)</p>
file_limit	<p>Anzahl der zu sendenden Dateianhänge begrenzen (wenn I/O Mode auf <i>Directory</i> gesetzt ist und somit mehrere Dateien eines Ordners angehängt werden sollen)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 0: Keine Dateien anhängen (Standard)
dir	<p>Lokaler Ordner auf dem X4 Server, z.B. im X4 Repository, in dem Dateien liegen, die als Dateianhänge per E-Mail gesendet werden sollen (wenn I/O Mode auf <i>Directory</i> gesetzt ist)</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <i>%USER_COLLECTION%\Folder</i>)</p>
filter	<p>Zu versendenden Datei-Anhänge nach Dateinamen filtern über die Platzhalter * und ?</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige Zeichenkette, so gilt beispielsweise <i>*.dat</i> nur für Dateien mit der Dateiendung <i>*.dat</i> • *: Alle Dateien verwenden (Standard)
EML	<p>Nachricht im EML-Format verschicken (dies erfordert eine EML-Nachricht als Adapter-Input) oder Nachricht im EML-Format empfangen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Im EML-Format verschicken • <i>nein</i>: Standard-E-Mail-Format verwenden (Standard)
base64	<p>Anhänge von Nachrichten (Multipart-Teile) immer Base64-kodieren (wenn die Eigenschaft <i>Operation</i> auf <i>Receive</i> gesetzt ist)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Alle Multipart-Teile Base64-kodieren • <i>nein</i>: Nicht Base64-kodieren (Standard)

encrypt	<p>Nachricht mit S/MIME verschlüsseln (<i>Multipart/Encrypted</i>)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: S/MIME-Verschlüsselung aktivieren (erfordert die Angabe zur Schlüssel-/Zertifikatsdatei in Parameter Pkcs12 und der Passphrase in Parameter passphrase) • <i>nein</i>: Keine Verschlüsselung (Standard)
sign	<p>Nachricht mit S/MIME signieren (<i>Multipart/Signed</i>)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: S/MIME-Signatur hinzufügen (erfordert die Angabe zur Schlüssel-/Zertifikatsdatei in Parameter Pkcs12 und der Passphrase in Parameter passphrase) • <i>nein</i>: Keine Signatur (Standard)
Pkcs12	<p>Pfad zur einer gültigen Schlüssel-/Zertifikatsdatei im <i>PKCS#12</i>-Format</p> <p>Mögliche Werte: Beliebige gültige Pfadangabe im lokalen Dateisystem des <i>X4 Servers</i> (z.B. <i>%USER_COLLECTION\Project\Folder\Key.p12</i>)</p>
passphrase	<p>Passwort/Passphrase zur Schlüssel-/Zertifikatsdatei im <i>PKCS#12</i>-Format, die Sie in Parameter Pkcs12 angegeben haben</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette</p>
propertiesType	<p>Datenformat für eine Properties-Datei, in der weitere <i>JavaMail</i>-Properties hinterlegt werden können</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>text</i>: Properties-Datei in Textformat • <i>xml</i>: Properties-Datei in XML-Format, siehe http://www.ibm.com/developerworks/library/j-tiger02254/index.html

propertiesUrl	<p>⚠ Veralteter Parameter</p> <p>Ab Version 6.2.0 der X4 Suite kann das zu verwendende Sicherheitsprotokoll über die Parameter <code>smtpSecurityProtocol</code> und <code>pop3SecurityProtocol</code> gesetzt werden.</p> <p>Beim Verwenden des nun veralteten Parameters <code>propertiesURL</code> wird eine Warnung im Server-Log ausgegeben.</p> <p>URL zu einer Properties-Datei, in der weitere <i>JavaMail</i>-Properties hinterlegt werden können, siehe https://javamail.java.net/nonav/docs/api/com/sun/mail/smtp/package-summary.html. Die verknüpfte Datei muss im Format vorliegen, das im Parameter <code>propertiesType</code> angegeben wurde.</p> <p>i Wird die <code>propertiesURL</code> gesetzt, werden die Parameter aus der Properties-Datei verwendet. Andernfalls werden die Adapter-Parameter verwendet.</p> <p>Mögliche Werte: Beliebige gültige URL, z.B. <code>xstore://Project/Folder/File.properties</code></p>
---------------	---

i Welche E-Mail-Adressen formal gültig sind, kann den Richtlinien *RFC 5321* und *RFC 5322* entnommen werden.

6.9 ERP

6.9.1 Abas ERP Connector



Abas ERP Connector: Stellt eine Verbindung zum Abas ERP-System über dessen *jedp*-Schnittstelle her und führt verschiedene Operationen aus.

Eigenschaften (Properties)

Operation	<p>Bestimmt, welche Operation der Connector durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> Query: Datensätze und Tabelleninhalte abfragen Info: Metadaten von abas abfragen Write: Datensätze und Tabelleninhalte hinzufügen oder ändern
-----------	--

Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.abas.AbasAdapter</code>: Hauptklasse (Standard)</p>
---------	---

<i>appName</i>	Name der Applikation Mögliche Werte: Beliebige Zeichenkette
<i>edpPort</i>	EDP-Port Mögliche Werte: Beliebige ganze Zahl, z.B. 6550
<i>mandant</i>	Name des Mandanten Mögliche Werte: Beliebige Zeichenkette, z.B. <i>systemtest</i>
<i>host</i>	IP-Adresse oder Host-Name des Servers Mögliche Werte: Beliebige gültige Zeichenkette ohne Protokollangabe, z.B. <i>example.com</i>
<i>password</i>	Passwort für die Authentifizierung Mögliche Werte: Beliebige Zeichenkette
<i>edpVersion</i>	EDP-Version Mögliche Werte: Beliebige ganze Zahl, z.B. 3.35(Standard)
<i>wait</i>	Lock-Verhalten der Session Mögliche Werte: <ul style="list-style-type: none"> • WAITINFINITE: Warten, bis Lock freigegeben ist • WAIT: Warten, bis Lock freigegeben oder die Zeitbeschränkung erreicht ist • NOWAIT: Fehler ohne zu warten sofort zurückgeben (Standard)
<i>timeout</i>	Zeitbeschränkung für den Verbindungsaufbau in Sekunden Mögliche Werte: Beliebige ganze Zahl, z.B. 30(standard)

Statuswerte

1 (OK)	Die Ausführung war erfolgreich
-1 (ERROR)	Bei der Ausführung kam es zu einem Fehler, z.B. wegen eines ungültigen XML-Inputs
0 (TIMEOUT)	Bei der Abfrage des abas ERP-Systems kam es zu einer Zeitüberschreitung

Input

Für die verschiedenen Operationen erwartet der Adapter folgende XML-Input-Strukturen:

Operation: Query

```
<Abas>
  <!-- Query ausführen -->
  <Query resulttag="...">
    <DatabaseGroup>00:01</DatabaseGroup>
    <Fields>
      <_ans/>
      <_str/>
      <_nummer/>
    </Fields>
    <Selection>nummer=20001!20002</Selection>
  </Query>
  <Query resulttag="...">
    <DatabaseGroup>01:01</DatabaseGroup>
    <Fields>
      <_schlag/>
      <_yvertragsnr/>
      <_kunde/>
    </Fields>
    <Selection>kunde=20001!20002</Selection>
  </Query>
</Abas>
```

Operation: Info

```
<Abas>
  <!-- Aktion ausführen -->
  <Info resulttag="...">
    <Nummer>1000</Nummer>
    <HeaderFields>
      <_bstart>1</_bstart>
    </HeaderFields>
    <TableFields>
      <_tysuch/>
      <_tyname/>
    </TableFields>
  </Info>
  <Info resulttag="...">
    <Nummer>1000</Nummer>
    <HeaderFields>
      <_bstart>1</_bstart>
    </HeaderFields>
    <TableFields>
      <_tysuch/>
    </TableFields>
  </Info>
</Abas>
```

Operation: Write

```

<Abas>
  <!-- Bestehenden Datensatz ändern -->
  <Update resulttag="...">
    <Database>00</Database>
    <Group>01</Group>
    <Nummer>20001</Nummer>
    <HeaderFields>
      <_str>...</_str>
    </HeaderFields>
  </Update>
  <!-- Neuer Datensatz -->
  <Insert resulttag="...">
    <Database>00</Database>
    <Group>01</Group>
    <HeaderFields>
      <_nummer>200171</_nummer>
      <_such>Testsuch</_such>
      <_ans2>Testans</_ans2>
      <_str2>Testgasse</_str2>
      <_plz2>12345</_plz2>
      <_nort2>Testhausen</_nort2>
      <_zbed>100</_zbed>
    </HeaderFields>
  </Insert>
  <!-- Zeile in bestehendem Datensatz ändern -->
  <Update resulttag="...">
    <Database>71</Database>
    <Group>02</Group>
    <Nummer>4</Nummer>
    <Table>
      <Row nr="2">
        <TableFields>
          <_ytg02beschreibung>blau</_ytg02beschreibung>
        </TableFields>
      </Row>
    </Table>
  </Update>
  <!-- Neue Zeile in bestehendem Datensatz -->
  <Update resulttag="...">
    <Database>71</Database>
    <Group>02</Group>
    <Nummer>4</Nummer>
    <Table>
      <Row nr="2" new="true">
        <TableFields>
          <_ytg02beschreibung>gelb</_ytg02beschreibung>
        </TableFields>
      </Row>
    </Table>
  </Update>
</Abas>

```

Ausgabe

Für die verschiedenen Operationen gibt der Adapter folgende XML-Ergebnisdokumente aus:

Operation: Query

```
<Result>
  <Adresse>
    <_ans>Max Mustermann</_ans>
    <_str>Musterallee 2</_str>
    <_nummer>20001</_nummer>
  </Adresse>
  <Adresse>
    <_ans>Manfred Mustermann</_ans>
    <_str>Testweg 1</_str>
    <_nummer>20002</_nummer>
  </Adresse>
  <Vertrag>
    <_schlag>RECHNUNG.2017.10</_schlag>
    <_yvertragsnr>01-000001-001</_yvertragsnr>
    <_kunde>123</_kunde>
  </Vertrag>
  <!-- ggf. weitere Knoten -->
</Result>
```

Operation: Info

```
<Result>
  <Test>
    <_tysuch>XY123</_tysuch>
    <_tyname>Test Abas-Server</_tyname>
  </Test>
  <Test2>
    <_tysuch>XY123</_tysuch>
  </Test2>
</Result>
```

Operation Write

```
<Result>
  <Testupdate1>updated</Testupdate1>
  <Testinsert1>inserted</Testinsert1>
  <Testupdate2>updated</Testupdate2>
  <Testupdate3>updated</Testupdate3>
</Result>
```

Abhängigkeiten

Der Connector verwendet die Bibliothek *jedp* von abas, jedp-3.3.5.jar.

- ❗ Die aktuell unterstützte Version ist 3.3.5.
Die Bibliothek wird aus lizenzrechtlichen Gründen nicht ausgeliefert. Zur Verwendung dieser Bibliothek im X4 Server-Installationsordner in Ordner
wildfly\modules\system\layers\base\de\softproject\x4\extensions\main\
kopieren, gemeinsam mit einer Datei `module.xml` mit folgendem Inhalt:

module.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<module xmlns="urn:jboss:module:1.1" name="de.softproject.x4.extensions">
  <resources>
    <resource-root path="jedp-3.3.5.jar" />
  </resources>
</module>
```

6.9.2 SAP Connector

6.9.2.1 IDoc-Strukturen

Wie IDocs im Raw-Format und XML-IDocs aufgebaut sind

6.9.2.1.1 IDoc im Raw-Format

Ein *IDoc Raw*-Dokument ist ein XML-Dokument, das neben Metadaten zur Nachricht und Strukturinformationen ein SAP *IDoc* im Rohformat als Text enthält. Das XML-Dokument ist nach folgendem Muster aufgebaut:

```

<X4SAP TID="<!--Transaktions-ID-->">
  <X4IDoc type="<!--IDoc-Typ-->" description="">
    <X4Mandatory>
      <X4RecipientPort><!--Empfänger-Port--></X4RecipientPort>
      <X4RecipientPartnerType><!--Partnertyp-Kürzel--></X4RecipientPartnerType>
      <X4RecipientPartnerNumber><!--Partnernummer--></X4RecipientPartnerNumber>
      <X4RecipientPartnerFunction />
      <X4RecipientAdress />
      <X4RecipientLogicalAddress />
      <X4SenderPort><!--Sender-Port--></X4SenderPort>
      <X4SenderPartnerType><!--Partnertyp-Kürzel--></X4SenderPartnerType>
      <X4SenderPartnerNumber><!--Partnernummer--></X4SenderPartnerNumber>
      <X4SenderPartnerFunction />
      <X4SenderAddress />
      <X4SenderLogicalAddress />
      <MessageType>
        <MSGTYPE><!--Nachrichtentyp--></MSGTYPE>
      </MessageType>
    </X4Mandatory>
    <X4Structure>
      <Tabellenname SEGMENT="1">
        <!--Segment 1 (Steuersatz)-->
      </Tabellenname>
    </X4Structure>
    <RAW>
      <!--Rohdaten der IDoc-Nachricht (Text)-->
    </RAW>
  </X4IDoc>
</X4SAP>

```

6.9.2.1.2 IDoc im XML-Format

Ein IDoc-XML-Dokument enthält eine IDoc-Nachricht, die komplett in XML-Elemente umgewandelt wurde. Zudem enthält das XML-Dokument Metadaten zur Nachricht und Strukturinformationen. Innerhalb des Elements <IDOC> werden sämtliche Tabellen hintereinander hinterlegt.

Ein IDoc im XML-Format ist nach folgendem Muster aufgebaut:

```

<X4SAP TID="<!--Transaktions-ID-->">
  <X4IDoc type="<!--IDoc-Typ-->" description="">
    <X4Mandatory>
      <X4RecipientPort><!--Empfänger-Port--></X4RecipientPort>
      <X4RecipientPartnerType><!--Partnertyp-Kürzel--></X4RecipientPartnerType>
      <X4RecipientPartnerNumber><!--Partnernummer--></X4RecipientPartnerNumber>
      <X4RecipientPartnerFunction />
      <X4RecipientAdress />
      <X4RecipientLogicalAddress />
      <X4SenderPort><!--Sender-Port--></X4SenderPort>
      <X4SenderPartnerType><!--Partnertyp-Kürzel--></X4SenderPartnerType>
      <X4SenderPartnerNumber><!--Partnernummer--></X4SenderPartnerNumber>
      <X4SenderPartnerFunction />
      <X4SenderAddress />
      <X4SenderLogicalAddress />
      <MessageType>
        <MSGTYPE><!--Nachrichtentyp--></MSGTYPE>
      </MessageType>
    </X4Mandatory>
    <X4Structure>
      <Tabellenname SEGMENT="1">
        <!--Segment 1 (Steuersatz)-->
      </Tabellenname>
    </X4Structure>
    <RAW>
      <!--Rohdaten der IDoc-Nachricht (Text)-->
    </RAW>
  </X4IDoc>
</X4SAP>

```

6.9.2.2 SAP Connector (JCo 2)



SAP Connector (JCo 2): Ermöglicht die Übermittlung von Daten vom X4 ESB an ein *SAP R/3*-System. Er ruft einen beliebigen Remote Function Call (RFC) bzw. einen beliebigen BAPI-Funktionsbaustein in einem SAP-System auf. Zudem können IDocs an *SAP R/3* versandt werden.

Ehemalige Bezeichnung bis Version 4.5.1: *SAP (JCo 2) Request Adapter*

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>Execute</i> : Funktions-Adapter ausführen
------------------	--


Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.x4.sapadapter.SAPAdapter</i> : Hauptklasse (Standard)
<i>x4.connectionPoolEnabled</i>	Connection Pooling verwenden
<i>x4.connectionPoolUID</i>	ID des Connection Pools. Der SAP-Adapter kann einen bereits existierenden Connection Pool verwenden, wenn dieselben Verbindungsparameter und dieselbe ID des Connection Pools gesetzt wurden.
<i>x4.connectionPoolMax</i>	Maximale Anzahl definierter Verbindungen im Connection Pool
<i>x4.transactionEnabled</i>	Transaktionale Verarbeitung aktivieren oder deaktivieren.
<i>x4.transactionID</i>	Transaktions-ID; alle Operationen des SAP-Adapters innerhalb eines Prozesses werden als eine Transaktion ausgeführt und besitzen dieselbe Transaktions-ID. Der SAP-Adapter kann eine bereits gestartete Transaktion verwenden, wenn sie dieselben Transaktionsparameter und dieselbe Transaktions-ID besitzen.
<i>jco.client.client</i>	SAP-spezifischer Parameter: SAP-Mandant
<i>jco.client.user</i>	SAP-spezifischer Parameter: Benutzername
<i>jco.client.passwd</i>	SAP-spezifischer Parameter: Passwort
<i>jco.client.lang</i>	SAP-spezifischer Parameter: Anmeldesprache
<i>jco.client.sysnr</i>	SAP-spezifischer Parameter: SAP-System-Nummer
<i>jco.client.ashost</i>	SAP-spezifischer Parameter: SAP-Applikations-Server
<i>jco.client.mshost</i>	SAP-spezifischer Parameter: SAP-Nachrichten-Server
<i>jco.client.gwhost</i>	SAP-spezifischer Parameter: Gateway Host
<i>jco.client.gwserv</i>	SAP-spezifischer Parameter: Gateway Service
<i>jco.client.r3name</i>	SAP-spezifischer Parameter: R/3-Name
<i>jco.client.group</i>	SAP-spezifischer Parameter: Gruppe der SAP-Applikations-Server

<i>jco.client.tpname</i>	SAP-spezifischer Parameter: Programm-ID einer externen Server-Anwendung
<i>jco.client.tphost</i>	SAP-spezifischer Parameter: Host der externen Server-Anwendung
<i>jco.client.type</i>	<p>SAP-spezifischer Parameter: Art des entfernten Hosts</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • 2: R/2 • 3: R/3 • E: Extern
<i>jco.client.trace</i>	<p>SAP-spezifischer Parameter: RFC-Trace</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • 0: RFC-Trace deaktivieren • 1: RFC-Trace aktivieren
<i>jco.client.codepage</i>	SAP-spezifischer Parameter: Anfängliche Codepage in SAP-Notation
<i>jco.client.abap_debug</i>	<p>SAP-spezifischer Parameter: ABAP-Debugging</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • 0: ABAP-Debugging deaktivieren • 1: ABAP-Debugging aktivieren
<i>jco.client.use_sapgui</i>	SAP-spezifischer Parameter: Name einer grafischen SAP-Benutzeroberfläche, die verwendet werden soll
<i>jco.client.getsso2</i>	<p>SAP-spezifischer Parameter: SSO-Ticket nach der Anmeldung erhalten</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • 0: Kein SSO-Ticket erhalten • 1: SSO-Ticket nach der Anmeldung erhalten
<i>jco.client.mysapso2</i>	SAP-spezifischer Parameter: Name des SAP-Cookies Version 2, der als Anmeldungs-Ticket verwendet werden soll
<i>jco.client.x509cert</i>	SAP-spezifischer Parameter: Name des X509-Zertifikats, das als Anmeldungs-Ticket verwendet werden soll

<i>jco.client.lcheck</i>	<p>SAP-spezifischer Parameter: Anmelde-Überprüfung beim Öffnen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • 0: Anmelde-Überprüfung deaktivieren • 1: Anmelde-Überprüfung beim Öffnen aktivieren
<i>jco.client.grt_data</i>	SAP-spezifischer Parameter: Zusätzliche Daten für die grafische SAP-Benutzeroberfläche
<i>jco.client.use_guihost</i>	SAP-spezifischer Parameter: Host, zu dem die entfernte grafische SAP-Benutzeroberfläche weitergeleitet werden soll
<i>jco.client.use_guiserv</i>	SAP-spezifischer Parameter: Dienst, zu dem die entfernte grafische SAP-Benutzeroberfläche weitergeleitet werden soll
<i>jco.client.use_guiprogid</i>	SAP-spezifischer Parameter: Programm-ID des Servers, der die entfernte grafische SAP-Benutzeroberfläche startet
<i>jco.client.snc_mode</i>	<p>SAP-spezifischer Parameter: <i>Secure network connection</i> (SNC)-Modus</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • 0: SNC-Modus deaktivieren • 1: SNC-Modus aktivieren
<i>jco.client.snc_partnername</i>	SAP-spezifischer Parameter: SNC-Partner, z. B. <i>p:CN=R3, O=XYZ-INC, C=EN</i>
<i>jco.client.snc_qop</i>	<p>SAP-spezifischer Parameter: SNC-Sicherheits-Level</p> <p>Mögliche Werte:</p> <p>1...9: SNC-Sicherheits-Level</p>
<i>jco.client.snc_myname</i>	SAP-spezifischer Parameter: SNC-Name (überschreibt den Standard-SNC-Partner)
<i>jco.client.snc_lib</i>	SAP-spezifischer Parameter: Pfad zu einer Bibliothek, die den SNC-Dienst bereitstellt
<i>jco.client.dest</i>	SAP-spezifischer Parameter: R/2-Ziel (destination)
<i>jco.client.saplogon_id</i>	SAP-spezifischer Parameter
<i>jco.client.extiddata</i>	SAP-spezifischer Parameter: Daten zur externen Authentifizierung (PAS)

<i>jco.client.extidtype</i>	SAP-spezifischer Parameter: Art der externen Authentifizierung (PAS)
-----------------------------	--

 Eine vollständige Beschreibung aller (Verbindungs-) Parameter finden Sie in der *SAP JCo* 2-Dokumentation.

6.9.2.2.1 RFC/BAPI aufrufen

Input

Der Adapter erwartet ein Input-XML-Dokument nach folgendem Muster, das den entsprechenden Funktionsbaustein ggf. mit Inputdaten und den betroffenen Tabellen aufruft:

```
<X4SAP>
  <bapicall name="BAPI-Name" id="IDoc-ID">
    <input>
      <INPUT>
        <!--Parameter und Parameterwerte -->
      </INPUT>
    </input>
    <table>
      <!--Tabellen-->
    </table>
  </bapicall>
</X4SAP>
```

Beispiel

Um BAPI IDOCTYPE_READ_COMPLETE in SAP aufzurufen, erwartet dieser Adapter folgendes Input-XML-Dokument:

```
<X4SAP>
  <bapicall name="IDOCTYPE_READ_COMPLETE" id="IDoc-ID">
    <input>
      <INPUT>
        <PI_CIMTYP />
        <PI_IDOCTYP> <!--IDoc-Typ--> </PI_IDOCTYP>
        <PI_RELEASE>SY-S</PI_RELEASE>
        <PI_VERSION>3</PI_VERSION>
      </INPUT>
    </input>
    <table>
      <TABLES>
        <PT_FIELDS>
          <SEGMENTTYP />
          <FIELDNAME />
          <INTLEN />
          <EXTLEN />
          <FIELD_POS />
          <BYTE_FIRST />
          <BYTE_LAST />
          <ROLLNAME />
          <DOMNAME />
          <DATATYPE />
          <DESCRP />
          <ISOCODE />
          <VALUETAB />
        </PT_FIELDS>
        <PT_FVALUES>
          <STRNAME />
          <FIELDNAME />
          <FLDVALUE_L />
          <FLDVALUE_H />
          <DESCRP />
        </PT_FVALUES>
        <PT_MESSAGES>
          <MESTYP />
          <DESCRP />
          <IDOCTYP />
          <CIMTYP />
          <RELEASED />
        </PT_MESSAGES>
        <PT_SEGMENTS>
          <NR />
          <SEGMENTTYP />
          <SEGMENTDEF />
          <QUALIFIER />
          <SEGLEN />
          <PARSEG />
          <PARPNO />
          <PARFLG />
          <MUSTFL />
          <OCCMIN />
          <OCCMAX />
          <HLEVEL />
          <DESCRP />
          <GRP_MUSTFL />
```



```
        <GRP_OCCMIN />
        <GRP_OCCMAX />
        <REFSEGTYP />
    </PT_SEGMENTS>
</TABLES>
</table>
</bapicall>
</X4SAP>
```

6.9.2.2.2 IDoc senden

Input

Um ein IDoc an SAP zu senden, ein XML-Dokument wie das folgende als Input für den Adapter verwenden:

```

<X4SAP>
  <X4IDoc type="MATMAS02" description="">
    <X4Mandatory>
      <X4RecipientPartnerType>LS</X4RecipientPartnerType>
      <X4RecipientPartnerNumber>X4_T</X4RecipientPartnerNumber>
      <X4RecipientPartnerFunction />
      <X4SenderPort>SAPT01</X4SenderPort>
      <X4SenderPartnerType>LS</X4SenderPartnerType>
      <X4SenderPartnerNumber>ST01_200</X4SenderPartnerNumber>
      <X4SenderPartnerFunction />
      <X4SenderAddress />
      <X4SenderLogicalAddress />
      <X4MsgType>
        <MSGTYPE>MATMAS02</MSGTYPE>
      </X4MsgType>
    </X4Mandatory>
    <X4Structure>
<MATMAS02>
  <IDOC BEGIN="1">
    <EDI_DC40 SEGMENT="1">
      <TABNAM>EDI_DC40</TABNAM>
      <MANDT/>
      <DOCNUM/>
      <DOCREL/>
      <STATUS/>
      <DIRECT/>
      <OUTMOD/>
      <EXPRSS/>
      <TEST/>
      <IDOC TYP>MATMAS02</IDOC TYP>
      <CIMTYP/>
      <MESTYP>MATMAS</MESTYP>
      <MESCOD/>
      <MESFCT/>
      <STD/>
      <STDVRS/>
      <STDMES/>
      <SNDPOR>SAPJC0IDOC</SNDPOR>
      <SNDPRT>LS</SNDPRT>
      <SNDPFC/>
      <SNDP RN>JC0CLNT000</SNDP RN>
      <SNDSAD/>
      <SNDLAD/>
      <RCVPOR/>
      <RCVPRT>LS</RCVPRT>
      <RCVPFC/>
      <RCVPRN>TSTCLNT000</RCVPRN>
      <RCVSAD/>
      <RCVLAD/>
      <CREDAT>20050519</CREDAT>
      <CRETIM>104537</CRETIM>
      <REFINT/>
      <REFGRP/>
      <REFMES/>
      <ARCKEY/>
      <SERIAL/>

```

```
</EDI_DC40>
<E1MARAM SEGMENT="1">
  <MSGFN>005</MSGFN>
  <MATNR>BOXCOOKIES</MATNR>
  <ERSDA>20020801</ERSDA>
  <ERNAM>TIGGER</ERNAM>
  <LAEDA/>
  <AENAM/>
  <PSTAT>KBG</PSTAT>
  <LVORM/>
  <MTART>FERT</MTART>
  <MBRSH>L</MBRSH>
  <MATKL>G1113</MATKL>
  <BISMT/>
  <MEINS>PCE</MEINS>
  <BSTME/>
  <ZEINR/>
  <ZEIAR/>
  <ZEIVR/>
  <ZEIFO/>
  <AESZN/>
  <BLATT/>
  <BLANZ>000</BLANZ>
  <FERTH/>
  <FORMT/>
  <GROES/>
  <WRKST/>
  <NORMT/>
  <LABOR/>
  <EKWSL/>
  <BRGEW>0.550</BRGEW>
  <NTGEW>0.000</NTGEW>
  <GEWEI>KGM</GEWEI>
  <VOLUM/>
  <VOLEH/>
  <BEHVO/>
  <RAUBE/>
  <TEMPB/>
  <TRAGR/>
  <STOFF/>
  <SPART/>
  <KUNNR/>
  <WESCH/>
  <BWVOR/>
  <BWSCL/>
  <SAISO/>
  <ETIAR/>
  <ETIFO/>
  <EAN11/>
  <NUMTP/>
  <LAENG/>
  <BREIT/>
  <HOEHE/>
  <MEABM/>
  <PRDHA/>
  <CADKZ/>
  <ERGEW/>
```

<ERGEI/>
<ERVOL/>
<ERVOE/>
<GEWTO/>
<VOLTO/>
<VABME/>
<KZKFG/>
<XCHPF/>
<VHART/>
<FUELG/>
<STFAK/>
<MAGRV/>
<BEGRU/>
<QMPUR/>
<RBNRM/>
<MHDRZ/>
<MHDHB/>
<MHDLP/>
<VPSTA>KBG</VPSTA>
<EXTWG/>
<MSTAE/>
<MSTAV/>
<MSTDE/>
<MSTDV/>
<KZUMW/>
<KOSCH/>
<NRFHG/>
<MFRPN/>
<MFRNR/>
<BMATN/>
<MPROF/>
<PROFL/>
<IHIVI/>
<ILOOS/>
<KZGVH/>
<XGCHP/>
<COMPL/>
<KZEFF/>
<RDMHD/>
<IPRKZ/>
<PRZUS/>
<MTPOS_MARA/>
<GEWTO_NEW/>
<VOLTO_NEW/>
<WRKST_NEW/>
<AENNR/>
<MATFI/>
<CMREL/>
<SATNR/>
<SLED_BBD/>
<GTIN_VARIANT/>
<GENNR/>
<SERLV/>
<RMATP/>
<E1MAKTM SEGMENT="1">
 <MSGFN>005</MSGFN>
 <SPRAS>D</SPRAS>

```
<MAKTX>Schachtel mit Keksen</MAKTX>
<SPRAS_ISO>DE</SPRAS_ISO>
</E1MAKTM>
<E1MAKTM SEGMENT="1">
  <MSGFN>005</MSGFN>
  <SPRAS>E</SPRAS>
  <MAKTX>Box of cookies</MAKTX>
  <SPRAS_ISO>EN</SPRAS_ISO>
</E1MAKTM>
<E1MARCM SEGMENT="1">
  <MSGFN>005</MSGFN>
  <WERKS>0001</WERKS>
  <PSTAT>BG</PSTAT>
  <LVORM/>
  <BWTTY/>
  <MAABC/>
  <KZKRI/>
  <EKGRP/>
  <AUSME/>
  <DISPR/>
  <DISMM/>
  <DISPO/>
  <PLIFZ>0</PLIFZ>
  <WEBAZ>0</WEBAZ>
  <PERKZ>M</PERKZ>
  <AUSSS>0.00</AUSSS>
  <DISLS/>
  <BESKZ>E</BESKZ>
  <SOBSL/>
  <MINBE/>
  <EISBE/>
  <BSTMI/>
  <BSTMA/>
  <BSTFE/>
  <BSTRF/>
  <MABST/>
  <LOSFX/>
  <SBDKZ/>
  <LAGPR/>
  <ALTSL/>
  <KZAUS/>
  <AUSDT/>
  <NFMAT/>
  <KZBED/>
  <MISKZ/>
  <FHORI/>
  <PFREI/>
  <FFREI/>
  <RGEKZ/>
  <FEVOR/>
  <BEARZ/>
  <RUEZT/>
  <TRANZ/>
  <BASMG/>
  <DZEIT/>
  <MAXLZ/>
  <LZEIH/>
```

<KZPRO/>
<GPMKZ/>
<UEETO/>
<UEETK/>
<UNETO/>
<WZEIT/>
<ATPKZ/>
<VZUSL/>
<HERBL/>
<INSMK/>
<SSQSS/>
<KZDKZ/>
<UMLMC/>
<LADGR/>
<XCHPF/>
<USEQU/>
<LGRAD/>
<AUFTL/>
<PLVAR/>
<OTYPE/>
<OBJID/>
<MTVFP/>
<PERIV/>
<KZKFK/>
<VRVEZ/>
<VBAMG/>
<VBEAZ/>
<LIZYK/>
<BWSCL/>
<KAUTB/>
<KORDB/>
<STAWN/>
<HERKL/>
<HERKR/>
<EXPME/>
<MTVER/>
<PRCTR/>
<TRAME/>
<MRPPP/>
<SAUFT/>
<FXHOR/>
<VRMOD/>
<VINT1/>
<VINT2/>
<STLAL/>
<STLAN/>
<PLNNR/>
<APLAL/>
<LOSGR/>
<SOBSK/>
<FRTME/>
<LGPRO/>
<DISGR/>
<KAUSF/>
<QZGTP/>
<TAKZT/>
<RWPRO/>

<COPAM/>
<ABCIN/>
<AWSLS/>
<SERNP/>
<STDPD/>
<SFEPR/>
<XMCNG/>
<QSSYS/>
<LFRHY/>
<RDPRF/>
<VRBMT/>
<VRBWK/>
<VRBDT/>
<VRBFK/>
<AUTRU>X</AUTRU>
<PREFE/>
<PRENC/>
<PRENO/>
<PREND/>
<PRENE/>
<PRENG/>
<ITARK/>
<PRFRQ/>
<KZKUP/>
<STRGR/>
<LGFSB/>
<SCHGT/>
<CCFIX/>
<EPRIO/>
<QMATA/>
<PLNTY/>
<MMSTA/>
<SFCPF/>
<SHFLG/>
<SHZET/>
<MDACH/>
<KZECH/>
<MMSTD/>
<MFRGR/>
<FVIDK/>
<INDUS/>
<MOWNR/>
<MOGRU/>
<CASNR/>
<GPNUM/>
<STEUC/>
<FABKZ/>
<MATGR/>
<LOGGR/>
<VSPVB/>
<DPLFS/>
<DPLPU/>
<DPLHO/>
<MINLS/>
<MAXLS/>
<FIXLS/>
<LTINC/>

```
<COMPL/>
<CONVT/>
<FPRFM/>
<SHPRO/>
<FXPRU/>
<KZPSP/>
<OCMPF/>
<APOKZ/>
<AHDIS/>
<EISLO/>
<NCOST/>
<MEGRU/>
<ROTATION_DATE/>
<UCHKZ/>
<UCMAT/>
</E1MARCM>
<E1MBEWM SEGMENT="1">
  <MSGFN>005</MSGFN>
  <BWKEY>0001</BWKEY>
  <BWTAR/>
  <LVORM/>
  <VPRSV>S</VPRSV>
  <VERPR>0.00</VERPR>
  <STPRS>15.50</STPRS>
  <PEINH>1</PEINH>
  <BKLAS>7920</BKLAS>
  <VMVPR/>
  <VMVER/>
  <VMSTP/>
  <VMPEI/>
  <VMBKL/>
  <VJVPR>S</VJVPR>
  <VJVER>0.00</VJVER>
  <VJSTP>15.50</VJSTP>
  <LFGJA>2002</LFGJA>
  <LFMON>08</LFMON>
  <BWTTY/>
  <ZKPRS/>
  <ZKDAT/>
  <BWPRS/>
  <BWPRH/>
  <VJBWS/>
  <VJBWH/>
  <VVJLB/>
  <VVMLB/>
  <VVSAL/>
  <ZPLPR/>
  <ZPLP1/>
  <ZPLP2/>
  <ZPLP3/>
  <ZPLD1/>
  <ZPLD2/>
  <ZPLD3/>
  <KALKZ/>
  <KALKL/>
  <XLIFO/>
  <MYPOL/>
```



```

        <BWPH1/>
        <BWPS1/>
        <ABWKZ/>
        <PSTAT>BG</PSTAT>
        <KALN1>000100126602</KALN1>
        <KALNR>000100126603</KALNR>
        <BWVA1/>
        <BWVA2/>
        <BWVA3/>
        <VERS1/>
        <VERS2/>
        <VERS3/>
        <HRKFT/>
        <KOSGR/>
        <PPRDZ/>
        <PPRDL/>
        <PPRDV/>
        <PDATZ/>
        <PDATL/>
        <PDATV/>
        <EKALR>X</EKALR>
        <VPLPR>0.00</VPLPR>
        <MLMAA/>
        <MLAST/>
        <VJBKL>7920</VJBKL>
        <VJPEI>1</VJPEI>
        <HKMAT/>
        <EKLAS/>
        <QKLAS/>
        <MTUSE/>
        <MTORG/>
        <OWNPR/>
        <BWPEI>0</BWPEI>
    </E1MBEWM>
</E1MARAM>
</IDOC>
</MATMAS02>
</X4Structure>
    </X4IDoc>
</X4SAP>

```

6.9.2.3 SAP Connector (JCo 3)




SAP Connector (JCo 3): Ermöglicht die Übermittlung von Daten vom X4 ESB an ein *SAP R/3*-System. Er erzeugt einen beliebigen Remote Function Call (RFC) bzw. ruft einen beliebigen BAPI-Funktionsbaustein in einem SAP-System via *JCo 3* auf. Zudem können IDocs an einen SAP-Host versandt werden.

Ehemalige Bezeichnung bis Version 4.5.1: *SAP (JCo 3) Request Adapter*

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>Execute</i> : Funktions-Adapter ausführen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.sapjco3.SAPAdapter</i> : Hauptklasse (Standard)
<i>PropertyFile</i>	URL zur Konfigurationsdatei für den Adapter Mögliche Werte: Beliebige innerhalb der Java-VM gültige URL (z.B. <i>xstore://Project/Folder/Configuration.properties</i>)
<i>ReplaceEscapedControlChars</i>	Maskierte Steuerzeichen (z.B.: <i>&#31;</i>) in den von der JCo-Bibliothek zurückgegebenen XML-Strings durch das Ersetzungszeichen () austauschen Mögliche Werte: <ul style="list-style-type: none"> • <i>true</i>: Maskierte Steuerzeichen ersetzen • <i>false</i>: Maskierte Steuerzeichen nicht ersetzen

6.9.2.3.1 RFC/BAPI aufrufen

Input

Der Adapter erwartet ein Input-XML-Dokument nach folgendem Muster, das den entsprechenden Funktionsbaustein ggf. mit Inputdaten und den betroffenen Tabellen aufruft:

```
<X4SAP>
  <bapical name="BAPI-Name" id="IDoc-ID">
    <input>
      <INPUT>
        <!--Parameter und Parameterwerte-->
      </INPUT>
    </input>
    <table>
      (Tabellen)
    </table>
  </bapical>
</X4SAP>
```

Beispiel

Um BAPI IDOCTYPE_READ_COMPLETE in SAP aufzurufen erwartet dieser Adapter folgendes Input-XML-Dokument:

```

<X4SAP>
  <bapicall name="IDOCTYPE_READ_COMPLETE" id="IDoc-ID">
    <input>
      <INPUT>
        <PI_CIMTYP />
        <PI_IDOCTYP> <!--IDoc-Typ--> </PI_IDOCTYP>
        <PI_RELEASE>SY-S</PI_RELEASE>
        <PI_VERSION>3</PI_VERSION>
      </INPUT>
    </input>
    <table>
      <TABLES>
        <PT_FIELDS>
          <SEGMENTTYP />
          <FIELDNAME />
          <INTLEN />
          <EXTLEN />
          <FIELD_POS />
          <BYTE_FIRST />
          <BYTE_LAST />
          <ROLLNAME />
          <DOMNAME />
          <DATATYPE />
          <DESCRP />
          <ISOCODE />
          <VALUETAB />
        </PT_FIELDS>
        <PT_FVALUES>
          <STRNAME />
          <FIELDNAME />
          <FLDVALUE_L />
          <FLDVALUE_H />
          <DESCRP />
        </PT_FVALUES>
        <PT_MESSAGES>
          <MESTYP />
          <DESCRP />
          <IDOCTYP />
          <CIMTYP />
          <RELEASED />
        </PT_MESSAGES>
        <PT_SEGMENTS>
          <NR />
          <SEGMENTTYP />
          <SEGMENTDEF />
          <QUALIFIER />
          <SEGLEN />
          <PARSEG />
          <PARPNO />
          <PARFLG />
          <MUSTFL />
          <OCCMIN />
          <OCCMAX />
          <HLEVEL />
          <DESCRP />
          <GRP_MUSTFL />
        </PT_SEGMENTS>
      </TABLES>
    </table>
  </bapicall>

```

```
        <GRP_OCCMIN />
        <GRP_OCCMAX />
        <REFSEGTYP />
    </PT_SEGMENTS>
</TABLES>
</table>
</bapicall>
</X4SAP>
```

6.9.2.3.2 IDoc senden

Um ein IDoc an SAP zu senden, ein XML-Dokument wie das folgende als Input für den Adapter verwenden:

```

<X4SAP>
  <X4IDoc type="MATMAS02" description="">
    <X4Mandatory>
      <X4RecipientPartnerType>LS</X4RecipientPartnerType>
      <X4RecipientPartnerNumber>X4_T</X4RecipientPartnerNumber>
      <X4RecipientPartnerFunction />
      <X4SenderPort>SAPT01</X4SenderPort>
      <X4SenderPartnerType>LS</X4SenderPartnerType>
      <X4SenderPartnerNumber>ST01_200</X4SenderPartnerNumber>
      <X4SenderPartnerFunction />
      <X4SenderAddress />
      <X4SenderLogicalAddress />
      <MessageType>
        <MSGTYPE>MATMAS02</MSGTYPE>
      </MessageType>
    </X4Mandatory>
    <X4Structure>
<MATMAS02>
  <IDOC BEGIN="1">
    <EDI_DC40 SEGMENT="1">
      <TABNAM>EDI_DC40</TABNAM>
      <MANDT/>
      <DOCNUM/>
      <DOCREL/>
      <STATUS/>
      <DIRECT/>
      <OUTMOD/>
      <EXPRSS/>
      <TEST/>
      <IDOCTYP>MATMAS02</IDOCTYP>
      <CIMTYP/>
      <MESTYP>MATMAS</MESTYP>
      <MESCOD/>
      <MESFCT/>
      <STD/>
      <STDVRS/>
      <STDMES/>
      <SNDPOR>SAPJC0IDOC</SNDPOR>
      <SNDPRT>LS</SNDPRT>
      <SNDPFC/>
      <SNDPRN>JC0CLNT000</SNDPRN>
      <SNDSAD/>
      <SNDLAD/>
      <RCVPOR/>
      <RCVPRT>LS</RCVPRT>
      <RCVPFC/>
      <RCVPRN>TSTCLNT000</RCVPRN>
      <RCVSAD/>
      <RCVLAD/>
      <CREDAT>20050519</CREDAT>
      <CRETIM>104537</CRETIM>
      <REFINT/>
      <REFGRP/>
      <REFMES/>
      <ARCKEY/>
      <SERIAL/>

```

```
</EDI_DC40>
<E1MARAM SEGMENT="1">
  <MSGFN>005</MSGFN>
  <MATNR>BOXCOOKIES</MATNR>
  <ERSDA>20020801</ERSDA>
  <ERNAM>TIGGER</ERNAM>
  <LAEDA/>
  <AENAM/>
  <PSTAT>KBG</PSTAT>
  <LVORM/>
  <MTART>FERT</MTART>
  <MBRSH>L</MBRSH>
  <MATKL>G1113</MATKL>
  <BISMT/>
  <MEINS>PCE</MEINS>
  <BSTME/>
  <ZEINR/>
  <ZEIAR/>
  <ZEIVR/>
  <ZEIFO/>
  <AESZN/>
  <BLATT/>
  <BLANZ>000</BLANZ>
  <FERTH/>
  <FORMT/>
  <GROES/>
  <WRKST/>
  <NORMT/>
  <LABOR/>
  <EKWSL/>
  <BRGEW>0.550</BRGEW>
  <NTGEW>0.000</NTGEW>
  <GEWEI>KGM</GEWEI>
  <VOLUM/>
  <VOLEH/>
  <BEHVO/>
  <RAUBE/>
  <TEMPB/>
  <TRAGR/>
  <STOFF/>
  <SPART/>
  <KUNNR/>
  <WESCH/>
  <BWVOR/>
  <BWSCL/>
  <SAISO/>
  <ETIAR/>
  <ETIFO/>
  <EAN11/>
  <NUMTP/>
  <LAENG/>
  <BREIT/>
  <HOEHE/>
  <MEABM/>
  <PRDHA/>
  <CADKZ/>
  <ERGEW/>
```

```
<ERGEI/>
<ERVOL/>
<ERVOE/>
<GEWTO/>
<VOLTO/>
<VABME/>
<KZKFG/>
<XCHPF/>
<VHART/>
<FUELG/>
<STFAK/>
<MAGRV/>
<BEGRU/>
<QMPUR/>
<RBNRM/>
<MHDRZ/>
<MHDHB/>
<MHDLP/>
<VPSTA>KBG</VPSTA>
<EXTWG/>
<MSTAE/>
<MSTAV/>
<MSTDE/>
<MSTDV/>
<KZUMW/>
<KOSCH/>
<NRFHG/>
<MFRPN/>
<MFRNR/>
<BMATN/>
<MPROF/>
<PROFL/>
<IHIVI/>
<ILOOS/>
<KZGVH/>
<XGCHP/>
<COMPL/>
<KZEFF/>
<RDMHD/>
<IPRKZ/>
<PRZUS/>
<MTPOS_MARA/>
<GEWTO_NEW/>
<VOLTO_NEW/>
<WRKST_NEW/>
<AENNR/>
<MATFI/>
<CMREL/>
<SATNR/>
<SLED_BBD/>
<GTIN_VARIANT/>
<GENNR/>
<SERLV/>
<RMATP/>
<E1MAKTM SEGMENT="1">
    <MSGFN>005</MSGFN>
    <SPRAS>D</SPRAS>
```



```
<MAKTX>Schachtel mit Keksen</MAKTX>
<SPRAS_ISO>DE</SPRAS_ISO>
</E1MAKTM>
<E1MAKTM SEGMENT="1">
  <MSGFN>005</MSGFN>
  <SPRAS>E</SPRAS>
  <MAKTX>Box of cookies</MAKTX>
  <SPRAS_ISO>EN</SPRAS_ISO>
</E1MAKTM>
<E1MARCM SEGMENT="1">
  <MSGFN>005</MSGFN>
  <WERKS>0001</WERKS>
  <PSTAT>BG</PSTAT>
  <LVORM/>
  <BWTTY/>
  <MAABC/>
  <KZKRI/>
  <EKGRP/>
  <AUSME/>
  <DISPR/>
  <DISMM/>
  <DISPO/>
  <PLIFZ>0</PLIFZ>
  <WEBAZ>0</WEBAZ>
  <PERKZ>M</PERKZ>
  <AUSSS>0.00</AUSSS>
  <DISLS/>
  <BESKZ>E</BESKZ>
  <SOBSL/>
  <MINBE/>
  <EISBE/>
  <BSTMI/>
  <BSTMA/>
  <BSTFE/>
  <BSTRF/>
  <MABST/>
  <LOSFX/>
  <SBDKZ/>
  <LAGPR/>
  <ALTSL/>
  <KZAUS/>
  <AUSDT/>
  <NFMAT/>
  <KZBED/>
  <MISKZ/>
  <FHORI/>
  <PFREI/>
  <FFREI/>
  <RGEKZ/>
  <FEVOR/>
  <BEARZ/>
  <RUEZT/>
  <TRANZ/>
  <BASMG/>
  <DZEIT/>
  <MAXLZ/>
  <LZEIH/>
```

<KZPRO/>
<GPMKZ/>
<UEETO/>
<UEETK/>
<UNETO/>
<WZEIT/>
<ATPKZ/>
<VZUSL/>
<HERBL/>
<INSMK/>
<SSQSS/>
<KZDKZ/>
<UMLMC/>
<LADGR/>
<XCHPF/>
<USEQU/>
<LGRAD/>
<AUFTL/>
<PLVAR/>
<OTYPE/>
<OBJID/>
<MTVFP/>
<PERIV/>
<KZKFK/>
<VRVEZ/>
<VBAMG/>
<VBEAZ/>
<LIZYK/>
<BWSCL/>
<KAUTB/>
<KORDB/>
<STAWN/>
<HERKL/>
<HERKR/>
<EXPME/>
<MTVER/>
<PRCTR/>
<TRAME/>
<MRPPP/>
<SAUFT/>
<FXHOR/>
<VRMOD/>
<VINT1/>
<VINT2/>
<STLAL/>
<STLAN/>
<PLNNR/>
<APLAL/>
<LOSGR/>
<SOBSK/>
<FRTME/>
<LGPRO/>
<DISGR/>
<KAUSF/>
<QZGTP/>
<TAKZT/>
<RWPRO/>

<COPAM/>
<ABCIN/>
<AWSLS/>
<SERNP/>
<STDPD/>
<SFEPR/>
<XMCNG/>
<QSSYS/>
<LFRHY/>
<RDPRF/>
<VRBMT/>
<VRBWK/>
<VRBDT/>
<VRBFK/>
<AUTRU>X</AUTRU>
<PREFE/>
<PRENC/>
<PRENO/>
<PREND/>
<PRENE/>
<PRENG/>
<ITARK/>
<PRFRQ/>
<KZKUP/>
<STRGR/>
<LGFSB/>
<SCHGT/>
<CCFIX/>
<EPRIO/>
<QMATA/>
<PLNTY/>
<MMSTA/>
<SFCPF/>
<SHFLG/>
<SHZET/>
<MDACH/>
<KZECH/>
<MMSTD/>
<MFRGR/>
<FVIDK/>
<INDUS/>
<MOWNR/>
<MOGRU/>
<CASNR/>
<GPNUM/>
<STEUC/>
<FABKZ/>
<MATGR/>
<LOGGR/>
<VSPVB/>
<DPLFS/>
<DPLPU/>
<DPLHO/>
<MINLS/>
<MAXLS/>
<FIXLS/>
<LTINC/>

```
<COMPL/>
<CONVT/>
<FPRFM/>
<SHPRO/>
<FXPRU/>
<KZPSP/>
<OCMPF/>
<APOKZ/>
<AHDIS/>
<EISLO/>
<NCOST/>
<MEGRU/>
<ROTATION_DATE/>
<UCHKZ/>
<UCMAT/>
</E1MARCM>
<E1MBEWM SEGMENT="1">
  <MSGFN>005</MSGFN>
  <BWKEY>0001</BWKEY>
  <BWTAR/>
  <LVORM/>
  <VPRSV>S</VPRSV>
  <VERPR>0.00</VERPR>
  <STPRS>15.50</STPRS>
  <PEINH>1</PEINH>
  <BKLAS>7920</BKLAS>
  <VMVPR/>
  <VMVER/>
  <VMSTP/>
  <VMPEI/>
  <VMBKL/>
  <VJVPR>S</VJVPR>
  <VJVER>0.00</VJVER>
  <VJSTP>15.50</VJSTP>
  <LFGJA>2002</LFGJA>
  <LFMON>08</LFMON>
  <BWTTY/>
  <ZKPRS/>
  <ZKDAT/>
  <BWPRS/>
  <BWPRH/>
  <VJBWS/>
  <VJBWH/>
  <VVJLB/>
  <VVMLB/>
  <VVSAL/>
  <ZPLPR/>
  <ZPLP1/>
  <ZPLP2/>
  <ZPLP3/>
  <ZPLD1/>
  <ZPLD2/>
  <ZPLD3/>
  <KALKZ/>
  <KALKL/>
  <XLIFO/>
  <MYPOL/>
```

```

    <BWPH1/>
    <BWPS1/>
    <ABWKZ/>
    <PSTAT>BG</PSTAT>
    <KALN1>000100126602</KALN1>
    <KALNR>000100126603</KALNR>
    <BWVA1/>
    <BWVA2/>
    <BWVA3/>
    <VERS1/>
    <VERS2/>
    <VERS3/>
    <HRKFT/>
    <KOSGR/>
    <PPRDZ/>
    <PPRDL/>
    <PPRDV/>
    <PDATZ/>
    <PDATL/>
    <PDATV/>
    <EKALR>X</EKALR>
    <VPLPR>0.00</VPLPR>
    <MLMAA/>
    <MLAST/>
    <VJBKL>7920</VJBKL>
    <VJPEI>1</VJPEI>
    <HKMAT/>
    <EKLAS/>
    <QKLAS/>
    <MTUSE/>
    <MTORG/>
    <OWNPR/>
    <BWPEI>0</BWPEI>
  </E1MBEWM>
</E1MARAM>
</IDOC>
</MATMAS02>
</X4Structure>
  </X4IDoc>
</X4SAP>

```

6.9.2.4 SAP Explorer(JCo 2)



SAP Explorer (JCo 2): Erzeugt Strukturbeschreibungen von IDocs und BAPIs, listet BAPIs oder IDocs oder erzeugt leere IDoc- und BAPI-Strukturen und gibt je nach Operation ein entsprechendes XML-Dokument aus.

Ehemalige Bezeichnung bis Version 4.5.1: *SAP(JCO 2)Explorer Adapter*

**Bitte beachten Sie:**

Zur Nutzung des *SAP Explorers (JCo 2)* werden Rechte zur Remote-Ausführung folgender RFC-Bausteine benötigt:

- RPY_BOR_TREE_INIT
- IDOCTYPES_LIST_WITH_MESSAGES
- IDOCTYPE_READ_COMPLETE
- RPY_OBJECTTYPE_READ
- RFC_FUNCTION_SEARCH

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Get BapiTree</i>: Liste aller verfügbaren BAPIs ausgeben • <i>Get IDocList</i>: Liste aller verfügbaren IDocs ausgeben • <i>Read Bapi</i>: Beschreibung der BAPI-Struktur ausgeben • <i>Read IDoc</i>: Beschreibung der IDoc-Struktur ausgeben • <i>Create Empty Bapi</i>: Leere BAPI-Struktur zur Verwendung in der X4 Suite erzeugen • <i>Create Empty Idoc</i>: Leere IDoc-Struktur zur Verwendung in der X4 Suite erzeugen • <i>SearchForBapis</i>: Liste aller BAPIs ausgegeben, die dem Suchausdruck entsprechen • <i>Nothing</i>: Input unverändert ausgeben
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.sapadapter.SAPBapidocNotationExplorer</i>: Hauptklasse (Standard)</p>
<i>PropertyFile</i>	<p>URL zur Konfigurationsdatei für den Adapter (hebt die Adapter-Konfiguration auf)</p> <p>Mögliche Werte: Beliebige innerhalb der Java-VM gültige URL (z. B. <i>xstore:///Project/Folder/Configuration.properties</i>)</p>
<i>Client</i>	SAP-Mandant
<i>User</i>	Benutzername

<i>Password</i>	Passwort
<i>Language</i>	Anmeldesprache
<i>SystemNumber</i>	SAP-System-Nummer
<i>RouterString</i>	Router-String
<i>SystemType</i>	Art des entfernten Hosts Mögliche Werte: <ul style="list-style-type: none"> • 2: R/2 • 3: R/3 • E: Extern

Input

Dieser Funktions-Adapter erwartet je nach gewählter Operation z.T. eine adapterspezifische Input-XML-Struktur.

Operation	
<i>Get BapiTree</i>	Für die Operation <code>Get BapiTree</code> wird kein Input-XML-Dokument benötigt
<i>Get IDocList</i>	Für die Operation <code>Get IDocList</code> wird kein Input-XML-Dokument benötigt
<i>Read Bapi</i>	Für die Operation <code>Read Bapi</code> erwartet dieser Funktions-Adapter folgendes Input-XML-Dokument: <div><Object><!--BAPI-ID--></Object></div>
<i>Read IDoc</i>	Für die Operation <code>Read IDoc</code> erwartet dieser Funktions-Adapter folgendes Input-XML-Dokument: <div><ReadIDoc><!--IDoc-ID--></ReadIDoc></div>
<i>Create Empty Bapi</i>	Für die Operation <code>Create Empty Bapi</code> erwartet dieser Funktions-Adapter folgendes Input-XML-Dokument: <div><Bapiname><!--Name des BAPIs--></Bapiname></div>
<i>Create Empty Idoc</i>	Für die Operation <code>Create Empty Idoc</code> erwartet dieser Funktions-Adapter folgendes Input-XML-Dokument: <div><IDoc><!--Name des IDocs--></IDoc></div>

Operation	
<i>SearchForBapis</i>	<p>Für die Operation <i>SearchForBapis</i> erwartet dieser Funktions-Adapter folgendes Input-XML-Dokument:</p> <div data-bbox="687 427 1434 512" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <pre><Search functionname="BAPI-/Funktionsname" /></pre> </div>
<i>Nothing</i>	Für die Operation <i>Nothing</i> wird kein Input-XML-Dokument benötigt.

6.9.2.5 SAP Explorer (JCo 3)



SAP Explorer (JCo3): Erzeugt Strukturbeschreibungen von IDocs und BAPIs, listet BAPIs oder IDocs oder erzeugt leere IDoc- und BAPI-Strukturen und gibt je nach Operation ein entsprechendes XML-Dokument aus.

Ehemalige Bezeichnung bis Version 4.5.1: *SAP(JCo 3)Explorer Adapter*



Bitte beachten Sie:


Zur Nutzung des *SAP Explorers (JCo 3)* werden Rechte zur Remote-Ausführung folgender RFC-Bausteine benötigt:

- RPY_BOR_TREE_INIT
- IDOCTYPES_LIST_WITH_MESSAGES
- IDOCTYPE_READ_COMPLETE
- RPY_OBJECTTYPE_READ
- RFC_FUNCTION_SEARCH

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>GetBapiTree</i>: Liste aller verfügbaren BAPIs ausgeben • <i>GetIDocList</i>: Liste aller verfügbaren IDocs ausgeben • <i>ReadBapi</i>: Beschreibung der BAPI-Struktur ausgeben • <i>ReadIDoc</i>: Beschreibung der IDoc-Struktur ausgeben • <i>CreateEmptyBapi</i>: Leere BAPI-Struktur zur Verwendung in der X4 Suite erzeugen • <i>CreateEmptyIDoc</i>: Leere IDoc-Struktur zur Verwendung in der X4 Suite erzeugen • <i>SearchForBapis</i>: Liste aller BAPIs ausgegeben, die dem Suchausdruck entsprechen
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.sapjco3.SAPBapildocNotationExplorer</i>: Hauptklasse (Standard)</p>
<i>PropertyFile</i>	<p>URL zur Konfigurationsdatei für den Adapter</p> <p>Mögliche Werte: Beliebige innerhalb der Java-VM gültige URL (z.B. <i>xstore://Project/Folder/Configuration.properties</i>)</p>
<i>Client</i>	SAP-Mandant (jco.client.client in der Properties-Datei)
<i>User</i>	Benutzername (jco.client.user in der Properties-Datei)
<i>Password</i>	Passwort (jco.client.passwd in der Properties-Datei)
<i>Language</i>	Anmeldesprache (jco.client.lang in der Properties-Datei)
<i>SystemNumber</i>	SAP-System-Nummer (jco.client.sysnr in der Properties-Datei)
<i>RouterString</i>	Router-String (jco.client.ashost in der Properties-Datei)
<i>SystemType</i>	<p>Art des entfernten Hosts (jco.client.type in der Properties-Datei)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • 2: R/2 • 3: R/3 • E: Extern
<i>ReplaceEscapedControlChars</i>	<p>Maskierte Steuerzeichen (z.B.: <i>&#31;</i>) in den von der JCo-Bibliothek zurückgegebenen XML-Strings durch das Ersetzungszeichen () austauschen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • true: Maskierte Steuerzeichen ersetzen • false: Maskierte Steuerzeichen nicht ersetzen

Input

Dieser Funktions-Adapter erwartet je nach gewählter Operation z.T. eine adapterspezifische Input-XML-Struktur.

Operation	
<i>Get BapiTree</i>	Für die Operation <code>Get BapiTree</code> wird kein Input-XML-Dokument benötigt.
<i>Get IDocList</i>	Für die Operation <code>Get IDocList</code> wird kein Input-XML-Dokument benötigt.
<i>Read Bapi</i>	Für die Operation <code>Read Bapi</code> erwartet dieser Funktions-Adapter folgendes Input-XML-Dokument: <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <code><Object><!--BAPI-ID--></Object></code> </div>
<i>Read IDoc</i>	Für die Operation <code>Read IDoc</code> erwartet dieser Funktions-Adapter folgendes Input-XML-Dokument: <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <code><ReadIDoc><!--IDoc-ID--></ReadIDoc></code> </div>
<i>Create Empty Bapi</i>	Für die Operation <code>Create Empty Bapi</code> erwartet dieser Funktions-Adapter folgendes Input-XML-Dokument: <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <code><Bapiname><!--Name des BAPIs--></Bapiname></code> </div>
<i>Create Empty Idoc</i>	Für die Operation <code>Create Empty Idoc</code> erwartet dieser Funktions-Adapter folgendes Input-XML-Dokument: <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <code><IDoc><!--Name des IDocs--></IDoc></code> </div>
<i>SearchForBapis</i>	Für die Operation <code>SearchForBapis</code> erwartet dieser Funktions-Adapter folgendes Input-XML-Dokument: <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <code><Search functionname="BAPI-/Funktionsname" /></code> </div>

6.9.2.6 SAP Listener (JCo 2)

Der *SAP Listener (JCo)* ermöglicht die Verarbeitung von eingehenden Kommunikationsanfragen via BAPI/RFC und von IDocs eines SAP-Systems. Er stellt dabei automatisch ggf. unterbrochene Verbindungen wieder her.

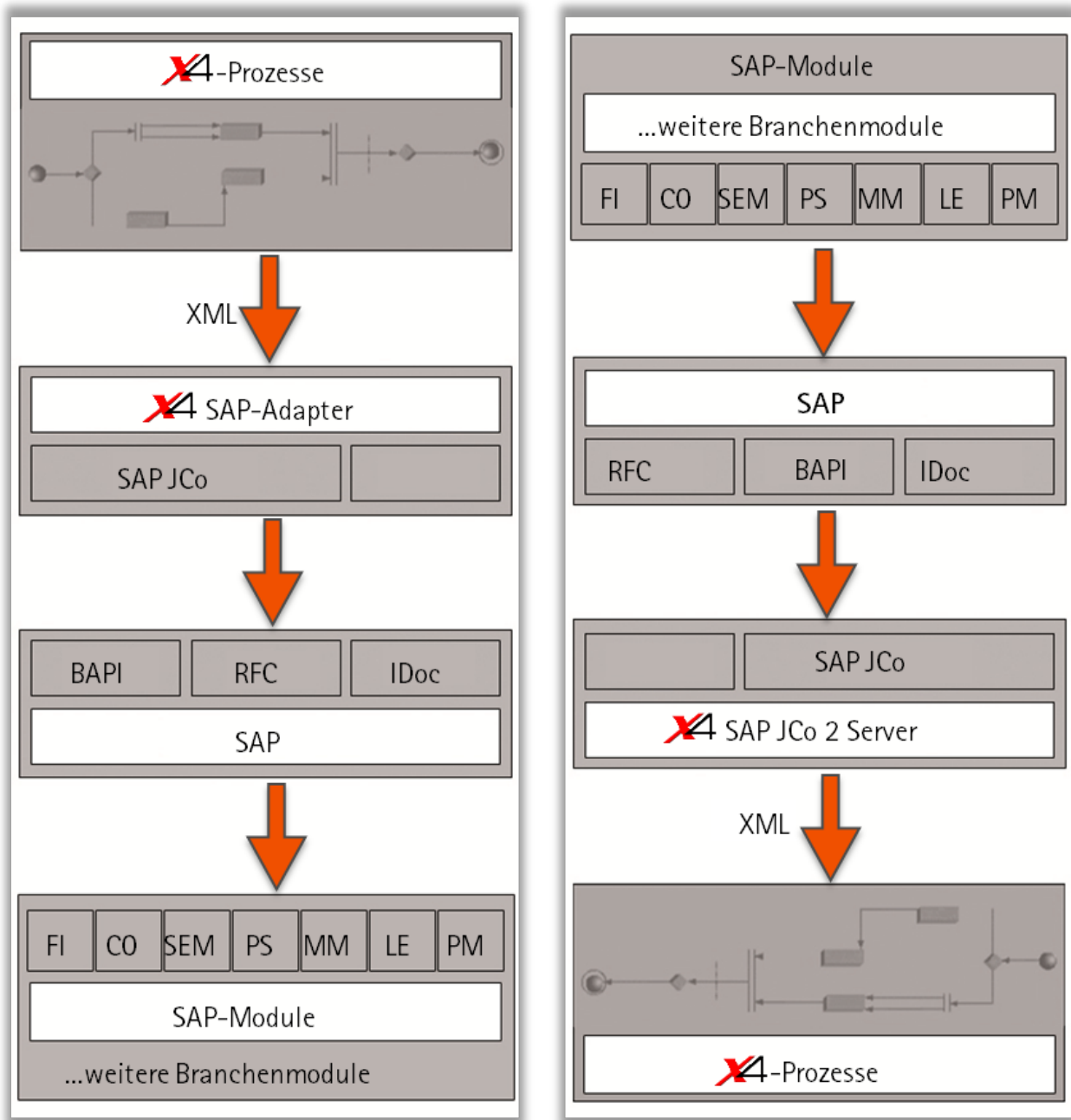
Ehemalige Bezeichnung bis Version 4.5.1: *X4 SAP JCo 2 Listener*

Funktionen von X4 SAP JCo 2

Der *X4 Server* ermöglicht die bidirektionale Anbindung von *SAP R/3*-Systemen über den *SAP Java Connector* in Version 2 (*JCo 2*). Dabei sind folgende Komponenten relevant:

- *SAP JCo 2 Server*: Ermöglicht die Verarbeitung von eingehenden Kommunikationsanfragen via BAPI/RFC und von IDocs eines SAP-Systems an den *X4 Server*. Der *SAP JCo 2 Server* stellt dabei automatisch ggf. unterbrochene Verbindungen wieder her.

- *SAP-Adapter*: In *X4*-Prozessen kann dieser Funktions-Adapter einen beliebigen Remote Function Call (RFC) bzw. einen beliebigen BAPI-Funktionsbaustein in einem SAP-System aufrufen. Zudem können IDocs an *SAP R/3* versandt werden, siehe *Adapter und Bausteine, SAP Connector (JCo 2)*.
- *SAP Explorer (JCo2)*: Dieser ermöglicht während der Prozess-Entwicklung das Erzeugen von Strukturbeschreibungen von IDocs und BAPIs. Zudem listet er BAPIs oder IDocs oder erzeugt leere IDoc- und BAPI-Strukturen und gibt je nach Operation ein entsprechendes XML-Dokument aus, siehe *Adapter und Bausteine, SAP Explorer (JCo 2)*.



Über den *JCo 2 Server* kann mit der X4 Suite ein SAP-System angebunden werden. Es können auch mehrere SAP-Systeme über eine jeweils separate *JCo Server*-Instanz angebunden werden. Zum Verwenden des *JCo 2-Servers* ist eine entsprechende Konfiguration erforderlich:

- X4-seitige Konfiguration in `x4config.xml`, siehe [SAP JCo 2 Server-Konfiguration](#)
- Definition von Verbindungs-Eigenschaften, siehe [Properties für den JCo 2 Server](#)
- Zuordnung der Funktionsbaustein-Aufrufe zu technischen Prozessen, siehe [Funktionsbaustein-/Methodendefinitionen \(X4 SAP JCo 2\)](#)

Identifikation von JCo 2-Servern über ProgID

Jede *JCo 2 Server*-Instanz bzw. jedes an SAP angebundene System kommuniziert über eine eindeutige *ProgID*, die in der Server-Properties-Datei, standardmäßig `jco_server_settings.prop`, konfiguriert wird und beim Start der Server-Instanz dem SAP-System bekannt gemacht wird. Pro SAP-System können mehrere ProgIDs gleichzeitig verwendet werden. Über die *ProgID* werden IDocs und RFC/BAPIs an das jeweilige Ziel-System eindeutig "adressiert".

6.9.2.6.1 Installation von X4 SAP JCo 2

Welche Schritte zur Installation von *X4 SAP JCo 2* erforderlich sind

Beispiel

Als Beispiel wird eine Installation für den *JBoss* Applikations-Server auf Basis eines 32-Bit-Windows-Systems beschrieben.

6.9.2.6.1.1 Voraussetzungen

Zur Installation eines *SAP JCo 2 Server* im *X4 Server* werden *SAP*-Bibliotheken benötigt, die z. T. betriebssystem- und architekturabhängig sind. Diese Bibliotheken werden auch für den [SAP Connector \(JCo 2\)](#) und den [SAP Explorer \(JCo 2\)](#) benötigt, die bereits in `x4.ear` enthalten sind.

Als registrierter SAP-Partner können Sie die erforderlichen Bibliotheken sowie weiterführende Dokumentation über das *SAPNet* von <http://service.sap.com/connectors> herunterladen.

Wählen Sie hierbei zueinander kompatible Versionsnummern der Bibliotheken (diese sind zudem betriebssystem- und architekturabhängig).

Achten Sie darauf, dass das verwendete Betriebssystem kompatibel mit der eingesetzten JDK-Version und der Version Ihres SAP-Systems ist. Folgende Matrix zeigt den Zusammenhang dieser drei Variablen.

		OS		SAP	
		32 bit	64 bit	32 bit	64 bit
JDK	32 bit	x	x	x	-
	64 bit	-	x	-	x

6.9.2.6.1.2 Bibliotheken installieren

1. Native Bibliotheken (je nach System) in den entsprechenden Ordner des Applikations-Servers für native Binär-Abhängigkeiten oder in das bin-Verzeichnis von Java kopieren.

Beispiel: Unter Windows 32-Bit: `librfc32.dll`, `msvcr71.dll`, `msvcp71.dll` und `sapjcorfc.dll` nach `C:\Windows\System32` oder nach `C:\X4\Server\jdk1.7\bin` kopieren

2. Folgende Java-Bibliotheken in den Classpath des Applikations-Servers kopieren:

- `sapidoc.jar`
- `sapjco.jar`
- `sapidocjco.jar`

Beispiel JBoss 4.2.3.GA: Diese Bibliotheken nach `C:`

`\X4\JBoss4.2.3.GA\server\default\lib` kopieren.

Beispiel JBoss AS 7.1.1: Diese Bibliotheken unter `C:\X4\Server\jboss-`

`as-7.1.1.Final\standalone\deployments` in das `lib`-Verzeichnis von `X4.ear` kopieren.

3. *X4 Server* neu starten.

6.9.2.6.1.3 Installation prüfen

Über den *SAP Explorer (JCo 2)* (siehe *X4 Experience*-Beispiel `X4Experience/AdapterSuites/X4SAPSuite/ SAP (JCo2) Explorer (Prototype) /Processes/CreateEmptyBapi.wrf`) können Sie prüfen, ob eine Verbindung aufgebaut werden kann:

1. Sicherstellen, dass die Konfiguration des *SAP JCo 2 Server* vollständig ist, siehe *X4 Server-Konfiguration (X4config.xml)*.
2. Adapterparameter `SystemType` auf `3` setzen (für eine Verbindung zu *SAP R/3*).
3. Sicherstellen, dass Adapteroperation `Create Empty Bapi` gewählt ist.
4. Prozess mit dem Input-XML `<Bapiname>BAPI's name</Bapiname>` ausführen und Ergebnis-XML prüfen.

6.9.2.6.2 Konfiguration des X4 SAP Listeners (JCo 2)

Welche Konfiguration *X4*-seitig angepasst werden muss, um den *JCo 2 Server* zu starten, von außen erreichbar zu machen und um den eingehenden RFC/BAPIs und IDocs entsprechende Handler-Prozesse in der *X4 Suite* zuzuordnen, erfahren Sie im Folgenden.

6.9.2.6.2.1 SAP JCo 2 Server-Konfiguration

Im Element `<jcoServer>` können Sie in `X4config.xml` die *SAP Java Connector*-Einstellungen für den *X4 Server* konfigurieren.

- ❗ Ab der *X4 Server*-Revision 27282 besitzt das Element `<jcoServer>` eine veränderte Struktur: IDoc-Handler und Function-Handler wurden in die *JCo 2 Server Properties* ausgelagert, siehe [Properties für den JCo 2 Server](#). Zudem ist nun möglich, *mehrere* Properties-Dateien und Funktionsbaustein-/Methoden-Definitionen in `X4config.xml` anzugeben, um Verbindungen zu mehreren SAP-Systemen zu ermöglichen.
- Falls mehrere Properties-Dateien und Funktionsbaustein-/Methoden-Definitionen angegeben werden, erfolgt deren Zuordnung über die Reihenfolge. d. h. die erste *JCo Server*-Properties-Datei in `<propertiesURL>` wird der ersten Funktionsbaustein-/Methoden-Definition in `<methodsURL>` zugewiesen, die zweite *JCo Server*-Properties-Datei wird der zweiten Funktionsbaustein-/Methoden-Definition zugewiesen etc.

Element `<jcoServer>`

<code><propertiesURL></code>	<p>URL zu einer Properties-Datei, die <i>JCo 2 Server</i>-Einstellungen enthält; Ab <i>X4 Server</i>-Version 4.2 können Sie hier mehrere durch Semikolon getrennte URLs zu verschiedenen Properties-Dateien angeben</p> <p>Mögliche Werte: Beliebige gültige URL (z. B. <code>file:///C:/X4/X4DB/0/jco_server_settings.prop</code>)</p>
<code><methodsURL></code>	<p>URL zu einem XML-Dokument, das die Zuordnung von Funktionsbausteinen/Methoden und dem entsprechenden Verarbeitungsprozess im <i>X4 Server</i> enthält; Ab <i>X4 Server</i>-Version 4.2 können Sie hier mehrere durch Kommas getrennte URLs zu verschiedenen Konfigurations-Dokumenten angeben</p> <p>Mögliche Werte: Beliebige gültige URL (z. B. <code>file:///C:/X4/X4DB/0/jco_method_def.xml</code>)</p>

Element `<sapadapter>`

Im Element `<sapadapter>`, das dem Element `<jcoServer>` in `X4config.xml` nachfolgt, können Sie die Standard-Verbindungseinstellungen für den *SAP Connector (JCo 2)* konfigurieren. Diese wird als Fallback-Konfiguration verwendet.

<code><propertiesURL></code>	URL zu einer Properties-Datei, die Verbindungseinstellungen enthält, die herangezogen werden, wenn der <i>SAP Connector (JCo 2)</i> keine Einstellungen findet.
------------------------------------	---

6.9.2.6.2 Properties für den JCo 2 Server

In einem Konfigurations-Dokument (standardmäßig `<X4>/X4DB/0/jco_server_settings.prop`) werden die Anzahl der *JCo Server*-Instanzen, Funktions-Handler-Prozesse und IDoc-Handler-Prozesse definiert. In der zentralen Konfigurationsdatei `<X4>/X4config.xml` muss für jede *JCo Server*-Instanz jeweils eine *JCo Server*-Konfigurationsdatei referenziert werden. Die Properties-Datei ist in folgende Blöcke unterteilt.

Allgemeine Konfiguration

Mit den folgenden Properties lässt sich der *JCo 2 Server* hinsichtlich der zulässigen Server-Instanzen und dem Umgang mit Verbindungen und Connection Pooling konfigurieren, siehe [Connection Pooling \(X4 SAP JCo 2\)](#).

```
# Number of server instances
server_instances_count = Anzahl der JCo Server-Instanzen
usePool                = Boolescher Wert
poolSize               = Zulässige Zahl der Verbindungen
```

Erläuterung:

- `server_instances_count`: Anzahl der *JCo Server*-Instanzen
- `usePool`: Connection Pooling aktivieren (*true*) oder deaktivieren (*false*), siehe [Connection Pooling \(X4 SAP JCo 2\)](#)
- `poolSize`: Anzahl der zulässigen Client-Verbindungen im Pool; Empfehlung: Einen Wert wählen, der mindestens der Summe der *JCo 2 Server*-Instanzen (identifiziert durch je eine *ProgID*) entspricht.

Funktions-Handler-Konfiguration

In den folgenden Properties werden die Funktions-Handler für RFC/BAPI-Aufrufe pro SAP-Host definiert. Die Konfiguration der Standard-Funktions-Handler-Prozesse ist optional und dient lediglich als Fallback, falls in den Funktionsbaustein-/Methodendefinitionen keine entsprechenden Handler-Prozesse zugeordnet wurden, siehe [Funktionsbaustein-/Methodendefinitionen \(X4 SAP JCo 2\)](#).

Commit- und Rollback-Handler können jedoch nur über diese Properties vereinbart werden.

```
# per jcosever instance settings
# this should always be set in the method definitions
jcosever.function.handler.userId = Benutzerordner in X4DB (optional)
jcosever.function.handler.process = Standard-Function-Handler-Prozess (optional)

jcosever.function.commit.userId = Benutzer-Repository für Commit-Prozess
jcosever.function.commit.process = Pfad zum X4-Prozess für Commit
jcosever.function.rollback.userId = Benutzer-Repository für Rollback-Prozess
jcosever.function.rollback.process = Pfad zum X4-Prozess für Rollback
```

Erläuterung:


- `jcosever.function.handler.userId`: Benutzer-Repository in X4DB (z. B. *1*), in dem der Funktionsaufruf-Handler-Prozess liegt
- `jcosever.function.handler.process`: optionale Pfadangabe zu einem X4-Prozess, der als Standard-Handler für Funktionsaufrufe dient
- `jcosever.function.commit.userId`: Benutzer-Repository in X4DB (z. B. *1*), in dem der *Commit*-Handler-Prozess für Funktionsaufrufe liegt
- `jcosever.function.commit.process`: Pfadangabe zu einem X4-Prozess, der als *Commit*-Handler für Funktionsaufrufe dient
- `jcosever.function.rollback.userId`: Benutzer-Repository in X4DB (z. B. *1*), in dem der *Rollback*-Handler-Prozess für Funktionsaufrufe liegt

- `jcosever.function.rollback.process`: Pfadangabe zu einem *X4*-Prozess, der als *Rollback*-Handler für Funktionsaufrufe dient

IDoc-Verarbeitung

In den folgenden Properties werden die Handler für eingehende IDocs definiert. Die Konfiguration des Standard-IDoc-Handler-Prozesses ist optional und dient lediglich als Fallback, falls in den Funktionsbaustein-Definitionen keine entsprechenden Handler-Prozesse zugeordnet wurden, siehe [Funktionsbaustein-/Methodendefinitionen \(X4 SAP JCo 2\)](#).

Pro *JCo 2 Server*-Instanz (identifiziert durch je eine *ProgID*) kann jeweils ein Commit- und Rollback-Handler über diese Properties vereinbart werden.

 Wenn keine Handler-Konfiguration gefunden wird, dann wird der entsprechende Commit bzw. Rollback übersprungen.

```
# this should always be set in the method definitions
jcosever.idoc.handler.userId    = Benutzerordner in X4DB (optional)
jcosever.idoc.handler.process   = Standard-IDoc-Handler-Prozess (optional)

# commit and rollback handlers for IDocs
jcosever.idoc.commit.userId    = Benutzer-Repository für Commit-Prozess
jcosever.idoc.commit.process   = Pfad zum X4-Prozess für Commit
jcosever.idoc.rollback.userId   = Benutzer-Repository für Rollback-Prozess
jcosever.idoc.rollback.process  = Pfad zum X4-Prozess für Rollback
```

Erläuterung:

- `jcosever.idoc.handler.userId`: Optionale Angabe des Benutzer-Repositorys in X4DB (z. B. *1*), in dem der IDoc-Handler-Prozess liegt (Fallback)
- `jcosever.idoc.handler.process`: Optionale Pfadangabe zu einem *X4*-Prozess, der als Standard-Handler für IDocs dient (Fallback)
- `jcosever.idoc.commit.userId`: Benutzer-Repository in X4DB (z. B. *1*), in dem der *Commit*-Handler-Prozess für IDocs liegt
- `jcosever.idoc.commit.process`: Pfadangabe zu einem *X4*-Prozess, der als *Commit*-Handler für IDocs dient
- `jcosever.idoc.rollback.userId`: Benutzer-Repository in X4DB (z. B. *1*), in dem der *Rollback*-Handler-Prozess für IDocs liegt
- `jcosever.idoc.rollback.process`: Pfadangabe zu einem *X4*-Prozess, der als *Rollback*-Handler für IDocs dient

Verbindungs-Einstellungen zum SAP-Host


In den folgenden Properties werden die Einstellungen für den Verbindungsaufbau zu einem SAP-Host für die jeweilige *JCo 2 Server*-Instanz definiert:


```
# connection String to SAP
jco.server.gwhost = Gateway-Host
jco.server.gwserv = Gateway-Service
jco.server.progid = Programm-ID
jco.server.trace = Boolescher Wert
jco.server.unicode = Boolescher Wert

# required for dynamic repository and metadata lookup
jco.client.client = Client-ID
jco.client.user = Benutzername
jco.client.passwd = Passwort
jco.client.lang = Client-Sprache
jco.client.sysnr = SAP-Systemnummer
jco.client.ashost = SAP-Applikations-Server
jco.client.trace = Boolescher Wert
```

Erläuterung:

- `jco.server.gwhost`: Gateway-Host
- `jco.server.gwserv`: Gateway-Service (normalerweise *sapgwNM*)
- `jco.server.progid`: Programm-ID des SAP-Hosts
- `jco.server.trace`: RFC Trace für den SAP-Server aktivieren (1) oder deaktivieren (0)
- `jco.server.unicode`: Im Unicode-Modus verbinden (1) oder nicht (0)
- `jco.client.client`: Client-ID (dreistelliger numerischer Wert)
- `jco.client.user`: Benutzername
- `jco.client.passwd`: Passwort
- `jco.client.lang`: Sprache des Clients
- `jco.client.sysnr`: SAP-Systemnummer
- `jco.client.ashost`: SAP-Applikations-Server
- `jco.client.trace`: RFC Trace aktivieren (1) oder deaktivieren (0)

 Eine vollständige Beschreibung aller (Verbindungs-)Parameter finden Sie in der *SAP JCo 2*-Dokumentation. Sämtliche Parameter werden vom *JCo 2 Server* unterstützt.

6.9.2.6.2.3 Funktionsbaustein-/Methodendefinitionen (X4 SAP JCo 2)

In einem Konfigurations-Dokument (standardmäßig `<X4>/X4DB/0/jco_method_def.xml`) kann für jeden Funktionsbaustein (Methode), die vom SAP-Host angesprochen wird, über den *JCo 2 Server* ein entsprechender X4-Prozess zur Weiterverarbeitung zugeordnet werden.

In der zentralen Konfigurationsdatei `<X4>/X4config.xml` muss für jede *JCo Server*-Instanz jeweils eine Funktionsbaustein-/Methodendefinition referenziert werden.

Erwartete XML-Struktur

In der Funktionsbaustein-/Methodendefinition können Sie innerhalb des Wurzel-Elements `<definitions>` beliebig viele Elemente `<method>` nach folgendem Muster anlegen:

```

<method>
  <name><!--Funktionsbaustein-/Methodenname--></name>
  <wait_for_result><!--Boolescher Wert--></wait_for_result>
  <class>de.softproject.integration.sapjcoserver.requesthandlers.WorkflowStarter</
class>
  <properties>
    <property Name="workflowname"><!--Pfad zum X4-Prozess--></property>
    <property Name="userID"><!--Benutzerordner in X4DB--></property>
    <property Name="raw"><!--Boolescher Wert--></property>
  </properties>
</method>


```

Für jede Funktionsbaustein-/Methodendefinition gilt:

- Element `<name>` enthält den Namen des durch SAP aufgerufenen Funktionsbaustein (Methode); Der Platzhalter `*` steht für alle anderen Funktionsbausteine, für die kein expliziter Funktionsbaustein definiert wurde.
- In Element `<wait_for_result>` können Sie konfigurieren, ob SAP auf die Antwort warten soll (`false`), oder ob der Funktionsbaustein-/Methodenaufruf asynchron erfolgt (`true`).
- Element `<class>` gibt die verwendete Klasse an, die zur Verarbeitung des Funktionsbausteins (bzw. der Methode) hinterlegt ist (normalerweise ist dies `de.softproject.integration.sapjcoserver.requesthandlers.WorkflowStarter` zum Starten von X4-Prozessen); in den Kindelementen von `<class>` geben Sie die zugehörigen Eigenschaften in jeweils einem Element `<property>` an.

Für die X4-Prozess-Starterklasse werden folgende Eigenschaften angegeben:

- `workflowname`: Pfad zum X4-Prozess, der die Verarbeitung des Funktionsbaustein-/Methodenaufrufs übernimmt
- `userID`: Repository-Benutzerordner, in dem der X4-Prozess liegt
- `raw`: IDoc im Raw-Format verarbeiten (`true`) oder (`false`)

 In älteren Funktionsbaustein-/Methodendefinitionen finden sich ggf. zusätzliche Angaben in einem Element `<type>` und in einem Element `<property Name="languageID">`. Diese besitzen keine Funktion und können daher weggelassen werden.

Beispiel

Im folgenden Beispiel wird für den Funktionsbaustein (Methode) `ORDERS05` der X4-Prozess `save_idoc.wrf` aufgerufen und es werden Daten im *IDoc Raw*-Format für die nachgeordnete Verarbeitung bereitgestellt.

```

<method>
  <name>ORDERS05</name>
  <wait_for_result>false</wait_for_result>
  <class>de.softproject.integration.sapjcoserver.requesthandlers.
    WorkflowStarter</class>
  <properties>
    <property Name="workflowname">Project/Folder/ProcessIDoc.wrf</property>
    <property Name="userID">1</property>
    <property Name="raw">true</property>
  </properties>
</method>

```

6.9.2.6.2.4 Connection Pooling (X4 SAP JCo 2)

Der *SAP JCo 2 Server* kann für jede Instanz einen Pool für Client-Verbindungen bereitstellen. Wenn kein Pool existiert, dann wird direkt eine *JCo Client*-Verbindung hergestellt.

Einstellungen zum Connection Pooling in der Server-Properties-Konfigurationsdatei, siehe [Properties für den JCo 2 Server](#):


- `usePool`: Connection Pooling aktivieren (*true*) oder deaktivieren (*false*)
- `poolSize`: Anzahl der zulässigen Client-Verbindungen im Pool; Empfehlung: Einen Wert wählen, der mindestens der Summe der *JCo 2 Server*-Instanzen (identifiziert durch je eine *ProgID*) entspricht.

Poolgröße und Anzahl SAP-Adapter

Erhöhen Sie den in `poolSize` angegebenen Wert um die Anzahl der in *X4*-Prozessen eingesetzten SAP-Adapter. Im [SAP Connector \(JCo 2\)](#) können Sie den zu verwendenden Pool über den Adapter-Parameter `x4.connectionPoolUID` (frei wählbar) festlegen.

6.9.2.6.2.5 JCo 2 Server-Konfiguration aktualisieren

Um geänderte Konfigurationen am *SAP JCo Server* zu aktualisieren, ist es nicht erforderlich, den kompletten *X4 Server* neuzustarten. Der *X4 Server* stellt hierzu eine JMX-MBean `de.softproject.X4.X4Management` mit folgenden drei Methoden bereit.

-  Bei *JBoss 4.2.3 GA* können Sie die JMX-Konsole über eine URL nach dem Muster `http://host:port/jmx-console/` im Browser aufrufen, z. B. <http://localhost:8080/jmx-console/>. Dort finden Sie die entsprechende MBean `de.softproject.X4.X4Management`.

<i>JCo Server</i> neu starten:	Methode <code>restartSAPJcoServer()</code> über Invoke aufrufen.
<i>JCo Server</i> herunterfahren:	Methode <code>stopSAPJcoServer()</code> über Invoke aufrufen.
<i>JCo Server</i> -Konfiguration auf Änderungen prüfen und diese einlesen:	Methode <code>reloadChangedSAPJcoServers()</code> über Invoke aufrufen.

6.9.2.7 SAP Listener (JCo 3)

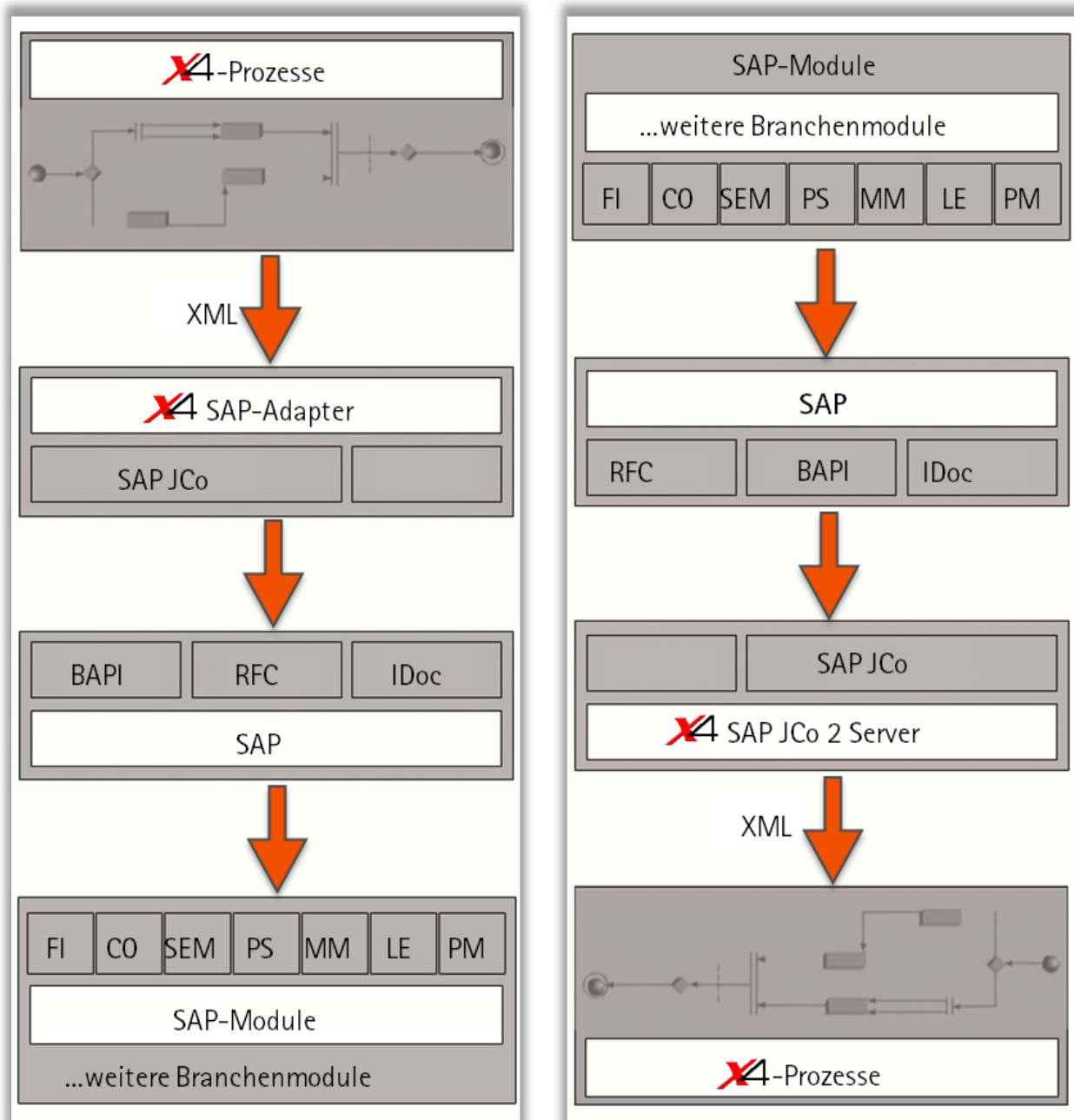
Der *SAP Listener (JCo 3)* ermöglicht die Verarbeitung von eingehenden Kommunikationsanfragen via BAPI/RFC und von IDocs eines SAP-Systems. Er stellt dabei automatisch ggf. unterbrochene Verbindungen wieder her.

Ehemalige Bezeichnung bis Version 4.5.1: *X4 SAP JCo 3 Listener*

Funktionen

Der *X4 Server* ermöglicht die bidirektionale Anbindung von *SAP R/3*-Systemen über den *SAP Java Connector* in Version 3.x (*JCo 3*). Dabei sind folgende Komponenten relevant:

- *SAP JCo 3 Server*: Ermöglicht die Verarbeitung von eingehenden Kommunikationsanfragen via BAPI/RFC und von IDocs eines SAP-Systems an den *X4 Server*. Der *SAP JCo 3 Server* stellt dabei automatisch ggf. unterbrochene Verbindungen wieder her.
- *SAP Connector (JCo 3)*: In technischen Prozessen kann dieser Funktions-Adapter einen beliebigen Remote Function Call (RFC) bzw. einen beliebigen BAPI-Funktionsbaustein in einem SAP-System aufrufen. Zudem können IDocs an *SAP R/3* versandt werden; siehe *Adapter und Bausteine, SAP Connector (JCo 3)*.
- *SAP Explorer (JCo 3)*: Dieser ermöglicht während der Prozess-Entwicklung das Erzeugen von Strukturbeschreibungen von IDocs und BAPIs. Zudem listet er BAPIs oder IDocs oder erzeugt leere IDoc- und BAPI-Strukturen und gibt je nach Operation ein entsprechendes XML-Dokument aus; siehe *Adapter und Bausteine, SAP Explorer (JCo 3)*.



Multi-SAP-Fähigkeit

Über den *JCo 3 Server* können mehrere SAP-Systeme über eine jeweils separate *JCo 3 Server*-Instanz angebunden werden. Zum Verwenden des *JCo 3*-Servers ist eine entsprechende Konfiguration erforderlich:

- X4-seitige Konfiguration in `x4config.xml`; siehe [SAP JCo 3 Server-Konfiguration \(X4config.xml\)](#)
- Definition von Verbindungs-Eigenschaften; siehe [JCo3 Properties](#)
- Zuordnung der Funktionsbaustein-Aufrufe zu technischen Prozessen und Commit/Rollback-Handler; siehe [Funktionsbaustein-/Methodendefinitionen \(X4 SAP JCo 3\)](#)

Identifikation von JCo 3-Servern über ProgID

Jede *JCo 3 Server*-Instanz bzw. jedes an SAP angebundene System kommuniziert über eine eindeutige *ProgID*, die in der Server-Properties-Datei, standardmäßig `jco_server_settings.prop`, konfiguriert wird und beim Start der Server-Instanz dem SAP-System bekannt gemacht wird.

Pro SAP-System können mehrere ProgIDs gleichzeitig verwendet werden. Über die ProgID werden IDocs und RFC/BAPIs werden an das jeweilige Ziel-System eindeutig "adressiert".

6.9.2.7.1 Installation von X4 SAP JCo 3

Welche Schritte zur Installation von *X4 SAP JCo 3* erforderlich sind



Beispiel

Als Beispiel wird eine Installation für den *JBoss* Applikations-Server auf Basis eines 32-Bit-Windows-Systems beschrieben

6.9.2.7.1.1 Voraussetzungen

Zur Installation eines *SAP JCo 3 Servers* im *X4 Server* werden *SAP*-Bibliotheken benötigt, die z.T. betriebssystem- und architekturabhängig sind. Diese Bibliotheken werden auch vom den *SAP Connector (JCo 3)* und dem *SAP Explorer (JCo 3)* benötigt.

Als registrierter SAP-Partner die erforderlichen Bibliotheken sowie weiterführende Dokumentation über das *SAPNet* von <http://service.sap.com/connectors> herunterladen.



Wählen Sie hierbei zueinander kompatible Versionsnummern der Bibliotheken (diese sind zudem betriebssystem- und architekturabhängig).

Achten Sie darauf, dass das verwendete Betriebssystem kompatibel mit der eingesetzten JDK-Version und der Version Ihres SAP-Systems ist. Folgende Matrix zeigt den Zusammenhang dieser drei Variablen.

		OS		SAP	
		32 bit	64 bit	32 bit	64 bit
JDK	32 bit	x	x	x	-
	64 bit	-	x	-	x

6.9.2.7.1.2 Bibliotheken installieren

- Native Bibliotheken (je nach System) in den entsprechenden Ordner des Applikations-Servers für native Binär-Abhängigkeiten oder in das `bin`-Verzeichnis von Java kopieren.
Beispiel: Unter Windows 32-Bit: `librfc32.dll`, `msvcr71.dll`, `msvcpr71.dll` und `sapjcorfc.dll` nach `C:\Windows\System32` oder nach `C:\X4\Server\jdk1.7\bin` kopieren.
- Folgende Java-Bibliotheken in den Classpath des Applikations-Servers kopieren:
 - `sapidoc.jar`
 - `sapjco.jar`
 - `sapidocjco.jar`

Beispiel JBoss 4.2.3.GA: Diese Bibliotheken nach `C:\X4\JBoss4.2.3.GA\server\default\lib` kopieren.

Beispiel JBoss AS 7.1.1: Diese Bibliotheken unter `C:\X4\Server\jboss-as-7.1.1.Final\standalone\deployments` in das `lib`-Verzeichnis von `X4.ear` kopieren.

3. *X4 Server* neu starten.

6.9.2.7.1.3 Installation prüfen

Verbindung prüfen über den *SAP Explorer (JCo 3)* (analog zu *X4 Experience*-Beispiel *X4Experience/AdapterSuites/ X4SAPSuite/SAP (JCo2) Explorer (Prototype)/ Processes/ CreateEmptyBapi.wrf*):

- Sicherstellen, dass die Konfiguration des *SAP JCo 3 Server* vollständig ist; siehe Konfiguration von *X4 SAP JCo 3*
- Adapterparameter `SystemType` auf `3` setzen (für eine Verbindung zu *SAP R/3*).
- Sicherstellen, dass Adapteroperation `Create Empty Bapi` gewählt ist.
- Prozess mit dem Input-XML `<Bapiname>BAPI 's name</Bapiname>` ausführen und Ergebnis-XML prüfen.

Auf dieser Seite:

Siehe auch:


- [Konfiguration von X4 SAP JCo 3](#)

6.9.2.7.2 Konfiguration von X4 SAP JCo 3

Welche Konfiguration *X4*-seitig angepasst werden muss, um den *JCo 3 Server* zu starten, von außen erreichbar zu machen und um den eingehenden RFC/BAPIs und IDocs entsprechende Handler-Prozesse im *X4 Server* zuzuordnen

6.9.2.7.2.1 SAP JCo 3 Server-Konfiguration (X4config.xml)

Wie der *X4 SAP JCo 3 Server* in `X4config.xml` konfiguriert wird

-  Die zentrale Konfigurationsdatei `<X4>\X4config.xml` enthält für den *X4 SAP JCo 3 Server* keine Konfigurationsoptionen mehr, sondern ggf. lediglich für eine separate *X4 SAP JCo 2*-Installation. Stattdessen befindet sich die *JCo 3*-Konfiguration in Properties-Dateien, die über die Adapter und/oder über die Listener-Konfiguration aufgerufen werden.

Siehe auch:

- [JCo 3 Properties](#)
- [JCo 3 Listener-Konfiguration](#)
- [Konfiguration über X4config.xml](#)

6.9.2.7.2.2 JCo 3 Listener-Konfiguration

Wie Sie eingehenden *JCo 3*-Verbindungen Properties und Methoden zuweisen

Die Konfiguration eingehender *JCo* 3-Verbindungen von einem oder mehreren SAP-Systemen befindet sich in Datei `sapListener.xml` in `<X4>\X4DB\0`. Dort lässt sich pro SAP-System ein *JCo* 3-Listener definieren, der wiederum Server-Properties und Methodendefinitionen (in einem spezifischen XML-Dokument) referenziert.

XML-Struktur

Zur Beschreibung der Struktur von `sapListener.xml` ist zudem ein XML-Schema-Definition `sapListener.xsd` im selben Ordner verfügbar. Diese XSD ist im Wurzel-Element `<listeners>` der Listener-Konfiguration angegeben.

```
<listener>
  <serverPropertiesURL>Pfad zur Properties-Datei</serverPropertiesURL>
  <methodDefinitionsURL>Pfad zur Methodendefinition</methodDefinitionsURL>
  <replaceEscapedControlChars>false</replaceEscapedControlChars><!-- true | false --
>
</listener>
```

Jedes Element `<listener>` mit einer *JCo* 3-Listener enthält folgende Kindelemente:

- `<serverPropertiesURL>`: Gültige URL zu einer Properties-Textdatei, die sämtliche Verbindungseinstellungen zum SAP-Host enthält
- `<methodDefinitionsURL>`: Gültige URL zu einem XML-Dokument, in dem sämtliche Methodendefinitionen bzw. Funktionsbausteineaufrufe und Transaktions-Handling-Prozesse definiert sind; siehe Funktionsbaustein-/Methodendefinitionen
- `<replaceEscapedControlChars>`: Maskierte Steuerzeichen (z.B.: ``) in den von der JCo-Bibliothek zurückgegebenen XML-Strings durch das Ersetzungszeichen (🔹) austauschen. Mögliche Werte sind `true` oder `false`

Beispiel

Im folgenden Beispiel werden zur Anbindung von zwei SAP-Hosts jeweils ein Listener mit den separaten Properties- und Methodendefinitionen bestimmt. Mit `replaceEscapedControlChars true` werden zudem in den von der JCo-Bibliothek zurückgegebenen XML-Strings alle maskierten Steuerzeichen (z.B.: ``) durch das Ersetzungszeichen (🔹) ausgetauscht.


```
<listeners xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="saplistener.xsd">
  <listener>
    <serverPropertiesURL>file:///C:/X4/X4DB/0/host1.prop</serverPropertiesURL>
    <methodDefinitionsURL>file:///C:/X4/X4DB/0/host1_method_def.xml
    </methodDefinitionsURL>
    <replaceEscapedControlChars>true</replaceEscapedControlChars>
  </listener>
  <listener>
    <serverPropertiesURL>file:///C:/X4/X4DB/0/host2.prop</serverPropertiesURL>
    <methodDefinitionsURL>file:///C:/X4/X4DB/0/host2_method_def.xml
    </methodDefinitionsURL>
    <replaceEscapedControlChars>true</replaceEscapedControlChars>
  </listener>
</listeners>
```

Auf dieser Seite:

Siehe auch:

- [JCo 3 Properties](#)
- [Funktionsbaustein-/Methodendefinitionen \(X4 SAP JCo 3\)](#)

6.9.2.7.2.3 JCo 3 Properties

Wie Sie die Verbindungsparameter zur Kommunikation zwischen Client und Host hinterlegen
Die Verbindungsdaten zur Kommunikation zwischen Client und Host können Sie global in einer Properties-Datei definieren. Für eingehende Verbindungen wird eine solche Properties-Datei pro *X4 SAP JCo 3*-Listener in `<X4>/X4DB/0/saplistener.xml` angegeben.

Für ausgehende Verbindungen über den *SAP Connector (JCo 3)* und den *SAP Explorer (JCo 3)* geben Sie den Pfad zu einer solchen Properties-Datei jeweils in Adapter-Parameter `PropertyFile` an.

Im Adapter gesetzte Parameterwerte überschreiben zur Laufzeit die Konfiguration in der Properties-Datei. Alle Properties, die in der Properties-Datei gesetzt werden (auch zusätzliche Properties), werden vom *X4 SAP JCo 3*-System übernommen.

Properties-Struktur

```
# connection String to SAP
jco.server.gwhost = Gateway-Host
jco.server.gwserv = Gateway-Service (normalerweise sapgwnn)
jco.server.progid = Programm-ID
jco.server.trace = RFC Trace aktivieren (1) oder deaktivieren (0)
jco.server.unicode = Im Unicode-Modus verbinden (1) oder nicht (0)

# equivalent to (dynamic) adapter parameter 'User'
jco.client.user = Benutzername

# equivalent to (dynamic) adapter parameter 'Password'
jco.client.passwd = Passwort

# equivalent to (dynamic) adapter parameter 'Client'
jco.client.client = SAP-Mandant (dreistelliger numerischer Wert)

# equivalent to (dynamic) adapter parameter 'Language'
jco.client.lang = Anmeldesprache

# equivalent to (dynamic) adapter parameter 'SystemNumber'
jco.client.sysnr = SAP-System-Nummer

# equivalent to (dynamic) adapter parameter 'RouterString'
jco.client.ashost = Router-String

# (optional) type of SAP system 2 = R/2, 3 = R/3, E = External

# equivalent to (dynamic) adapter parameter 'SystemType'
jco.client.type = Art des entfernten Hosts

# set to true if transactions are used
# equivalent to (dynamic) adapter parameter 'TransactionEnabled'
x4.transactionEnabled = Transaktionale Verarbeitung aktivieren (true)
oder deaktivieren (false)

# Should be set for each initiated transaction
# equivalent to (dynamic) adapter parameter 'TransactionId'
x4.transactionID = Transaktions-ID (für jede initiierte Transaktion)
```

❗ Eine vollständige Beschreibung aller (Verbindungs-)Parameter finden Sie in der *SAP JCo 3*-Dokumentation. Sämtliche Parameter werden vom *X4 SAP JCo 3 Server* unterstützt.

Beispiel

Im Folgenden ein Beispiel für eine Properties-Datei mit einigen gesetzten Werten:

```

# equivalent to (dynamic) adapter parameter 'User'
jco.client.user = <user>

# equivalent to (dynamic) adapter parameter 'Password'
jco.client.passwd = <pwd>

# equivalent to (dynamic) adapter parameter 'Client'
jco.client.client = 800

# equivalent to (dynamic) adapter parameter 'Language'
jco.client.lang = DE

# equivalent to (dynamic) adapter parameter 'SystemNumber'
jco.client.sysnr = 00

# equivalent to (dynamic) adapter parameter 'RouterString'
jco.client.ashost =

# (optional) type of SAP system 2 = R/2, 3 = R/3, E = External
# equivalent to (dynamic) adapter parameter 'SystemType'
jco.client.type = 3

# set to true if transactions are used
# equivalent to (dynamic) adapter parameter 'TransactionEnabled'
x4.transactionEnabled = false

# Should be set for each initiated transaction
# equivalent to (dynamic) adapter parameter 'TransactionId'
x4.transactionID = 1

```

Auf dieser Seite:

Siehe auch:

- [JCo 3 Listener-Konfiguration](#)
- [Funktionsbaustein-/Methodendefinitionen \(X4 SAP JCo 3\)](#)

6.9.2.7.2.4 Funktionsbaustein-/Methodendefinitionen (X4 SAP JCo 3)

Wie Sie im SAP-Listener zu RFC/BAPis jeweils einen verarbeitenden X4-Prozess zuordnen können
In einem spezifischen XML-Dokument (z. B. <X4>/X4DB/0/jco_method_def.xml) können Sie für jeden Funktionsbaustein (Methode), der vom SAP-Host angesprochen wird, über den *JCo 3 Server* einen entsprechenden X4-Prozess zur Weiterverarbeitung zuordnen.

XML-Struktur für die SAP-Listener-Konfiguration

Die SAP-Listener-Konfigurationsdatei <X4><X4>/X4DB/0/saplistener.xml referenziert für jede *JCo 3 Server*-Instanz jeweils ein XML-Dokument mit Funktionsbaustein-/Methodendefinitionen. Darin können Sie innerhalb des Wurzel-Elements <definitions> beliebig viele Elemente <method> nach folgendem Muster anlegen:

```

<method>
  <name>Funktionsbaustein-/Methodenname</name>
  <wait_for_result>Boolescher Wert</wait_for_result>
  <class>de.softproject.integration.sapjcoserver.requesthandlers.
    WorkflowStarter</class>
  <properties>
    <property name="workflowname">Pfad zum X4-Prozess</property>
    <property name="userID">Benutzerordner in X4DB</property>
    <property name="raw">Boolescher Wert</property>
  </properties>
</method>

```

Für jede Funktionsbaustein-/Methodendefinition gilt:


- Element `<name>` enthält den Namen des durch SAP aufgerufenen Funktionsbaustein (Methode); Der Platzhalter `*` steht für alle anderen Funktionsbausteine, für die kein expliziter Funktionsbaustein definiert wurde. Zudem lassen sich hier Transaktions-Handler definieren (siehe unten).
- In Element `<wait_for_result>` können Sie konfigurieren, ob SAP auf die Antwort warten soll (`false`), oder ob der Funktionsbaustein-/Methodenaufruf asynchron erfolgt (`true`).
- Element `<class>` gibt die verwendete Klasse an, die zur Verarbeitung des Funktionsbausteins (bzw. der Methode) hinterlegt ist (normalerweise ist dies `de.softproject.integration.sapjcoserver.requesthandlers.WorkflowStarter` zum Starten von X4-Prozessen); in den Kindelementen von `<class>` geben Sie die zugehörigen Eigenschaften in jeweils einem Element `<property>` an.

Für die X4-Prozess-Starterklasse werden folgende Eigenschaften angegeben:

- `workflowname`: Pfad zum X4-Prozess, der die Verarbeitung des Funktionsbaustein-/Methodenaufrufs übernimmt
- `userID`: Repository-Benutzerordner, in dem der X4-Prozess liegt
- `raw`: IDoc im Raw-Format verarbeiten (`true`) oder (`false`)

Kompatibilitätshinweis

In älteren Funktionsbaustein-/Methodendefinitionen finden sich ggf. zusätzliche Angaben in einem Element `<type>` und in einem Element `<property name="languageID">`. Diese besitzen keine Funktion und können daher weggelassen werden.

 Beachten Sie bei der Migration auf JCo 3, dass unter JCo 2 das Attribut `name` innerhalb des Elementes `<property>` groß geschrieben wurde. Unter JCo 3 wird dieses klein geschrieben.

Transaktions-Handler für IDocs und BAPI-Aufrufe

In X4 SAP JCo 3 wurde das Transaktions-Handling in die Methodendefinition integriert. Zur Verarbeitung von eingehenden IDocs und BAPI-Aufrufen können Sie wie folgt einen *Commit*-Handler und einen *Rollback*-Handler definieren und jeweils einen X4-Prozess zuordnen, indem Sie in jeweils einer der Methodendefinition innerhalb von Element `<name>` folgende Werte hinterlegen:

- *IDOC+COMMIT*: IDoc-Transaktion abschließen (*Commit*)
- *IDOC+ROLLBACK*: IDoc-Transaktion abbrechen und zurücksetzen (*Rollback*)

- *BAPI+COMMIT*: BAPI-Transaktion abschließen (*Commit*)
- *BAPI+ROLLBACK*: BAPI-Transaktion abbrechen und zurücksetzen (*Rollback*)

Globaler Handler

Falls kein spezifischer Handler zutrifft, können Sie einen globalen Handler definieren, indem Sie in der Methodendefinition in Element `<name>` als Wert einen Stern (*) eintragen.

Beispiel

Im folgenden Beispiel wird für den Funktionsbaustein (Methode) `ORDERS05` der *X4*-Prozess `save_idoc.wrf` aufgerufen und es werden Daten im *IDoc Raw*-Format für die nachgeordnete Verarbeitung bereitgestellt. Zudem wird ein BAPI-Commit-Handler sowie ein globaler Handler ohne zugeordneten *X4*-Prozess definiert.

```
<method>
  <name>ORDERS05</name>
  <wait_for_result>false</wait_for_result>
  <class>de.softproject.integration.sapjcoserver.requesthandlers.
    WorkflowStarter</class>
  <properties>
    <property Name="workflowname">Project/Folder/ProcessIDoc.wrf</property>
    <property Name="userID">1</property>
    <property Name="raw">true</property>
  </properties>
</method>
<method>
  <name>BAPI+COMMIT</name>
  <type>0</type>
  <wait_for_result>false</wait_for_result>
  <class>de.softproject.integration.sapjco3server.requesthandlers.
    WorkflowStarter</class>
  <properties>
    <property name="workflowname">Project/Folder/BAPI_Commit.wrf</property>
    <property name="userID">1</property>
  </properties>
</method>
<method>
  <name>*</name>
  <type>1</type>
  <wait_for_result>false</wait_for_result>
  <class>de.softproject.integration.sapjco3server.requesthandlers.
    WorkflowStarter</class>
  <properties>
  </properties>
</method>
```

Auf dieser Seite:

6.10 File Processing

6.10.1 ASCII to XML

6.10.1.1 ASCII to XML Converter



ASCII to XML Converter: Liest ein Textdokument ein und gibt ein XML-Dokument mit unveränderter Zeichenkodierung aus, das den Inhalt des Textdokuments mit einem CDATA-Bereich umschließt.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>WrapData</i> : Text-Dokument in ein XML-Dokument mit CDATA-Bereich umwandeln
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.ASCII2XML</i> : Hauptklasse (Standard)
----------------	---

6.10.1.2 ASCII Writer



ASCII Writer: Konvertiert ein XML-Dokument gemäß den Regeln einer Deskriptordatei in ein strukturiertes Textdokument mit unveränderter Zeichenkodierung.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>Execute</i> : ASCII-Dokument schreiben, gemäß den Regeln einer in Parameter <i>DescFile</i> festgelegten ASCII-Deskriptor-Datei
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.asciwriter.ASCIIWriter</i> : Hauptklasse (Standard)
----------------	--

<i>DescFile</i>	<p>Name der Deskriptordatei, die das Textdokument in seiner Datenstruktur beschreibt. Die Deskriptordatei sollte im Benutzerordner des X4 Repositorys abgelegt sein (z. B. <code>x4DB/1/Project/Descr.ad</code>)</p> <p>Mögliche Werte: Pfad und Dateiname (z. B. <code>/Project/Descr.ad</code>)</p>
-----------------	--

Statuswerte

-1	Beim Ausführen des Adapters trat ein Fehler auf (siehe <i>X4 Server</i> -Protokoll).
0	Der Adapter wurde erfolgreich ausgeführt.

6.10.1.3 Text to Lines Converter

Text to Lines Converter: Liest alle Zeilen eines Textdokuments oder bestimmte Zeilen ein und gibt ein XML-Dokument aus, das für jede Zeile ein XML-Element enthält.

Als Input verwendet der Adapter ein Textdokument aus dem vorhergehenden Prozess-Schritt. Mit Operation *Iterate* kann der Adapter das Input-Dokument schrittweise durchlaufen und für jede Iteration im technischen Prozess wird ein Ergebnis-Dokument ausgegeben.

Beachten Sie:

- Wenn Sie innerhalb eines Prozess-Kontextes mehrere *Text to Lines Converter* definieren, dann vergeben Sie in Parameter *id* eine eindeutige ID für den jeweiligen Adapter, z.B. *1001*, *1002* etc. So wird für das Selektieren im Iterator-Modus und für das Zurücksetzen eines Iterators gewährleistet, dass nur der *Text to Lines Converter Adapter* mit der entsprechenden ID angesprochen wird (siehe Operation *Clear*).
- Wenn Sie in Parameter *id* den Standardwert *0* belassen, dann wird die Nummer des Prozess-Schritts (*Action ID*) im aktuellen Prozess als ID verwendet. Diese ID ist möglicherweise nicht eindeutig, wenn sich in einem Subprozess ein *Text to Lines Converter* mit der gleichen Action-ID befindet.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Parse</i>: Zeilen des Textdokuments einlesen • <i>Iterate</i>: Iterator-Modus aktivieren und Textdokument zeilenweise einlesen • <i>Clear</i>: Iterator eines <i>Text to Lines Converter Adapters</i> zurücksetzen, der dieselbe ID besitzt wie der gewünschte <i>Text to Lines Converter Adapter</i> mit der Operation <i>Iterate</i> (siehe Parameter <i>id</i>)
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.text2lines.TextToLinesAdapter</i>: Hauptklasse (Standard)</p>
<i>start</i>	<p>Erste Zeile, die eingelesen werden soll (Zählweise beginnt bei 0)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl • \emptyset: Ab einschließlich der ersten Zeile einlesen (Standard)
<i>count</i>	<p>Anzahl der Zeilen, die eingelesen werden sollen (bei Operation <i>Iterate</i> pro Durchgang)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • \emptyset: Keine Beschränkung, d. h. alle Zeilen auf einmal einlesen (Standard)

<i>id</i>	<p>Eindeutige ID des <i>Text to Lines Converter Adapter</i>-Bausteins innerhalb des Prozess-Kontextes bzw. der Prozess-Instanz (wenn Sie mehrere voneinander unabhängig iterierende <i>Text to Lines Converter Adapter</i> in demselben Prozess-Kontext verwenden möchten)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • \emptyset: Aktuelle Action-ID (Nummer des Prozess-Schritts) als ID verwenden; Beachten Sie hierbei, dass ein <i>Text to Lines Converter Adapter</i> in einem Subprozess die gleiche Action-ID besitzen könnte. (Standard) • Beliebige ganze Zahl außer \emptyset, um den jeweiligen <i>Text to Lines Converter Adapter</i>-Baustein innerhalb des Prozess-Kontextes eindeutig zu kennzeichnen; Empfehlung: einen Wert >1000 verwenden, um nicht zufällig eine bereits bestehende Action-ID zu "erwischen"
-----------	---

Ausgabe

Dieser Adapter gibt ein UTF-8-kodiertes XML-Dokument mit folgender Struktur aus. Für jede Zeile wird ein XML-Element `<Line/>` erzeugt, das die Zeilennummer und den Inhalt der jeweiligen Zeile enthält:

```
<Text>
  <Line lineNo="Zeilennummer">!--Inhalt der Zeile--></Line>
</Text>
```

Statuswerte

-1	Fehler bei der Ausführung des Adapters (Details siehe Server-Log)
0	Der Adapter gibt keine Daten (mehr) aus. Dieser Status kann bei Operation <i>Iterate</i> als Abbruchbedingung einer Schleife im technischen Prozess ausgewertet werden und gibt an, dass keine weiteren Datensätze verfügbar sind.
1	Der Adapter gibt ein Ergebnis aus. Dieser Status kann bei Operation <i>Iterate</i> als Schleifeneintrittsbedingung im technischen Prozess ausgewertet werden und gibt an, dass ein weiterer Datensatz verfügbar ist.

6.10.1.4 Text to Lines Converter from URL



Text to Lines Converter from URL: Liest Zeilen eines Textdokuments von einer beliebigen gültigen URL ein und gibt ein XML-Dokument aus, das die gewünschten Zeilen enthält. Mit der Operation *Iterate* kann der Adapter das Input-Dokument schrittweise durchlaufen und für jede Iteration im technischen Prozess wird ein Ergebnis-Dokument ausgegeben.

- i** Wenn Sie innerhalb eines Prozess-Kontextes mehrere *Text to Lines Converter from URL* definieren, dann vergeben Sie in Parameter *id* eine eindeutige ID für den jeweiligen Adapter, z.B. *1001*, *1002* etc. So wird für das Selektieren im Iterator-Modus und für das Zurücksetzen eines Iterators gewährleistet, dass nur der entsprechende Adapter angesprochen wird (siehe Operation *Clear*).
- Wenn Sie in Parameter *id* den Standardwert *0* belassen, dann wird die Nummer des Prozess-Schritts (*Action ID*) im aktuellen Prozess als ID verwendet. Diese ID ist möglicherweise nicht eindeutig, wenn sich in einem Subprozess ein *Text to Lines Converter from URL* mit der gleichen Action-ID befindet.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>Parse</i>: Gesamtes Textdokument einlesen • <i>Iterate</i>: Iterator-Modus aktivieren und Textdokument zeilenweise einlesen • <i>Clear</i>: Iterator eines <i>Text to Lines Converter from URL Adapters</i> zurücksetzen, der dieselbe ID besitzt wie der gewünschte <i>Text to Lines Converter from URL</i> mit der Operation <i>Parse</i> (siehe Parameter <i>id</i>)
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.text2lines.TextToLinesUrlAdapter</i> : Hauptklasse (Standard)
<i>start</i>	Erste Zeile, die eingelesen werden soll (Zählweise beginnt bei 0) Mögliche Werte: <ul style="list-style-type: none"> • Beliebige ganze Zahl • <i>0</i>: Ab einschließlich der ersten Zeile einlesen (Standard)
<i>count</i>	Anzahl der Zeilen, die eingelesen werden sollen (bei Operation <i>Iterate</i> pro Durchgang) Mögliche Werte: <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • <i>0</i>: Keine Beschränkung, d. h. alle Zeilen auf einmal einlesen (Standard)

<i>id</i>	<p>Eindeutige ID des <i>Text to Lines Converter from URL</i>-Bausteins innerhalb des Prozess-Kontextes bzw. der Prozess-Instanz (wenn Sie mehrere voneinander unabhängig iterierende <i>Text to Lines Converter from URL</i> in demselben Prozess-Kontext verwenden möchten)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>0</i>: Aktuelle Action-ID (Nummer des Prozess-Schritts) als ID verwenden; Beachten Sie hierbei, dass ein <i>Text to Lines Converter from URL</i> in einem Subprozess die gleiche Action-ID besitzen könnte. (Standard) • Beliebige ganze Zahl außer <i>0</i>, um den jeweiligen <i>Text to Lines Converter from URL</i>-Baustein innerhalb des Prozess-Kontextes eindeutig zu kennzeichnen; Empfehlung: einen Wert <i>>1000</i> verwenden, um nicht zufällig eine bereits bestehende Action-ID zu "erwischen"
<i>url</i>	<p>URL zum einzulesenden Textdokument</p> <p>Mögliche Werte: Beliebige gültige URL (z.B. <i>xstore://Project/Folder/File.txt</i>)</p>
<i>encoding</i>	<p>Zeichenkodierung des Input-Textdokuments</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige gültige Zeichenkodierung (z.B. <i>UTF-8</i>) • (<i>leer</i>): Standard-Zeichenkodierung des Systems verwenden

Ausgabe

Dieser Adapter gibt ein UTF-8-kodiertes XML-Dokument mit folgender Struktur aus. Für jede Zeile wird ein XML-Element `<Line/>` erzeugt, das die Zeilennummer und den Inhalt der jeweiligen Zeile enthält:

```
<Text>
  <Line lineNo="Zeilennummer"><!--Inhalt der Zeile--></Line>
</Text>
```

Beispiel

Das Beispiel `TextToLinesConverterFromURL.wrf` für den Iterator-Modus sowie für den einfachen Einlesemodus finden Sie im Demosystem *X4 Experience*.

Statuswerte

<i>-1</i>	Fehler bei der Ausführung des Adapters (Details siehe Server-Log)
<i>0</i>	Der Adapter gibt keine Daten (mehr) aus. Dieser Status kann im Iterator-Modus als Abbruchbedingung einer Schleife im technischen Prozess ausgewertet werden und gibt an, dass keine weiteren Datensätze verfügbar sind.

1	Der Adapter gibt ein Ergebnisdokument aus. Dieser Status kann im Iterator-Modus als Schleifeneintrittsbedingung im technischen Prozess ausgewertet werden und gibt an, dass ein weiterer Datensatz verfügbar ist.
---	---

6.10.2 Base64 Batch Decoder



Base64 Batch Decoder: Dekodiert Base64-kodierte Daten aus einer adapterspezifischen XML-Struktur, teilt das Ergebnis in separate Dateien auf und speichert diese in einem Ordner.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Decode</i> : Daten aus dem XML-Dokument in Base64-kodierte Dateien schreiben
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.docadapter.DocAdapter</i> : Hauptklasse (Standard)
<i>directory</i>	Zielordner, in den die dekodierten Dateien gespeichert werden sollen Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>%USER_COLLECTION%\Project\batch\</i>)

Input

Dieser Adapter erwartet eine adapterspezifische Input-XML-Struktur. Diese enthält für jede Datei, die dekodiert werden soll, den Dateinamen und den Base64-kodierten Dateiinhalt. Jedes Element `<document/>` steht hierbei für eine zu dekodierende Datei.

```
<data>
  <documents>
    <document>
      <filename><!--Dateiname--></filename>
      <content><!--Base64-kodierte Daten--></content>
    </document>
  </documents>
</data>
```

6.10.3 Base64 Converter



Base64 Converter: Enkodiert oder dekodiert Daten in Base64-Kodierung. Beim Dekodieren unterscheidet der Adapter anhand des MIME-Typs: ist der MIME-Typ `text/xml` oder `application/`

xml, dann wird das Input-Dokument als XML-Dokument interpretiert und der Textinhalt des Wurzel-Elements dekodiert, ansonsten wird das komplette Dokument als Base64-kodierte Zeichenkette interpretiert.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none">• <i>Encode</i>: Daten Base64-encodieren• <i>Decode</i>: Daten in Base64-Kodierung dekodieren (abhängig vom MIME-Typ)
------------------	--

Parameter

Adapter	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.Base64Adapter</i> : Hauptklasse (Standard)
chunk	Beim Encodieren: Ausgabe in Blöcke mit einer Länge von 76 Zeichen unterteilen Mögliche Werte: <ul style="list-style-type: none">• <i>ja</i>: Zeichenkette in Zeilenblöcke unterteilen• <i>nein</i>: Zeichenkette nicht unterteilen (Standard)
encoding	Verwendete Zeichenkodierung Mögliche Werte: Zeichenkette mit der verwendeten Zeichenkodierung, z. B. UTF-8 (Standard), windows-1251, ISO 8859-1, ...

6.10.4 Charset Changer



Charset Changer: Ändert die Zeichenkodierung eines Dokuments.



Die Zeichen des Input-Dokuments werden nicht in den Ziel-Zeichenkodierung konvertiert! Hierfür den Adapter *Encoding Converter* verwenden, siehe [Encoding Converter](#).

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>Execute</i> : Zeichensatz der Datei ändern
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.CharsetChanger</i> : Hauptklasse (Standard)
<i>InCharset</i>	Zeichensatz des Input-Dokuments Mögliche Werte: <ul style="list-style-type: none"> • Beliebige in Java bekannte Zeichenkodierung (z. B. <i>Windows-1252</i>) • <i>UTF-8</i>: Unicode-Zeichenkodierung UTF-8 (Standard)
<i>OutCharset</i>	Zeichensatz des Ausgabe-Dokuments Mögliche Werte: <ul style="list-style-type: none"> • Beliebige in Java bekannte Zeichenkodierung (z. B. <i>ISO-8859-1</i>) • <i>UTF-8</i>: Unicode-Zeichenkodierung UTF-8

6.10.5 CSV

6.10.5.1 CSV Converter



CSV Converter: Liest ein CSV-Textdokument (*comma separated values*) aus dem vorhergehenden Prozess-Schritt ein und gibt für die angegebenen Datensätze ein XML-Dokument in UTF-8-Zeichenkodierung aus, das ggf. die Spaltenbezeichnungen als Elementnamen verwendet.

Zeilen vor Beginn der CSV-Datenzeilen sowie Kommentarzeilen können ausgelassen werden. Sowohl CR (*carriage return*) als auch CRLF (*carriage return line feed*) werden als Zeilenumbrüche interpretiert. Mit Operation *Iterate* kann der Adapter das Input-Dokument schrittweise durchlaufen und für jede Iteration im X4-Prozess wird ein Ergebnis-Dokument ausgegeben.


Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Parse</i>: CSV-Textdokument in einem Schritt einlesen • <i>Iterate</i>: CSV-Textdokument schrittweise einlesen • <i>Clear</i>: Iterator eines <i>CSV Converters</i> zurücksetzen, der dieselbe ID besitzt wie der gewünschte <i>CSV Converter</i> mit der Operation <i>Iterate</i> (siehe Parameter <i>id</i>)
------------------	--

Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.csv.CsvAdapter</i>: Hauptklasse (Standard)</p>
delimiter	<p>Das Trennzeichen zwischen den einzelnen Datenfeldern einer Zeile im CSV-Textdokument</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Ein beliebiges Zeichen (keine Escape-Sequenzen wie <code>\t</code>) • <code>;</code>: Semikolon (Standard)
rowName	<p>Elementname im ausgegebenen XML-Dokument, das jede Datenzeile umschließt</p> <p>Mögliche Werte: Beliebiger gültiger XML-Elementname (ohne Namespace)</p>
columnNamesInFirstRow	<p>Datenfelder der ersten Zeile in der CSV-Textdatei als Spaltenbezeichnungen interpretieren und Datenfeld-Elemente im XML-Dokument entsprechend benennen, falls es sich um gültige XML-Elementnamen handelt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Die erste Zeile im CSV-Textdokument enthält Spaltenbezeichnungen, die sich als gültige XML-Elementnamen (ohne Namespace-Angaben) eignen. • <i>nein</i>: CSV-Textdokument beginnt direkt mit Datenzeilen; Die Elemente werden <code><Column></code> oder <code><Column_n></code> (bei aktiviertem Parameter <code>useIndexedColumns</code>) benannt. (Standard)

columnNamesAfterStart	<p>Werte aus der Zeile, die in Parameter start angegeben wurde, als Spaltenbezeichnungen interpretieren und im XML-Dokument die Elemente entsprechend benennen, falls es sich um gültige XML-Elementnamen handelt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Die übersprungenen Reihen ignorieren und den Spaltenbezeichnungen aus der ersten verarbeiteten Zeile verwenden • <i>nein</i>: Die erste verarbeitete Zeile enthält keine Spaltenbezeichnungen; Die Elemente werden <Column> oder <Column_n> (bei aktiviertem Parameter useIndexedColumns) benannt. (Standard)
excel	<p>Bestimmt, ob das CSV-Textdokument von <i>MS Excel</i> erzeugt wurde (kein Standard-CSV-Format) und dementsprechend konvertiert werden muss</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: CSV-Textdokument im <i>MS Excel CSV</i>-Format verarbeiten • <i>nein</i>: Standard-CSV-Textdokument verarbeiten (Standard)
start	<p>Erste Datenzeile, die eingelesen werden soll (die Zählung beginnt bei 0; Wenn die CSV-Datei eine Zeile mit Spaltenbezeichnungen enthält, dann wird diese ausgelassen.)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 0: Alle Datenzeilen ab der ersten einlesen (Standard)
count	<p>Anzahl der Datensätze, die eingelesen werden sollen (bei Operation Iterate pro Durchgang)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 0: Keine Beschränkung, d. h. alle Datensätze auf einmal einlesen (Standard)

commentStart	<p>Präfix zur Kennzeichnung von Kommentarzeilen; Zutreffende Zeilen werden ausgelassen.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige Zeichenkette (jedoch keine Trenn- oder Anführungszeichen)(z. B. ****) • <i>(leer)</i>: Keine Kommentarbehandlung; Zeilen werden als Datensätze interpretiert. (Standard)
useIndexedColumns	<p>Den Datenelementnamen die entsprechende Spaltennummer (beginnend bei 0) hinzufügen, wenn keine Spaltenbezeichnungen verfügbar sind</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Spaltennummer an den Elementnamen anhängen (z. B. <i><Column_0></i>) • <i>nein</i>: Alle Datenelemente erhalten den Namen <i><Column></i> (Standard)
id	<p>Eindeutige ID des <i>CSV Converter Adapter</i>-Bausteins innerhalb des Prozess-Kontextes bzw. der Prozess-Instanz (wenn Sie mehrere voneinander unabhängig iterierende <i>CSV Converter Adapter</i> in demselben Prozess-Kontext verwenden möchten)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>0</i>: Aktuelle Action-ID (Nummer des Prozess-Schritts) als ID verwenden; Beachten Sie hierbei, dass ein <i>CSV Converter Adapter</i> in einem Subprozess die gleiche Action-ID besitzen könnte. (Standard) • Beliebige ganze Zahl außer <i>0</i>, um den jeweiligen <i>CSV Converter Adapter</i>-Baustein innerhalb des Prozess-Kontextes eindeutig zu kennzeichnen; Empfehlung: einen Wert <i>>1000</i> verwenden, um nicht zufällig eine bereits bestehende Action-ID zu verwenden.
inputEncoding	<p>Definiert, welche Zeichenkodierung im Adapter-Input verwendet wird.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • UTF-8 • windows-1251 • ISO 5589-1 <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> Als Zeichenkodierung für den Adapter-Output wird immer UTF-8 verwendet.</p> </div>

Statuswerte

-1	Fehler bei der Ausführung des Adapters (Details siehe Server-Log)
0	Der Adapter gibt keine Daten (mehr) aus. Dieser Status kann bei Operation Iterate als Abbruchbedingung einer Schleife im X4-Prozess ausgewertet werden und gibt an, dass keine weiteren Datensätze verfügbar sind.
1	Der Adapter gibt ein Ergebnis aus. Dieser Status kann bei Operation Iterate als Schleifeneintrittsbedingung im X4-Prozess ausgewertet werden und gibt an, dass ein weiterer Datensatz verfügbar ist.

Input

Der Adapter erwartet eine beliebige strukturierte CSV-Textdatei (ggf. auch von *MS Excel* erzeugt) mit einem beliebigem Trennzeichen und optionalen Kommentarzeilen.

Ausgabe

Dieser Funktions-Adapter gibt ein UTF-8-kodiertes XML-Dokument mit folgender Struktur aus. Für jede verarbeitete Zeile wird ein Element erzeugt, das ggf. den Namen aus Parameter `rowName` enthält:

```
<Csv>
  <!-- Wenn Spaltenbezeichnungen übernommen werden -->
  <Elementname rowNo="Zeilennummer im Ausgangsdokument">
    <Spaltenbezeichnung>Wert</Spaltenbezeichnung>
    <!--...-->
  </Elementname>

  <!-- Wenn keine Spaltenbezeichnungen verfügbar sind -->
  <Elementname rowNo="Zeilennummer im Ausgangsdokument">
    <Column>Wert</Column>
    <!--...-->
  </Elementname>

  <!-- Wenn keine Spaltenbezeichnungen verfügbar sind
        und Parameter 'useIndexedColumns' aktiviert ist -->
  <Elementname rowNo="Zeilennummer im Ausgangsdokument">
    <Column_Spaltennummer>Wert</Column_Spaltennummer>
    <!--...-->
  </Elementname>
</Csv>
```

Beispiel

Eine CSV-Textdatei mit Kommentaren soll verarbeitet werden, die erst in der zweiten Zeile die Spaltenbeschreibungen enthält, beispielsweise:

```

**** This line is just a comment and should not be processed. ****
"METADATA","Value 1","Value 2","Value 3","Value 4","Value 5"
"CustomerID","CompanyName","ContactName","ContactTitle","Address","PostalCode"
"ALFKI","Alfreds Futterkiste","Maria Anders","Sales","Obere Str. 57","12209"
**** This line is just a comment and should not be processed. ****
"ANTON","Moreno Taquería","Antonio Moreno","Owner","Mataderos  2312","05023"

```

Hierzu wird der Adapter-Parameter `columnNamesAfterStart` aktiviert und Parameter `start` auf `1` gesetzt, weil sowohl Kommentar- als auch Spaltenbeschriftungszeile nicht mitgezählt werden. Mit dem Wert `Dataset` für Parameter `rowName` wird beispielsweise folgendes XML-Dokument erzeugt:

```

<Csv>
  <Dataset rowNo="2">
    <CustomerID>ALFKI</CustomerID>
    <CompanyName>Alfreds Futterkiste</CompanyName>
    <ContactName>Maria Anders</ContactName>
    <ContactTitle>Sales</ContactTitle>
    <Address>Obere Str. 57</Address>
    <PostalCode>12209</PostalCode>
  </Dataset>
  <Dataset rowNo="3">
    <CustomerID>ANTON</CustomerID>
    <CompanyName>Moreno Taquería</CompanyName>
    <ContactName>Antonio Moreno</ContactName>
    <ContactTitle>Owner</ContactTitle>
    <Address>Mataderos 2312</Address>
    <PostalCode>05023</PostalCode>
  </Dataset>
</Csv>

```

6.10.5.2 CSV Converter from URL



CSV Converter from URL: Liest ein CSV-Textdokument (*comma separated values*) von einer beliebigen gültigen URL ein und gibt für die angegebenen Datensätze ein XML-Dokument in UTF-8-Zeichenkodierung aus, das ggf. die Spaltenbezeichnungen als Elementnamen verwendet. Zeilen vor Beginn der CSV-Datentzeilen sowie Kommentarzeilen können ausgelassen werden. Sowohl CR (*carriage return*) als auch CRLF (*carriage return line feed*) werden als Zeilenumbrüche interpretiert. Mit Operation `Iterate` kann der Adapter das Input-Dokument schrittweise durchlaufen und für jede Iteration im technischen Prozess wird ein Ergebnis-Dokument ausgegeben.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Parse</i>: CSV-Textdokument in einem Schritt einlesen • <i>Iterate</i>: CSV-Textdokument schrittweise einlesen • <i>Clear</i>: Iterator eines <i>CSV Converter from URL Adapters</i> zurücksetzen, der dieselbe ID besitzt wie der gewünschte <i>CSV Converter from URL Adapters</i> mit der Operation <i>Iterate</i> (siehe Parameter <i>id</i>)
------------------	---

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.csv.CsvAdapter</i>: Hauptklasse (Standard)</p>
<i>delimiter</i>	<p>Das Trennzeichen zwischen den einzelnen Datenfeldern einer Zeile im CSV-Textdokument</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Ein beliebiges Zeichen (keine Escape-Sequenzen wie <i>lt</i>) • <i>;</i>: Semikolon (Standard)
<i>rowName</i>	<p>Elementname im ausgegebenen XML-Dokument, das jede Datenzeile umschließt</p> <p>Mögliche Werte: Beliebiger gültiger XML-Elementname (ohne Namespace)</p>
<i>columnNamesInFirstRow</i>	<p>Datenfelder der ersten Zeile in der CSV-Textdatei als Spaltenbezeichnungen interpretieren und Datenfeld-Elemente im XML-Dokument entsprechend benennen, falls es sich um gültige XML-Elementnamen handelt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Die erste Zeile im CSV-Textdokument enthält Spaltenbezeichnungen, die sich als gültige XML-Elementnamen (ohne Namespace-Angaben) eignen. • <i>nein</i>: CSV-Textdokument beginnt direkt mit Datenzeilen; Die Elemente werden <i><Column></i> oder <i><Column_n></i> (bei aktiviertem Parameter <i>useIndexedColumns</i>) benannt. (Standard)

<i>columnNamesAfterStart</i>	<p>Werte aus der Zeile, die in Parameter <i>start</i> angegeben wurde, als Spaltenbezeichnungen interpretieren und im XML-Dokument die Elemente entsprechend benennen, falls es sich um gültige XML-Elementnamen handelt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Die übersprungenen Reihen ignorieren und den Spaltenbezeichnungen aus der ersten verarbeiteten Zeile verwenden • <i>nein</i>: Die erste verarbeitete Zeile enthält keine Spaltenbezeichnungen; Die Elemente werden <code><Column></code> oder <code><Column_n></code> (bei aktiviertem Parameter <code>useIndexedColumns</code>) benannt. (Standard)
<i>excel</i>	<p>Bestimmt, ob das CSV-Textdokument von <i>MS Excel</i> erzeugt wurde (kein Standard-CSV-Format) und dementsprechend konvertiert werden muss</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: CSV-Textdokument im <i>MS Excel CSV</i>-Format verarbeiten • <i>nein</i>: Standard-CSV-Textdokument verarbeiten (Standard)
<i>start</i>	<p>Erste Datenzeile, die eingelesen werden soll (die Zählung beginnt bei 0; Wenn die CSV-Datei eine Zeile mit Spaltenbezeichnungen enthält, dann wird diese ausgelassen.)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 0: Alle Datenzeilen ab der ersten einlesen (Standard)
<i>count</i>	<p>Anzahl der Datensätze, die eingelesen werden sollen (bei Operation <i>Iterate</i> pro Durchgang)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 0: Keine Beschränkung, d. h. alle Datensätze auf einmal einlesen (Standard)

<i>commentStart</i>	<p>Präfix zur Kennzeichnung von Kommentarzeilen; Zutreffende Zeilen werden ausgelassen.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige Zeichenkette (jedoch keine Trenn- oder Anführungszeichen)(z. B. ****) • (<i>leer</i>): Keine Kommentarbehandlung; Zeilen werden als Datensätze interpretiert. (Standard)
<i>useIndexedColumns</i>	<p>Den Datenelementnamen die entsprechende Spaltennummer (beginnend bei 0) hinzufügen, wenn keine Spaltenbezeichnungen verfügbar sind</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Spaltennummer an den Elementnamen anhängen (z. B. <i><Column_0></i>) • <i>nein</i>: Alle Datenelemente erhalten den Namen <i><Column></i> (Standard)
<i>id</i>	<p>Eindeutige ID des <i>CSV Converter from URL Adapter</i>-Bausteins innerhalb des Prozess-Kontextes bzw. der Prozess-Instanz (wenn Sie mehrere voneinander unabhängig iterierende <i>CSV Converter from URL Adapter</i> in demselben Prozess-Kontext verwenden möchten)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>0</i>: Aktuelle Action-ID (Nummer des Prozess-Schritts) als ID verwenden; Beachten Sie hierbei, dass ein <i>CSV Converter from URL Adapter</i> in einem Subprozess die gleiche Action-ID besitzen könnte. (Standard) • Beliebige ganze Zahl außer <i>0</i>, um den jeweiligen <i>CSV Converter from URL Adapter</i>-Baustein innerhalb des Prozess-Kontextes eindeutig zu kennzeichnen; Empfehlung: einen Wert <i>> 1000</i> verwenden, um nicht zufällig eine bereits bestehende Action-ID zu "erwischen"
<i>url</i>	<p>URL zum einzulesenden CSV-Textdokument</p> <p>Mögliche Werte:</p> <p>Beliebige gültige URL (z. B. <i>xstore://Project/Folder/File.csv</i>)</p>

<i>encoding</i>	Zeichenkodierung des CSV-Textdokuments Mögliche Werte: <ul style="list-style-type: none"> • Beliebige gültige Zeichenkodierung (z. B. <i>UTF-8</i>) • (<i>leer</i>): Standard-Zeichenkodierung der Laufzeitumgebung verwenden
-----------------	---

Statuswerte

-1	Fehler bei der Ausführung des Adapters (Details siehe Server-Log)
0	Der Adapter gibt keine Daten (mehr) aus. Dieser Status kann bei Operation Iterate als Abbruchbedingung einer Schleife im technischen Prozess ausgewertet werden und gibt an, dass keine weiteren Datensätze verfügbar sind.
1	Der Adapter gibt ein Ergebnis aus. Dieser Status kann bei Operation Iterate als Schleifeneintrittsbedingung im technischen ausgewertet werden und gibt an, dass ein weiterer Datensatz verfügbar ist.

Input

Der Adapter erwartet eine beliebige strukturierte CSV-Textdatei (ggf. auch von *MS Excel* erzeugt) mit einem beliebigem Trennzeichen und optionalen Kommentarzeilen.

Ausgabe

Dieser Funktions-Adapter gibt ein UTF-8-kodiertes XML-Dokument mit folgender Struktur aus. Für jede verarbeitete Zeile wird ein Element erzeugt, das ggf. den Namen aus Parameter *rowName* enthält:

```
<Csv>
  <!-- Wenn Spaltenbezeichnungen übernommen werden -->
  <Elementname rowNo="Zeilennummer im Ausgangsdokument">
    <Spaltenbezeichnung>Wert</Spaltenbezeichnung>
    <!--...-->
  </Elementname>

  <!-- Wenn keine Spaltenbezeichnungen verfügbar sind -->
  <Elementname rowNo="Zeilennummer im Ausgangsdokument">
    <Column>Wert</Column>
    <!--...-->
  </Elementname>

  <!-- Wenn keine Spaltenbezeichnungen verfügbar sind
        und Parameter 'useIndexedColumns' aktiviert ist -->
  <Elementname rowNo="Zeilennummer im Ausgangsdokument">
    <Column_Spaltennummer>Wert</Column_Spaltennummer>
    <!--...-->
  </Elementname>
</Csv>
```

Beispiel

Eine CSV-Textdatei mit Kommentaren soll verarbeitet werden, die erst in der zweiten Zeile die Spaltenbeschreibungen enthält, beispielsweise:

```
**** This line is just a comment and should not be processed. ****
"METADATA","Value 1","Value 2","Value 3","Value 4","Value 5"
"CustomerID","CompanyName","ContactName","ContactTitle","Address","PostalCode"
"ALFKI","Alfreds Futterkiste","Maria Anders","Sales","Obere Str. 57","12209"
**** This line is just a comment and should not be processed. ****
"ANTON","Moreno Taquería","Antonio Moreno","Owner","Mataderos  2312","05023"
```

Hierzu wird der Adapter-Parameter `columnNamesAfterStart` aktiviert und Parameter `start` auf `1` gesetzt, weil sowohl Kommentar- als auch Spaltenbeschriftungszeile nicht mitgezählt werden. Mit dem Wert `Dataset` für Parameter `rowName` wird beispielsweise folgendes XML-Dokument erzeugt:

```
<Csv>
  <Dataset rowNo="2">
    <CustomerID>ALFKI</CustomerID>
    <CompanyName>Alfreds Futterkiste</CompanyName>
    <ContactName>Maria Anders</ContactName>
    <ContactTitle>Sales</ContactTitle>
    <Address>Obere Str. 57</Address>
    <PostalCode>12209</PostalCode>
  </Dataset>
  <Dataset rowNo="3">
    <CustomerID>ANTON</CustomerID>
    <CompanyName>Moreno Taquería</CompanyName>
    <ContactName>Antonio Moreno</ContactName>
    <ContactTitle>Owner</ContactTitle>
    <Address>Mataderos 2312</Address>
    <PostalCode>05023</PostalCode>
  </Dataset>
</Csv>
```

6.10.5.3 XML to CSV Converter



XML to CSV Converter: Konvertiert eine XML-Datei in eine CSV-Datei.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> <i>Convert</i>: Konvertiert die XML-Datei in eine CSV-Datei.
------------------	--

Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.xml2csv.XML2CSVAdapter</code>: Hauptklasse (Standard)</p>
headerInOutput	<p><i>Erforderlich.</i> Legt fest, ob es in der zu erzeugenden CSV-Datei eine Kopfzeile gibt.</p> <p>Mögliche Werte: <code>true</code> (Standard) / <code>false</code></p>
delimiter	<p><i>Erforderlich.</i> Legt das Trennzeichen der zu erzeugenden CSV-Datei fest.</p> <p>Mögliche Werte: Beliebiges Trennzeichen, z. B. <code>;</code> , <code>,</code> (Standard) oder <code>\t</code> (Tab als Trennzeichen)</p>
lineEnd	<p><i>Erforderlich.</i> Legt das Zeichen für das Zeilenende in der zu erzeugenden CSV-Datei fest.</p> <p>Mögliche Werte: Zeichen für das Zeilenende, z. B. <code>\n</code> (Standard Unix), <code>\r\n</code> (Standard MS Windows)</p>
quotechar	<p>Legt das Anführungszeichen für die Elemente innerhalb der zu erzeugenden CSV-Datei fest.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebiges Anführungszeichen, z. B. <code>"</code> • Standard: Nicht gesetzt
escapechar	<p>Legt das Zeichen fest, mit dem bestimmte Zeichen maskiert werden, die ansonsten als Steuerzeichen erkannt würden.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • beliebiges Zeichen, z. B. <code>"</code> • Standard: Nicht gesetzt
encoding	<p>Legt die Zeichenkodierung für die Konvertierung fest.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Angabe einer Zeichenkodierung, z. B. <code>UTF-8</code> • Standard: <code>UTF-8</code>

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt.
-1	Die Operation konnte nicht ausgeführt werden.

Input

Der Adapter erwartet die folgende Input-Struktur:

Input-Struktur

```
<?xml version="1.0" encoding="UTF-8"?>
<Csv>
  <Header>
    <Column name="columnName1" />
    <Column name="columnName2" />
    <Column ...
  </Header>
  <Records>
    <Record>
      <Column name="columnName1">value1</Column>
      <Column name="columnName2">value2</Column>
      <Column ...
    </Record>
    <Record>
      <Column name="columnName1">value1</Column>
      <Column name="columnName2">value2</Column>
      <Column ...
    </Record>
    <Record ...
  </Records>
</Csv>
```

- ❗ Wenn der Parameter `headerInOutput` auf `true` gesetzt wurde, dann wird in der Ausgabe-CSV-Datei eine Kopfzeile erzeugt. Die Spaltennamen werden über das Attribut `name` innerhalb des `Header/Column`-Elements festgelegt. Es werden so viele Spalten erzeugt, wie es `name`-Attribute gibt.
- Wenn der Parameter `headerInOutput` auf `false` gesetzt wurde, dann wird in der Ausgabe-CSV-Datei keine Kopfzeile erzeugt. Es werden so viele Spalten erzeugt, wie maximal als `Column`-Elemente innerhalb eines `Record`-Elements vorkommen.
- Jeder Eintrag muss in einem `Record`-Element stehen. Wenn eine Kopfzeile definiert ist, dann werden die Werte der Spalte mit dem entsprechenden `name` zugeordnet.

Ausgabe

Der Adapter gibt ein CSV-Dokument in der Struktur aus, die über das Eingabedokument definiert wurde.

Beispiel

Beispiel Input XML to CSV Converter

```
<?xml version="1.0" encoding="UTF-8"?>
<Csv>
  <Header>
    <Column name="Monday" />
    <Column name="Tuesday" />
    <Column name="Wednesday" />
  </Header>
  <Records>
    <Record>
      <Column name="Monday">First day</Column>
      <Column name="Tuesday">Second day</Column>
    </Record>
    <Record>
      <Column name="Monday">First day</Column>
      <Column name="Tuesday">Second day</Column>
      <Column name="Wednesday">Third day</Column>
    </Record>
  </Records>
</Csv>
```

Die Adapter-Parameter wurden wie folgt gesetzt:

- headerInOutput: *true*
- delimiter: ,
- encoding: *UTF-8*

Obiger Input führt zu folgendem Output:

Beispiel Output XML to CSV Converter

```
Monday,Tuesday,Wednesday
First day,Second day,
First day,Second day,Third day
```

6.10.6 Encoding Converter



Encoding Converter: Konvertiert die Zeichenkodierung eines XML-Dokuments. Dabei werden sowohl die Deklaration der Zeichenkodierung (XML-Prolog) als auch die eigentliche Zeichenkodierung des Inhaltes geändert.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Execute</i> : Zeichen der Input-XML-Daten in die in Parameter TargetEncoding angegebene Zeichenkodierung konvertieren
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.encodingconversion.DocumentEncodingAdapter</i> : Hauptklasse (Standard)
<i>TargetEncoding</i>	Neue Zeichenkodierung des ausgegebenen XML-Dokuments Mögliche Werte: Beliebige von der Java-Laufzeitumgebung unterstützte Zeichenkodierung (z.B. <i>ISO-8859-1</i>)

6.10.7 Hex Converter



Hex Converter: Konvertiert Binärdaten hexadezimal und dekodiert hexadezimal kodierte Daten.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>Encode</i>: Daten hexadezimal kodieren • <i>Decode</i>: Hexadezimale Daten dekodieren
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.HexAdapter</i> : Hauptklasse (Standard)
----------------	---

6.10.8 Image Processing

6.10.8.1 Image to Image Converter



Image to Image Converter: Konvertiert und skaliert eine Bilddatei in ein anderes Bildformat. Folgende Formate können gelesen und erzeugt werden: BMP, GIF, JPEG, JPEG 2000, PNG, RAW, TIFF und WBMP. Die Bilddateien werden in einem adapterspezifischen Input-XML-Dokument angegeben. Der Adapter bietet die Möglichkeit der Angabe von Formatierungsoptionen bei der Umwandlung.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>Transformation</i> : Bildformat konvertieren
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.image2image.ImageToImageAdapter</i> : Hauptklasse (Standard)
<i>width</i>	Breite des ausgegebenen Bildes in Pixel (wird ignoriert, wenn Parameter <i>maxSize</i> gesetzt ist) Mögliche Werte: Beliebige ganze Zahl
<i>height</i>	Höhe des ausgegebenen Bildes in Pixel (wird ignoriert, wenn Parameter <i>maxSize</i> gesetzt ist) Mögliche Werte: Beliebige ganze Zahl
<i>maxSize</i>	Wenn das Bild im ursprünglichen Seitenverhältnis skaliert werden soll: Maximale Breite oder Höhe des ausgegebenen Bildes in Pixel (Parameter <i>width</i> und <i>height</i> werden ignoriert) Mögliche Werte: <ul style="list-style-type: none"> • Beliebige ganze Zahl • 0: Ursprüngliches Seitenverhältnis <i>nicht</i> beibehalten und Parameter <i>width</i> und <i>height</i> zur Skalierung verwenden (Standard)
<i>scaleAlgorithm</i>	Verwendeter Skalierungs-Algorithmus Mögliche Werte: <ul style="list-style-type: none"> • <i>SCALE_DEFAULT</i>: Standard-Algorithmus verwenden • <i>SCALE_FAST</i>: Schnellen Algorithmus verwenden • <i>SCALE_SMOOTH</i>: Kantenglättungs-Algorithmus verwenden • <i>SCALE_REPLICATE</i>: Pixelwiederholungs-Algorithmus verwenden • <i>SCALE_AREA-AVERAGING</i>: Flächen-Durchschnittsberechnungs-Algorithmus verwenden

<i>outputFormat</i>	Ausgegebenes Bildformat Mögliche Werte: <ul style="list-style-type: none"> • <i>BMP</i>: Windows Bitmap erzeugen • <i>GIF</i>: GIF-komprimiertes Bild erzeugen • <i>JPEG</i>: JPEG-komprimiertes Bild erzeugen • <i>JPEG 2000</i>: Bild im <i>JPEG 2000</i>-Format erzeugen • <i>PNG</i>: PNG-Bild erzeugen • <i>RAW</i>: Rohdaten-Bild erzeugen • <i>TIFF</i>: Bild im TIF-Format erzeugen • <i>WBMP</i>: Bild im <i>Wireless Bitmap</i>-Format erzeugen
---------------------	---

6.10.8.2 Multipage TIFF Creator



Multipage TIFF Creator: Erstellt ein mehrseitiges TIFF-Dokument (*Multipage Tagged Image File Format*) aus Bilddateien vom Typ BMP, GIF, FlashPix, JPEG, PNG, PNM, TIFF und WBPM. Die Bilddateien werden in einem adapterspezifischen Input-XML-Dokument angegeben. Der Adapter gibt eine mehrseitige TIFF-Grafik als Binärdatenstrom mit einer maximalen Größe von 4 GB aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>CreateMultiTiff</i> : Mehrseitiges TIFF-Dokument erstellen
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.tiff.Image2Tiff</i> : Hauptklasse (Standard)
----------------	--

<i>compression</i>	Kompressionstyp Mögliche Werte: <ul style="list-style-type: none"> • <i>Deflate</i>: Per Deflate verlustfrei komprimieren • <i>JPEG</i>: JPEG-Komprimierung (verlustbehaftet) verwenden • <i>None</i>: Nicht komprimieren (Standard) • <i>Packbits</i>: Per PackBits-Lauflängenkodierung verlustfrei komprimieren
<i>deflateLvl</i>	Deflate-Level (wenn Kompressionstyp <i>Deflate</i> gewählt wurde und die <i>Deflate</i> -Komprimierung auf das gesamte TIFF-Dokument angewendet werden soll) Mögliche Werte: <ul style="list-style-type: none"> • Beliebige ganze Zahl von 1–9 (komprimiert bei 1 am schnellsten, bei 9 am ausführlichsten) • 0: Keine <i>Deflate</i>-Komprimierung auf das gesamte TIFF anwenden (Standard)
<i>resX</i>	Horizontaler Bildschirm-/Druck-Auflösungswert Mögliche Werte: <ul style="list-style-type: none"> • Beliebige ganze Zahl (z. B. 72) • 0: Keinen Auflösungswert setzen
<i>resY</i>	Vertikaler Bildschirm-/Druck-Auflösungswert Mögliche Werte: <ul style="list-style-type: none"> • Beliebige ganze Zahl (z. B. 72) • 0: Keinen Auflösungswert setzen
<i>resUnit</i>	Einheit für die Bildschirm-/Druck-Auflösung Mögliche Werte: <ul style="list-style-type: none"> • <i>No absolute unit</i>: Keine absolute Einheit setzen • <i>Inch</i>: Zoll als Einheit verwenden (DPI bzw. PPI) • <i>Centimeter</i>: Zentimeter als Einheit verwenden

Input

Dieser Adapter erwartet eine adapterspezifische Input-XML-Struktur, in der sämtliche zu verarbeitenden Bilddateien über ein gültiges URL-Protokoll referenziert sind. Über ein optionales

Attribut `thumbnail_scale` können Sie angeben, ob eine skalierte Version des Bilds zusätzlich als Seite in die TIFF-Datei eingefügt werden soll. Der Attributwert gibt dabei den Skalierungsfaktor an.

```
<imagechooser>
  <image url="URL zur Bilddatei" thumbnail_scale="Skalierungsfaktor" />
</imagechooser>
```

Beispiel

Die folgende Input-XML-Struktur referenziert zwei Grafiken, eines per `xstore://` aus dem X4 Repository, das zweite über eine absolute Pfadangabe mit `file://`. Der Funktions-Adapter erzeugt eine dreiseitige TIFF-Datei, die das erste Bild unverändert enthält, auf der zweiten Seite die um 50% verkleinerte Version des ersten Bildes sowie auf der dritten Seite das zweite Bild enthält.

```
<imagechooser>
  <image url="xstore://Folder/Image.tif" thumbnail_scale="0.5" />
  <image url="file:///C:/Folder/Picture.jpg" />
</imagechooser>
```

6.10.8.3 TIFF Properties Reader



TIFF Properties Reader: Ermittelt die Eigenschaften einer TIFF-Grafik. Dieser Adapter erwartet eine TIFF-Grafik als Input und gibt sämtliche ermittelten TIFF-Eigenschaften (Abmessungen, Zahl der Seiten, Bittiefe, Orientierung etc.) als XML-Dokument aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>ReadFromImage</i> : Eigenschaften aus der TIFF-Grafik auslesen
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.tiff.TIFFproperties</i> : Hauptklasse (Standard)
----------------	---

6.10.9 Java Properties File Reader



Java Properties File Reader: Liest eine Java-Properties-Textdatei und speichert alle oder ausgewählte Properties in einer Hash-Tabelle, die während der Laufzeit des Servers für alle technischen Prozesse global verfügbar ist.

Jeder Property ist dabei ein eindeutiger Schlüssel zugeordnet. Der Adapter kann zudem die globale Hash-Tabelle lesen und deren Inhalt komplett oder nach einem RegEx-Filter in einem XML-Dokument ausgeben.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>ReadFromFile</i>: Properties-Textdatei einlesen und Properties komplett oder mit einem RegEx-Filter in einer globaler Hash-Tabelle speichern • <i>ReadFromMap</i>: Globale Hash-Tabelle lesen und deren Inhalt in einem XML-Dokument ausgeben
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.properties.PropertiesReader</i> : Hauptklasse (Standard)
<i>file</i>	Pfad zur Properties-Textdatei im Repository oder absoluter Pfad im Dateisystem des <i>X4 Servers</i> Mögliche Werte: <ul style="list-style-type: none"> • Beliebiger Repository-Pfad (z.B. <i>%USER_COLLECTION%/Project/Folder/sample.properties</i>) • Beliebiger absoluter Pfad (z.B. <i>C:\Data\sample.properties</i>)
<i>fileKey</i>	Schlüssel, der den eingelesenen Properties vorangestellt wird, um diese eindeutig identifizierbar zu machen. So wird beispielsweise eine Property <i>sample.property</i> mit dem Schlüssel <i>x4propkey</i> ergänzt und als <i>x4propkey.sample.property</i> in der Hash-Tabelle gespeichert. Mögliche Werte: Beliebige Zeichenkette (z.B. <i>x4propkey</i>)
<i>filterRegex</i>	Regulärer Ausdruck, nach dem die Properties gefiltert werden sollen (nur bei Operation <i>ReadFromFile</i>); mehrere Filterausdrücke können durch Semikolon (;) getrennt angegeben werden Mögliche Werte: Beliebiger regulärer Ausdruck (z.B. <i>.*sample.*</i>)

6.10.10 QR Code Generator

QR Code Generator: Erzeugt QR-Codes und liest diese aus.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Consume: Liest einen QR-Code aus. • Generate: Erzeugt einen QR-Code.
-----------	--

Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.qrcode.QRCodeGenerator</code>: Hauptklasse (Standard)</p>
imageWidth	<p><i>Erforderlich.</i> Breite des zu erzeugenden QR-Codes in Pixel.</p> <p>Mögliche Werte: Beliebige Ganzzahl (Standard: 350)</p> <div> <p> Der erzeugte QR-Code ist immer quadratisch. Wenn Höhe und Breite nicht identisch sind, wird mit leeren Pixeln aufgefüllt.</p> </div>
imageHeight	<p><i>Erforderlich.</i> Höhe des zu erzeugenden QR-Codes in Pixel.</p> <p>Mögliche Werte: Beliebige Ganzzahl (Standard: 350)</p> <div> <p> Der erzeugte QR-Code ist immer quadratisch. Wenn Höhe und Breite nicht identisch sind, wird mit leeren Pixeln aufgefüllt.</p> </div>
imageFormat	<p><i>Erforderlich.</i> Dateiformat der Grafikdatei für den zu erzeugenden QR-Code.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • JPEG 2000 • PNG • RAW • TIFF • WBMP

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt.
-1	Die Operation konnte nicht ausgeführt werden.

Input

Der Adapter erwartet abhängig von der auszuführenden Operation unterschiedliche Daten als Input.

- Consume-Operation: Grafik mit QR-Code
- Generate-Operation: Reiner Text

Output:

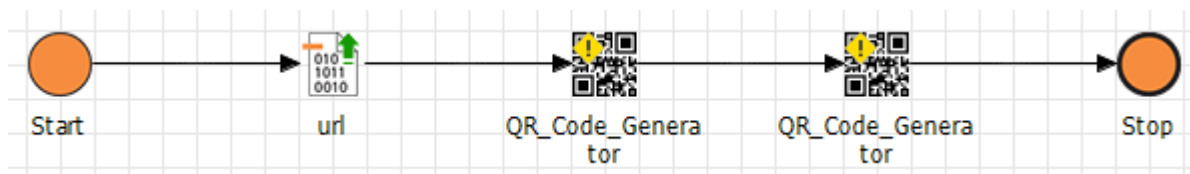
Der Adapter liefert abhängig von der ausgeführten Operation unterschiedliche Daten als Output.

- Consume-Operation: Inhalt, der aus dem QR-Code ausgelesen wurde.
- Generate-Operation: Grafik mit QR-Code

Beispiel

Mit dem QR Code Generator können beispielsweise QR-Codes erzeugt und ausgelesen werden, die eine URL enthalten.

Der Prozess dazu sieht folgendermaßen aus:



Label	Datei	Beschreibung
url	url.txt	Textdatei, die die URL <code>https://softproject.de/de/</code> enthält.
QR_Code_Generator	Adapter QR Code Generator	Operation Generate
QR_Code_Generator	Adapter QR Code Generator	Operation Consume

Parameter des Adapters:

- `imageWidth`: 350
- `imageHeight`: 350
- `imageFormat`: PNG

6.10.11 Regular Expression



Regular Expression: Durchsucht ein Input-Textdokument nach einem Muster, das in einem Regulären Ausdruck (*Regex*) angegeben ist, und gibt ein XML-Dokument aus, das für jede übereinstimmende Zeile ein XML-Element enthält.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Grep</i> : Textdokument nach dem in Parameter <i>pattern</i> angegebenen Muster durchsuchen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.grep.Grep:</i> Hauptklasse (Standard)
<i>pattern</i>	Regex-Muster, nach dem gesucht wird (Regulärer Ausdruck) Mögliche Werte: Beliebiger Regulärer Ausdruck (z. B. <i>.*Test.*</i>)

Ausgabe

Dieser Adapter gibt ein UTF-8-kodiertes XML-Dokument mit folgender Struktur aus. Für jede übereinstimmende Zeile wird ein XML-Element `<Line/>` erzeugt, das die Zeilennummer und den Inhalt der jeweiligen Zeile enthält:

```
<Text>
  <Line lineNo="Zeilennummer">Inhalt der Zeile</Line>
</Text>
```

6.10.12 Stream Splitter

Stream Splitter: Zerlegt einen Datenstrom in Teile, deren Größe in Byte festgelegt werden kann. Der Adapter gibt Datenstrom-Teile mit der gewünschten Länge nacheinander aus oder gibt die Teile in aufsteigender oder absteigender Reihenfolge aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> <i>Cut</i>: Datenstrom aufteilen und jeweils einen Teil als Datenstrom ausgeben <i>Clear</i>: Zähler auf Anfangswert zurücksetzen und keinen Teil ausgeben
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.streamsplitter.StreamSplitter:</i> Hauptklasse (Standard)
<i>BlockSize</i>	Größe der Blöcke in Bytes (der letzte Teil des Binärdatenstroms kann kleiner sein) Mögliche Werte: Beliebige ganze Zahl

<i>SegmentCount</i>	<p>Adapter-Status beeinflussen (nützlich in Kombination mit einem <i>Condition</i>-Baustein, wenn nur ein bestimmter Teil des Binärdatenstroms, beispielsweise der letzte Teil, verarbeitet werden soll)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>none</i>: Status nicht beeinflussen (Adapter erhält den Status <i>1</i>, wenn ein Teil ausgegeben wird)(Standard) • <i>increment</i>: Als Status den aktuellen Zählerwert ausgeben, beginnend bei <i>1</i> • <i>decrement</i>: Als Status den aktuellen Zählerwert ausgeben, beginnend bei der höchsten Zählernummer, und den Zählerwert um 1 dekrementieren
---------------------	--

Statuswerte

<i>-1(error)</i>	Während der Ausführung des Funktions-Adapters trat ein Fehler auf. Nähere Details finden Sie im Server-Log.
<i>0(successful)</i>	Adapter wurde erfolgreich ausgeführt, gab jedoch keine Daten aus, weil keine weiteren Teile verfügbar sind (Operation Cut) oder weil der Adapter keine Daten verarbeitete, sondern lediglich den Iterator zurücksetzte (Operation Clear).
<i>1(successful)</i>	Adapter wurde erfolgreich ausgeführt und gab den aktuellen Teil des Binärdatenstroms aus. Wenn der Parameter SegmentCount auf <i>increment</i> gesetzt wurde: Der Adapter gab den ersten Teil des Binärdatenstroms aus.
<i>2 ... n</i>	Nur wenn der Parameter SegmentCount auf <i>increment</i> oder <i>decrement</i> gesetzt wurde: Der Adapter gibt den zweiten oder <i>n</i> -ten Teil des Binärdatenstroms aus.

6.10.13 Structured Text

6.10.13.1 Structured Text Converter



Structured Text Converter: Wandelt ein XML- in ein Text-Dokument oder ein Text- in ein XML-Dokument gemäß den Regeln einer ASCII-Deskriptordatei um.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>TextToXML</i>: Textdokument einlesen und in ein XML-Dokument schreiben • <i>XmlToText</i>: XML-Daten in ein Textdokument schreiben
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.text.StructuredTextConverter</i> : Hauptklasse (Standard)
<i>DescFile</i>	Name der ASCII-Deskriptordatei, die das Textdokument in seiner Datenstruktur beschreibt. Die ASCII-Deskriptordatei sollte im Benutzerordner des X4 Repositorys abgelegt sein (z. B. <i>X4DB/1/Project/Descr.ad</i>) Mögliche Werte: Pfad und Dateiname (z. B. <i>/Project/Descr.ad</i>)
<i>encoding</i>	Zeichensatz, in dem das Textdokument kodiert sein soll Mögliche Werte: <ul style="list-style-type: none"> • Beliebige Zeichenkette für einen Zeichensatz (z. B. <i>UTF-8</i>) • <i>(leer)</i>: Zeichensatz des verarbeiteten XML-Dokuments verwenden (Standard)
<i>Strict</i>	Bestimmt, ob der Datenstrom aus dem vorangehenden Prozess-Schritt entsprechend lang sein muss, um die vollständige Struktur zu lesen Mögliche Werte: <ul style="list-style-type: none"> • <i>true</i>: Vollständige Struktur muss gelesen werden • <i>false</i>: Datenstrom kann beliebig lang sein (Standard)

6.10.13.2 COBOL Copybook Reader



COBOL Copybook Reader: Liest ein Textdokument im *COBOL Copybook*-Format, parst es gemäß den Regeln der ASCII-Deskriptordatei und gibt ein XML-Dokument aus.

① Zeichen, die keine gültigen XML-Zeichen sind, werden durch den *Unicode Replacement Character* #xFFFD ersetzt. Gültige XML-Zeichen sind folgende Unicode-Zeichen:

- #x9
- #xA
- #xD
- #x20 – #xD7FF
- #xE000 – #xFFFD

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>Execute</i> : Textdokument im <i>COBOL Copybook</i> -Format einlesen
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.CopyBookReader</i> : Hauptklasse (Standard)
<i>DescFile</i>	Name der ASCII-Deskriptordatei, die das Textdokument im <i>COBOL Copybook</i> -Format in seiner Datenstruktur beschreibt. Die ASCII-Deskriptordatei sollte im Benutzerordner des X4 Repositorys abgelegt sein (z.B. X4DB/1/Project/Descr.ad) Mögliche Werte: Pfad und Dateiname (z.B. <i>/Project/Descr.ad</i>)
<i>Strict</i>	Bestimmt, ob der Datenstrom aus dem vorangehenden Prozess-Schritt entsprechend lang sein muss, um die vollständige Struktur zu lesen Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: Vollständige Struktur muss gelesen werden • <i>nein</i>: Datenstrom kann beliebig lang sein (Standard)

6.10.13.3 COBOL Copybook Writer



COBOL Copybook Writer: Konvertiert ein XML-Dokument gemäß den Regeln einer ASCII-Deskriptordatei in ein Textdokument im *COBOL Copybook*-Format.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>Execute</i> : XML-Daten in ein Textdokument im <i>COBOL Copybook</i> -Format schreiben
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.CopyBookWriter</i> : Hauptklasse (Standard)
<i>DescFile</i>	Name der ASCII-Deskriptordatei, die das Textdokument im <i>COBOL Copybook</i> -Format in seiner Datenstruktur beschreibt. Die ASCII-Deskriptordatei sollte im Benutzerordner des X4 Repositorys abgelegt sein (z.B. X4DB/1/Project/Descr.ad) Mögliche Werte: Pfad und Dateiname (z.B. <i>/Project/Descr.ad</i>)
<i>encoding</i>	Zeichensatz, in dem das Textdokument im <i>COBOL Copybook</i> -Format kodiert sein soll Mögliche Werte: <ul style="list-style-type: none"> • Beliebige Zeichenkette für einen Zeichensatz (z.B. <i>UTF-8</i>) • <i>(leer)</i>: Zeichensatz des verarbeiteten XML-Dokuments verwenden (Standard)
<i>DebugMode</i>	Debug-Modus

6.10.14 Text Replacement



Text Replacement: Durchsucht ein Textdokument nach einer Zeichenkette (Groß-/Kleinschreibung wird beachtet) und gibt ein Textdokument mit derselben Zeichenkodierung aus, in dem jede gesuchte Zeichenkette durch eine neue Zeichenkette ersetzt wurde.

- i** Falls Sie eine Zeichenkette in einem XML-Dokument ersetzen möchten, beachten Sie, dass der Adapter immer ein reines Textdokument ausgibt und es sich bei diesem somit nicht mehr zwingend um wohlgeformtes XML handelt!
Die Zeichenkodierung des Dokuments muss explizit gesetzt sein, z. B. über den *Document Properties Changer*.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Replace</i> : Zeichenkette ersetzen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.replace.Replace</i> : Hauptklasse (Standard)
<i>searchString</i>	Zeichenkette, nach der das Input-Textdokument durchsucht werden soll Mögliche Werte: Beliebige Zeichenkette
<i>replaceString</i>	Zeichenkette, durch die <i>searchString</i> im Ergebnis-Textdokument ersetzt werden soll Mögliche Werte: Beliebige Zeichenkette

Input

Dieser Adapter erwartet ein beliebiges Textdokument.

Ausgabe

Der Adapter gibt ein Textdokument aus, in dem der gefundene Text ersetzt wurde.

6.10.15 Transliteration Adapter

Transliteration Adapter: Transliteriert Text aus einem Format in ein anderes, z. B. von UTF-8 nach LATIN1 (ä → ae etc.). Der Adapter führt weder Übersetzungen durch, noch ändert er den Zeichensatz.

Eigenschaften (Properties)

Operation	Vom Adapter ausgeführte Operation Mögliche Werte: <ul style="list-style-type: none"> • <i>GetAvailableIDs</i>: Führt alle verfügbaren Transliteratoren auf • <i>Transliterate</i>: Transliteriert den Text gemäß eines spezifizierten Transliterators (id) • <i>TransliterateFromRule</i>: Transliteriert den Text gemäß einer spezifizierten Regel (siehe http://userguide.icu-project.org/transforms/general/rules)
------------------	--

Parameter

Adapter	Adapter-Hauptklasse (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.transliteration.TransliterationAdapter</i> : Hauptklasse (Standard)
id	ID des verwendeten Transliterators Mögliche Werte: Wert, der von der Operation <i>GetAvailableIDs</i> ausgegeben wird
rule	Verwendete Regel für die Transliteration Mögliche Werte: Beliebige Regel spezifiziert unter http://userguide.icu-project.org/transforms/general/rules

Statuswerte

1	Der Adapter wurde erfolgreich ausgeführt
-1	Es kam zu einem Fehler während der Adapter-Ausführung

Input

Der Adapter erwartet ein Text- oder XML-Dokument als Input.

Ausgabe

Der Adapter gibt als Ergebnis das transliterierte Text- oder XML-Dokument aus.

6.11 Geo Location

6.11.1 Bing Maps



Bing Maps: Greift auf *Bing Maps* über dessen Webservice-Schnittstelle in Version 1 zu und ermittelt die geografischen Informationen zu einer Adresse. Zudem kann eine Route inklusive Wegpunkten berechnet und eine URL zu einem *Bing Maps* Kartenausschnitt erzeugt werden. Der Adapter gibt seine Ergebnisse wahlweise als XML- oder JSON-Dokument aus.



Um die *Bing Maps API* nutzen zu können, wird ein *Bing Maps API Key* benötigt.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Geocode</i>: Zu einer Adresse sämtliche Geoinformationen (Anschrift, Bezirk/Land, Nation, Koordinaten etc.) ermitteln • <i>ReverseGeocode</i>: Zu einer Dezimalkoordinate sämtliche Geoinformationen (Anschrift, Bezirk/Land, Nation, Koordinaten etc.) ermitteln • <i>GetDirections</i>: Route zwischen einem Start- und Zielpunkt mit beliebig vielen Wegpunkten ermitteln • <i>CreateBingURL</i>: URL zu einem <i>Bing Maps</i> Webservice
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.geo.microsoft.BingMapsAdapter</i> : Hauptklasse (Standard)</p>
<i>bingMapsKey</i>	Schlüssel zur Nutzung der <i>Bing Maps API</i>
<i>outputFormat</i>	<p>Ausgabeformat für Anfragen definieren</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>xml</i>: XML-Dokument ausgeben • <i>json</i>: JSON-Dokument ausgeben
<i>timeout</i>	<p>Zeitbeschränkung für den Verbindungsaufbau in Millisekunden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl • <i>0</i>: Keine Zeitbeschränkung • <i>20000</i>: Nach 20 Sekunden den Verbindungsaufbau abbrechen
<i>pxHost</i>	<p>IP-Adresse oder Host-Name eines Proxy-Servers</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette (z. B. 192.168.1.1)</p>
<i>pxPort</i>	<p>Portnummer des Proxy-Servers</p> <p>Mögliche Werte: Beliebige ganze Zahl</p>

<i>pxLogin</i>	Benutzername für den Proxy-Server Mögliche Werte: Beliebige Zeichenkette
<i>pxPwd</i>	Passwort für den Proxy-Server Mögliche Werte: Beliebige Zeichenkette

Statuswerte

<i>1(successful)</i>	Die Anfrage war erfolgreich und ein Ergebnis wurde zurückgegeben (Entspricht dem HTTP-Status-Code 200)
<i>0(empty)</i>	Die Anfrage war zwar erfolgreich, jedoch wurde kein Ergebnis zurückgegeben
<i>-1(failed)</i>	Die Anfrage schlug aufgrund eines technischen Fehlers fehl
<i>-HTTP-Status-Code</i>	Die Anfrage wurde ausgeführt, jedoch mit einem Status-Code ungleich 200 beendet. Fehlerinterpretation entsprechend dem HTTP-Status-Code.

6.11.1.1 Operation: Geocode

Mit der Operation Geocode lassen sich sämtliche Geoinformationen zu einer Adresse ermitteln.

Input

Für Operation Geocode erwartet der Adapter eine XML-Input-Struktur nach folgendem Muster, die eine vollständige Anschrift enthält (bei unvollständigen Anschriften ist die Anfrage ggf. nicht eindeutig):

```
<Geocode>
  <Address>
    <Location><!--Location, e.g. Disneyland--></Location>
    <Street><!--Street and number--></Street> -->
    <ZIP><!--ZIP code--></ZIP>
    <City><!--City--></City>
    <Country><!--Country--></Country>
  </Address>
</Geocode>
```

Ausgabe

Für Operation Geocode wird ein XML- oder JSON-Dokument ausgegeben, das die Geoinformationen zur angegebenen Adresse beschreibt.

6.11.1.2 Operation: ReverseGeocode

Mit Operation ReverseGeocode lassen sich sämtliche Geoinformationen zu einer Dezimalkoordinate ermitteln.

Input

Für Operation `ReverseGeocode` erwartet der Adapter eine XML-Input-Struktur, die eine Dezimalkoordinaten-Angabe enthält und nach folgendem Muster aufgebaut ist:

```
<ReverseGeocode>
  <LatLng>
    <data><!--Decimal coordinate in format NN.NNNNNNNNNNNNNN,
EE.EEEEEEEEEEEEEEE--></data>
  </LatLng>
</ReverseGeocode>
```

Ausgabe

Für Operation `ReverseGeocode` wird ein XML- oder JSON-Dokument ausgegeben, das die Geoinformationen zur angegebenen Dezimalkoordinate beschreibt.

6.11.1.3 Operation: `GetDirections`

Mit Operation `GetDirections` lässt sich die Route zwischen einem Start- und Zielpunkt mit beliebig vielen Wegpunkten ermitteln.**Input**

Für Operation `GetDirections` erwartet der Adapter eine XML-Input-Struktur nach folgendem Muster, die Start-, Ziel und beliebige Wegpunkte als Dezimalkoordinaten oder Adresse enthält:

```
<Routing>
  <Origin>
    <!-- Address or GeoDecimal -->
    <Address>
      <Street><!--Street and number--></Street>
      <ZIP><!--ZIP code--></ZIP>
      <City><!--Place--></City>
      <Country><!--Country--></Country>
    </Address>
  </Origin>
  <Waypoints>
    <!-- Any number of waypoints -->
    <!-- Address or GeoDecimal -->
    <Waypoint>
      <Address>
        <Location><!--Location--></Location>
        <City><!--Place--></City>
      </Address>
    </Waypoint>
  </Waypoints>
  <Destination>
    <!-- Address or GeoDecimal -->
    <GeoDecimal>
      <Latitude><!--Latitude--></Latitude>
      <Longitude><!--Longitude--></Longitude>
    </GeoDecimal>
  </Destination>
</Routing>
```

Ausgabe

Für Operation `GetDirections` wird ein XML- oder JSON-Dokument ausgegeben, das eine für PKW berechnete Route beschreibt. Falls mehrere Routenvarianten bestehen, wird die schnellste Route ausgegeben.

6.11.1.4 Operation: CreateBingURL

Mit Operation `CreateBingURL` lässt sich eine URL zu einer *Bing Maps* Kartenausschnitts-Grafik erzeugen. **Input**


Für Operation `CreateBingURL` erwartet der Adapter eine XML-Input-Dokument, das im Element `Path` den URL-Part der *Bing Static Maps API* sowie in verschiedenen `Param`-Elementen den zu erzeugenden Kartenausschnitt definiert. Ein Parameter mit Attribut `name="mapSize"` enthält dabei die Pixelabmessungen des zu erzeugenden Kartenausschnitts.

Beispiel-Input

```
<BingMaps>
  <Path>Imagery/Map/Road/Routes</Path>
  <Param name="mapSize">640x640</Param>

  <Param name="wp.0">
    Seattle,WA;64;1
  </Param>

  <Param name="wp.1">
    Redmond,WA;66;2
  </Param>
</BingMaps>
```

 Für mehr Informationen zum Erzeugen und Formatieren von *Bing Maps* Kartenausschnitts-Grafiken, siehe Bing Static Map API unter <https://msdn.microsoft.com/en-us/library/ff701724.aspx>.

Ausgabe

Für Operation `CreateBingURL` wird ein XML- oder JSON-Dokument ausgegeben, das eine URL zu einem dem Input entsprechenden *Bing Maps*-Kartenausschnitt enthält:

```
<HREF>
  https://dev.virtualearth.net/REST/v1/Imagery/Map/Road/Routes?
  mapSize=640%2C640&wp.0=Seattle%2CWA%3B64%3B1&wp.
  1=Redmond%2CWA%3B66%3B2&key=AjGQc0kb94CdwAWZxnYndqRtpQRj7_Vmy4FDZoho2zyfHfyAcsDHx
  uaDLMpprFm-&o=xml
</HREF>
```

6.11.2 Geo Calculation



Geo Calculation: Konvertiert Linienzüge oder Polygone, deren Wegpunkte in den Formaten *GeoDecimal*, *GeoE5*, *GooglePoint*, *GoogleLine*, *PtvLine* oder *PtvPoint* angegeben werden, in die Formate *GeoDecimal*, *GeoE5*, *GooglePoint* oder *GoogleLine*, berechnet die Länge von Linienzügen (Routenlänge) und die Kantenlängen von Polygonen, ermittelt (Kanten-) Schnittpunkte zwischen Linienzügen/Polygonen (für Gebietsgrenzberechnungen) und ermittelt Begrenzungsrahmen für ein oder mehrere Polygone/Linienzüge (z.B. zur Bestimmung eines Kartenausschnitts).

Es können für eine Linie, die ein Polygon schneidet, die darin liegenden und außerhalb liegenden Liniensegmente bestimmt werden (z.B. zur Unterscheidung von Fahrten innerhalb und außerhalb eines Gebiets). Zudem lassen sich Linienzüge/Polygone vereinfachen, sodass für eine Kartendarstellung überflüssige Wegpunkte entfernt werden.



Einschränkungen

Bei Polygon-Berechnungen, die über mehr als die Hälfte einer Erdhalbkugel gehen (mehr als 180°), können ggf. unvorhergesehene Ergebnisse auftreten. Liniensegmente dürfen nicht mehr als 170° umfassen. Falls Sie derartige Operationen durchführen möchten, kontaktieren Sie zunächst den SoftProject-Support über support@softproject.de.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>LineConversion</i>: Linienzüge oder Polygone, deren Wegpunkte in den Formaten <i>GeoDecimal</i>, <i>GeoE5</i>, <i>GooglePoint</i>, <i>GoogleLine</i>, <i>PtvLine</i> oder <i>PtvPoint</i> angegeben werden, in die Formate <i>GeoDecimal</i>, <i>GeoE5</i>, <i>GooglePoint</i> oder <i>GoogleLine</i> konvertieren • <i>LineLength</i>: Länge von Linienzügen oder die gesamte Kantenlänge von Polygonen in Kilometern berechnen • <i>LineIntersection</i>: (Kanten-)Schnittpunkte zwischen Polygonen oder Linienzügen ermitteln • <i>BoundingBox</i>: Begrenzungsrahmen (kleinstmögliches Rechteck nach Mercator-Zylinder-Projektion in der <i>Ebene</i>) für Polygone oder Linienzüge ermitteln und jeweils mit Breiten-/Höhengrad der Eckpunkte ausgeben • <i>SegmentLineByPolygon</i>: Für einen Linienzug, der ein Polygon schneidet, die darin liegenden und außerhalb liegenden Liniensegmente bestimmen • <i>LineSimplificationPercent</i>: Linienzüge/Polygone vereinfachen, sodass für eine Kartendarstellung überflüssige Wegpunkte proportional zur Gesamtstrecke entfernt werden • <i>LineSimplificationKilometer</i>: Linienzüge/Polygone vereinfachen, sodass für eine Kartendarstellung überflüssige Wegpunkte entfernt werden
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter</i>. <i>geo.calc.GeoCalculationsAdapter</i>: Hauptklasse (Standard)</p>
<i>outputFormat</i>	<p>Geodatenformat, in dem das Ergebnis ausgegeben wird</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>GeoDecimal</i>: Geodezimal kodierte Koordinaten mit Breitengrad und Längengrad (Latitude/Longitude)(z.B. <i>48.952/8.391</i>) • <i>GeoE5</i>: <i>Geo E5</i>-kodierte Koordinaten mit Breitengrad und Längengrad (Latitude/Longitude)(z.B. <i>4895176/839096</i>) • <i>GooglePoint</i>: <i>GooglePoint</i>-kodierte Koordinaten (z. B. <i>o{wiHozer@}</i>) • <i>GoogleLine</i>: <i>GoogleLine</i>-kodierter Linienzug, bestehend aus <i>GooglePoint</i>-Wegpunkten (z.B. <i>o{wiHozer@or@ow@_yF?</i>)

<i>percent</i>	<p>Nur für Operation LineSimplificationPercent: Breite des Korridors, innerhalb dessen der Linienzug vereinfacht werden soll, relativ zur Gesamtstreckenlänge in Prozent geteilt durch 100</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige Zahl zwischen 0 und 1 mit Punkt als Dezimaltrennzeichen (z.B. 0.5) • 0.01: Innerhalb eines 1 Prozent breiten Korridors alle zusätzlichen Wegpunkte entfernen (Standard)
<i>kilometer</i>	<p>Nur für Operation LineSimplificationKilometer: Breite des Korridors, innerhalb dessen der Linienzug vereinfacht werden soll, in Kilometern</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive Zahl oder 0 mit Punkt als Dezimaltrennzeichen (z.B. 0.5) • 0.01: Innerhalb eines 10 Meter breiten Korridors alle zusätzlichen Wegpunkte entfernen (Standard)

Statuswerte

1 (successful)	Die Adapter-Operation wurde erfolgreich ausgeführt
0 (empty)	<p>Bei Operation LineIntersection: die Zahl der Linienzüge ist nicht 2</p> <p>Bei Operation SegmentLineByPolygon: der Input enthält kein Element Polygon, Polyline oder es fehlen beide Elemente, oder es liegt ein Fehler in der Geometrie vor (Polygon mit weniger als drei Punkte, Polyline mit weniger als zwei Punkten)</p>
-1 (failed)	Der Adapter-Input ist kein wohlgeformtes XML-Dokument

Input

Allgemein werden für diesen Adapter Geodaten für Punkte und Linien im Input-XML-Dokument einheitlich notiert. Geodezimal kodierte Koordinaten werden dabei wie folgt angegeben:

```
<GeoDecimal>
  <Latitude><!--Breitengrad im Format NN.NNNNN--></Latitude>
  <Longitude><!--Längengrad im Format EE.EEEEE--></Longitude>
</GeoDecimal>
```

Geo E5-kodierte Koordinaten werden stets als ganzzahlige Werte (geodezimal kodierter Koordinatenwert mit 100.000 multipliziert) angegeben:

```
<GeoE5>
  <Latitude><!--Breitengrad im Format NNNNNNN--></Latitude>
  <Longitude><!--Längengrad im Format EEEEEEE--></Longitude>
</GeoE5>
```

Google Point-kodierte Koordinaten werden wie folgt angegeben:

```
<GooglePoint><!--GooglePoint-Code--></GooglePoint>
```

Google Line-kodierte Linienzüge werden wie folgt definiert:

```
<GoogleLine><!--GoogleLine-Code--></GoogleLine>
```

PTV Point-kodierte Koordinaten (in *PTV*-Mercator-Projektion) werden wie folgt angegeben:

```
<PtvPoint>
  <X><!--"Easting" als Zahl ggf. mit Komma--></X>
  <Y><!--"Northing" als Zahl ggf. mit Komma--></Y>
</PtvPoint>
```

PTV Line-kodierte Linienzüge (in *PTV*-Mercator-Projektion) werden wie folgt angegeben. Dabei werden 1-n Wertepaare, bestehend aus "Easting" und "Northing" aneinandergefügt, sodass eine stets gerade Anzahl an Werten hinterlegt sind. Jeder Wert wird durch ein Komma getrennt.

```
<PtvLine><!--Easting1,Northing1,Easting...,Northing...--></PtvLine>
```

6.11.2.1 Operation: Line Conversion

Mit der Operation *LineConversion* lassen sich Linienzüge oder Polygone, deren Wegpunkte in den Formaten *GeoDecimal*, *GeoE5*, *GooglePoint*, *GoogleLine*, *PtvLine* oder

PtvPoint angegeben werden, in die Formate *GeoDecimal*, *GeoE5*, *GooglePoint* oder *GoogleLine* konvertieren.

Input

Für die Operation *LineConversion* erwartet der Adapter eine Input-XML-Struktur nach folgendem Muster, die beliebig viele Polygone (jeweils in einem Element *Polygon*) oder Linienzüge (jeweils in einem Element *Polyline*) enthalten kann, deren Eck-/Wegpunkte in den Formaten *GeoDecimal*, *GeoE5*, *GooglePoint*, *GoogleLine*, *PtvLine* oder *PtvPoint* innerhalb gleichnamiger Kindelemente (siehe oben) angegeben werden.

Für jedes Polygon oder für jeden Linienzug (*Polyline*) können Sie beliebige Attribute definieren, die ins Ausgabedokument unverändert übernommen werden.

```

<AnyRoot>
  <!-- 0-n Linienzüge -->
  <Polyline attribut="Wert"> <!-- Beliebige Attribute -->
    <!-- 1-n Punktdefinitionen in den Formaten GeoDecimal, GeoE5, GooglePoint oder
    PtvPoint-->
  </Polyline>

  <!-- 0-n Polygone -->
  <Polygon attribut="Wert"> <!-- Beliebige Attribute -->
    <!-- 1-n Punktdefinitionen in den Formaten GeoDecimal, GeoE5, GooglePoint oder
    PtvPoint-->
  </Polygon>
</AnyRoot>

```

Ausgabe

Für die Operation `LineConversion` wird ein XML-Dokument ausgegeben, das die gleiche Struktur wie das Input-Dokument besitzt und sämtliche Attribute, die in Element `Polygon` oder `Polyline` definiert werden, unverändert enthält. Je nach gewähltem Ausgabeformat werden die Punktangaben ins Format `GeoDecimal`, `GeoE5` oder `GooglePoint` konvertiert oder Linienzüge ins `GoogleLine`-Format konvertiert und in den entsprechenden Elementen hinterlegt.

6.11.2.2 Operation: `LineLength`

Mit Operation `LineLength` lässt sich Länge von Linienzügen oder die gesamte Kantenlänge von Polygonen in Kilometern berechnen.

Input

Für Operation `LineLength` erwartet der Adapter die gleiche Input-XML-Struktur wie für Operation `LineConversion` (siehe oben), in der Sie `Polygone` und/oder `Linienzüge` (*Polylines*) sowie ggf. zusätzliche Attribute definieren.

Ausgabe

Für Operation `LineLength` wird ein XML-Dokument nach folgendem Muster ausgegeben, das für jedes Element `Polygon` oder `Polyline`, das in der Input-XML-Struktur definiert wurde und ein `Polygon` oder einen `Linienzug` enthält, ein Element `Length` besitzt, das die `Linienlänge` (bei `Polygonen` dessen gesamte `Kantenlänge`) in Kilometern angibt.

Sämtliche Attribute, die im Input-XML in den Elementen `Polygon` oder `Polyline` definiert waren, werden unverändert ausgegeben.

```

<Data>
  <!-- Für jedes Polygon-/Polyline-Element -->
  <Length attribut="Wert"><!--Länge in Kilometern (mit 3 Nachkommastellen)--></
Length>
</Data>

```

6.11.2.3 Operation: LineIntersection

Mit Operation `LineIntersection` lassen sich (Kanten-)Schnittpunkte zwischen Polygonen oder Linienzügen ermitteln.

Input

Für Operation `LineIntersection` erwartet der Adapter die gleiche Input-XML-Struktur wie für Operation `LineConversion` (siehe oben), in der Sie zwei Polygone, zwei Linienzüge (*Polylines*) oder ein Polygon und einen Linienzug definieren. Zusätzliche Attribute werden jedoch nicht ins Ausgabe-XML-Dokument übernommen.

Ausgabe

Für Operation `LineIntersection` wird ein XML-Dokument ausgegeben, das in Element `Intersection` alle ermittelten (Kanten-)Schnittpunkte enthält. Je nach gewähltem Ausgabeformat werden dabei die Punktangaben in jeweils einem Kindelement `GeoDecimal`, `GeoE5` oder `GooglePoint` ausgegeben.

Alternativ kann über Ausgabeformat `GoogleLine` eine Linie in einem gleichnamigen Kindelement ausgegeben werden, die entlang dieser Schnittpunkte führt:

```
<Data>
  <Intersection>
    <!-- 0-n Schnittpunkte/-linien -->
    <Punkt-/Linienformat><!--Punkt-/Liniendefinition--></Punkt-/Linienformat>
  </Intersection>
</Data>
```

6.11.2.4 Operation: BoundingBox

Mit Operation `BoundingBox` lassen sich Begrenzungsrahmen (kleinstmögliches Rechteck nach Mercator-Zylinder-Projektion in der *Ebene*) für Polygone oder Linienzüge ermitteln und jeweils mit Breiten-/Höhengrad der Eckpunkte ausgeben.

Input

Für Operation `BoundingBox` erwartet der Adapter die gleiche Input-XML-Struktur wie für Operation `LineConversion` (siehe oben), in der Sie Polygone, Linienzüge (*Polylines*) sowie ggf. zusätzliche Attribute definieren.

Ausgabe

Für Operation `BoundingBox` wird ein XML-Dokument ausgegeben, das für jedes im Input-Dokument definierte Polygon oder für jeden Linienzug die Eckpunkte eines rechteckigen Begrenzungsrahmens enthält. Zudem enthält es sämtliche Attribute, die im Input-XML-Dokument in Element `Polygon` oder `PolyLine` definiert werden.

Die erforderlichen Eckpunkte (Breiten-/Höhengrad) zum Aufspannen des Begrenzungsrahmens werden wie folgt ausgegeben:

```

<Data>
  <!-- Für jedes Polygon-/Polyline-Element -->
  <BoundingBox attribut="Wert"> <!-- Beliebige Attribute -->
    <MinLatitude><!--Geodezimal kodierter Breitengrad von Eckpunkt 1--></
MinLatitude>
    <MaxLatitude><!--Geodezimal kodierter Breitengrad von Eckpunkt 2--></
MaxLatitude>
    <MinLongitude><!--Geodezimal kodierter Längengrad von Eckpunkt 1--></
MinLongitude>
    <MaxLongitude><!--Geodezimal kodierter Längengrad von Eckpunkt 2--></
MaxLongitude>
  </BoundingBox>
</Data>

```

6.11.2.5 Operation: SegmentLineByPolygon

Mit Operation `SegmentLineByPolygon` lassen sich für einen Linienzug, der ein Polygon schneidet, die darin liegenden und außerhalb liegenden Liniensegmente bestimmen.

Input

Für Operation `SegmentLineByPolygon` erwartet der Adapter die gleiche Input-XML-Struktur wie für Operation `LineConversion` (siehe oben), in der Sie *genau ein* Polygon in Element `Polygon` sowie *genau einen* Linienzug (*Polyline*) in Element `Polyline` definieren. Zusätzliche Attribute werden jedoch nicht ins Ausgabe-XML-Dokument übernommen.

Ausgabe

Für Operation `SegmentLineByPolygon` wird ein XML-Dokument ausgegeben, das in den Elementen `OriginalPolygon` und `OriginalPolyline` die Punkte des ursprünglichen Polygons und Linienzugs im gewählten Ausgabeformat sowie dessen (Kanten-)Länge in Kilometern in Attribut `length` enthält.

Innerhalb von Element `FinalPolygon` und `FinalPolyline` werden die korrigierten Punkte des Polygons und Linienzugs im gewählten Ausgabeformat sowie die (Kanten-)Länge des Polygons und des Linienzugs in Attribut `length` in Kilometern ausgegeben.

Innerhalb von Element `Inside` werden für jedes Liniensegment, das innerhalb des Polygons liegt, in einem Element `Part` sämtliche Punkte ausgegeben. Je nach gewähltem Ausgabeformat werden dabei die Punktangaben in jeweils einem Kindelement `GeoDecimal`, `GeoE5` oder `GooglePoint` ausgegeben. Zudem werden in jeweils einem Attribut `length` die Längen der innenliegenden Liniensegmente in Kilometern ausgegeben.

Innerhalb von Element `Outside` werden für jedes Liniensegment, das außerhalb des Polygons liegt, in einem Element `Part` sämtliche Punkte ausgegeben. Je nach gewähltem Ausgabeformat werden dabei die Punktangaben in jeweils einem Kindelement `GeoDecimal`, `GeoE5`, `GooglePoint` oder als `GoogleLine` ausgegeben. Zudem werden in jeweils einem Attribut `length` die Längen der außenliegenden Liniensegmente in Kilometern ausgegeben.

Innerhalb von Element `Edge` werden Linien ausgegeben, die sich exakt auf der Schnittlinie befinden.

```

<Result>
  <OriginalPolyline length="Länge des ursprünglichen Linienzugs in Kilometern">
    <!-- 0-n Punkte/Linien -->
    <Punkt-/Linienformat><!--Punkt-/Liniendefinition--></Punkt-/Linienformat>
  </OriginalPolyline>
  <OriginalPolygon length="Kantenlänge des ursprünglichen Polygons in Kilometern">
    <!-- 0-n Punkte/Linien -->
    <Punkt-/Linienformat><!--Punkt-/Liniendefinition--></Punkt-/Linienformat>
  </OriginalPolygon>
  <FinalPolyline length="Länge des korrigierten Linienzugs in Kilometern">
    <!-- 0-n Punkte/Linien -->
    <Punkt-/Linienformat><!--Punkt-/Liniendefinition--></Punkt-/Linienformat>
  </FinalPolyline>
  <FinalPolygon length="Kantenlänge des korrigierten Polygons in Kilometern">
    <!-- 0-n Punkte/Linien -->
    <Punkt-/Linienformat><!--Punkt-/Liniendefinition--></Punkt-/Linienformat>
  </FinalPolygon>
  <Inside>
    <!-- 0-n Liniensegmente -->
    <Part length="Länge des innenliegenden Linienzugs in Kilometern">
      <!-- 1-n Punkte/Linien -->
      <Punkt-/Linienformat><!--Punkt-/Liniendefinition--></Punkt-/Linienformat>
    </Part>
  </Inside>
  <Outside>
    <!-- 0-n Liniensegmente -->
    <Part length="Länge des außenliegenden Linienzugs in Kilometern">
      <!-- 1-n Punkte/Linien -->
      <Punkt-/Linienformat><!--Punkt-/Liniendefinition--></Punkt-/Linienformat>
    </Part>
  </Outside>
  <Edge>
    <!-- 0-n Liniensegmente -->
    <Part length="Länge des exakt auf der Linie liegenden Liniensegments in
Kilometern">
      <!-- 1-n Punkte/Linien -->
      <Punkt-/Linienformat><!--Punkt-/Liniendefinition--></Punkt-/Linienformat>
    </Part>
  </Edge>
</Result>

```

6.11.2.6 Operation: LineSimplificationPercent / LineSimplificationKilometer

Mit den Operationen LineSimplificationPercent und LineSimplificationKilometer lassen sich gemäß dem *Douglas-Peucker-Algorithmus* Linienzüge/Polygone vereinfachen, sodass für eine Kartendarstellung überflüssige Wegpunkte innerhalb eines Korridors entfernt werden. Die Korridorbreite wird relativ zur Gesamtstrecke bzw. anhand einer absoluten Korridorbreite in Kilometern definiert.

Input

Für Operation LineSimplificationPercent und LineSimplificationKilometer erwartet der Adapter die gleiche Input-XML-Struktur wie für Operation LineConversion (siehe oben), in der Sie beliebige Linienzüge (*Polyline*) in jeweils einem Element *Polyline* oder beliebige Polygone in jeweils

einem Element `Polygon` definieren. Zusätzliche Attribute werden ins Ausgabe-XML-Dokument übernommen.

Ausgabe

Für Operation `LineSimplificationPercent` und `LineSimplificationKilometer` wird ein XML-Dokument ausgegeben, das die gleiche Struktur wie das Input-XML-Dokument besitzt und sämtliche Attribute, die in Element `Polygon` oder `PolyLine` definiert werden, unverändert enthält. Je nach Korridorbreite (definiert in Parameter `percent` bzw. `kilometer`) werden die Linienzüge/Polygone entsprechend vereinfacht. Je nach gewähltem Ausgabeformat werden die Punktangaben ins Format `GeoDecimal`, `GeoE5` oder `GooglePoint` konvertiert oder Linienzüge ins `GoogleLine`-Format konvertiert und in den entsprechenden Elementen hinterlegt

6.11.3 Google Maps



Google Maps: Greift auf *Google Maps* über dessen Webservice-Schnittstelle zu und ermittelt zu einer Adresse die geografischen Informationen. Zudem kann eine Route inklusive Wegpunkten berechnet und eine URL zu einem *Google Maps*-Kartenausschnitt erzeugt werden. Der Adapter gibt seine Ergebnisse wahlweise als XML- oder JSON-Dokument aus.

i Standardmäßig wird die kostenfreie *Google Maps API* verwendet; Um sämtliche Leistungen uneingeschränkt zu nutzen, können Sie mit diesem Adapter auch die *Google Maps API Premier* verwenden, indem Sie entsprechende Zugangsdaten in den Adapterparametern `clientID` und `privateKey` hinterlegen.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>Geocode</i> : Zu einer Adresse sämtliche Geoinformationen (Anschrift, Bezirk/Land, Nation, Koordinaten etc.) ermitteln • <i>ReverseGeocode</i>: Zu einer Dezimalkoordinate sämtliche Geoinformationen (Anschrift, Bezirk/Land, Nation, Koordinaten etc.) ermitteln • <i>GetDirections</i>: Route zwischen einem Start- und Zielpunkt mit beliebig vielen Wegpunkten ermitteln • <i>CreateGoogleURL</i>: URL zu einer <i>Google Maps</i>-Kartenausschnitts-Grafik erzeugen (bei Nutzung der <i>Google Maps API Premier</i> zudem mit signierter URL)
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <code>de.softproject.integration.adapter.geo.google.GoogleMapsAdapter</code> : Hauptklasse (Standard)
----------------	---

<i>clientID</i>	<p>Kundennummer (Client ID) zur Nutzung der <i>Google Maps API Premier</i></p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige gültige Zeichenkette • (<i>leer</i>): Kostenlose <i>Google Maps API</i> verwenden (Standard)
<i>privateKey</i>	<p>Private Key für das <i>Google Maps API Premier</i> (wenn Parameter <i>clientID</i> gesetzt ist)</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette</p>
<i>outputFormat</i>	<p><i>Google Maps</i>-Ausgabeformat für Anfragen definieren</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>xml</i>: XML-Dokument ausgeben • <i>json</i>: JSON-Dokument ausgeben
<i>timeout</i>	<p>Zeitbeschränkung für den Verbindungsaufbau in Millisekunden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl • <i>0</i>: Keine Zeitbeschränkung • <i>20000</i>: Nach 20 Sekunden den Verbindungsaufbau abbrechen
<i>pxHost</i>	<p>IP-Adresse oder Host-Name eines Proxy-Servers</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette (z.B. 192.168.1.1)</p>
<i>pxPort</i>	<p>Portnummer des Proxy-Servers</p> <p>Mögliche Werte: Beliebige ganze Zahl</p>
<i>pxLogin</i>	<p>Benutzername für den Proxy-Server</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>pxPwd</i>	<p>Passwort für den Proxy-Server</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>

Statuswerte

<i>1(successful)</i>	Die Anfrage war erfolgreich und ein Ergebnis wurde zurückgegeben
<i>0(empty)</i>	Die Anfrage war zwar erfolgreich, jedoch wurde kein Ergebnis zurückgegeben
<i>-1(failed)</i>	Die Anfrage schlug aufgrund eines technischen Fehlers fehl

6.11.3.1 Operation: Geocode

Mit der Operation Geocode lassen sich zu einer Adresse sämtliche Geoinformationen (Anschrift, Bezirk/Land, Nation, Koordinaten etc.) ermitteln.

Input

Für Operation Geocode erwartet der Adapter eine XML-Input-Struktur nach folgendem Muster, die eine vollständige Anschrift enthält (bei unvollständigen Anschriften ist die Anfrage ggf. nicht eindeutig):

```
<Geocode>
  <Address>
    <Street><!--Straße und Hausnummer--></Street>
    <ZIP><!--Postleitzahl--></ZIP>
    <City><!--Ort--></City>
    <Country><!--Land--></Country>
  </Address>
</Geocode>
```

Ausgabe

Für Operation Geocode wird ein XML- oder JSON-Dokument ausgegeben, das die Geoinformationen (Anschrift, Bezirk/Land, Nation, Koordinaten etc.) zur angegebenen Adresse gemäß <http://code.google.com/intl/en/apis/maps/documentation/geocoding/#XML> beschreibt.

6.11.3.2 Operation: ReverseGeocode

Mit Operation ReverseGeocode lassen sich zu einer Dezimalkoordinate sämtliche Geoinformationen (Anschrift, Bezirk/Land, Nation, Koordinaten etc.) ermitteln.

Input

Für Operation ReverseGeocode erwartet der Adapter eine XML-Input-Struktur nach folgendem Muster, die eine Dezimalkoordinatenangabe enthält:

```
<ReverseGeocode>
  <LatLng>
    <data><!--Dezimalkoordinate im Format NN.NNNNNNN, EE.EEEEEEE--></data>
  </LatLng>
</ReverseGeocode>
```

Ausgabe

Für Operation ReverseGeocode wird ein XML- oder JSON-Dokument ausgegeben, das die Geoinformationen (Anschrift, Bezirk/Land, Nation, Koordinaten etc.) zur angegebenen Dezimalkoordinate gemäß <http://code.google.com/intl/en/apis/maps/documentation/geocoding/#XML> beschreibt.

6.11.3.3 Operation: GetDirections

Mit Operation GetDirections lässt sich die Route zwischen einem Start- und Zielpunkt mit beliebig vielen Wegpunkten ermitteln.

Input

Für Operation GetDirections erwartet der Adapter eine XML-Input-Struktur nach folgendem Muster, die Start-, Ziel und beliebige Wegpunkte als Dezimalkoordinaten, Adresse oder im GooglePoint-Format enthält:

```
<GoogleRouting xmlns:exslt="http://exslt.org/common">
  <Origin>
    <!--Dezimalkoordinate, Adresse/POI oder GooglePoint-->
  </Origin>
  <Waypoints>
    <!-- Any number of waypoints -->
    <Waypoint>
      <!--Dezimalkoordinate, Adresse/POI oder GooglePoint-->
    </Waypoint>
  </Waypoints>
  <Destination>
    <!--Dezimalkoordinate, Adresse/POI oder GooglePoint-->
  </Destination>
</GoogleRouting>
```

Adressen werden in folgender XML-Struktur angegeben:

```
<Address>
  <Street><!--Straße und Hausnummer--></Street>
  <ZIP><!--Postleitzahl--></ZIP>
  <City><!--Ort--></City>
  <Country><!--Land--></Country>
</Address>
```

GooglePoints werden in folgendem XML-Element angegeben:

```
<Address>
  <Location><!--Point of Interest--></Location>
  <City><!--Ort--></City>
</Address>
```

GooglePoints werden in folgendem XML-Element angegeben:

```
<GooglePoint><!--GooglePoint-Code--></GooglePoint>
```

Dezimalkoordinaten werden in folgender XML-Struktur angegeben:

```
<GeoDecimal>
  <Latitude><!--NN.NNNNNNN--></Latitude>
  <Longitude><!--EE.EEEEEEE--></Longitude>
</GeoDecimal>
```

Ausgabe

Für Operation `GetDirections` wird ein XML- oder JSON-Dokument ausgegeben, das eine für PKW berechnete Route gemäß <http://code.google.com/intl/en/apis/maps/documentation/directions/#XML> beschreibt. Falls mehrere Routenvarianten bestehen, wird die schnellste Route ausgegeben.

6.11.3.4 Operation: CreateGoogleURL

Mit Operation `CreateGoogleURL` lässt sich eine URL zu einer *Google Maps*-Kartenausschnitts-Grafik erzeugen (bei Nutzung der *Google Maps API Premier* zudem mit signierter URL).

Input

Für Operation `CreateGoogleURL` erwartet der Adapter eine XML-Input-Struktur nach folgendem Muster, die in Element `Path` den URL-Part des *Google Static Maps APIs* (siehe <http://code.google.com/intl/de-DE/apis/maps/documentation/staticmaps/>) sowie in verschiedenen `Param`-Elementen den zu erzeugenden Kartenausschnitt definiert. Ein Parameter mit Attribut `name="size"` enthält dabei die Pixelabmessungen des zu erzeugenden Kartenausschnitts.

Jedes `Param`-Element mit Attribut `name="markers"` definiert eine Stecknadelmarkierung mit Farbe (z. B. `color:blue`) und Beschriftung (ein Großbuchstabe, z. B. `label:X`).

Ein `Param`-Element mit Attribut `name="path"` definiert, jeweils durch `|` getrennt, Pfad-Eigenschaften (z. B. Farbe `color:blue`) sowie eine mit dem Präfix `enc:` versehene Route, bestehend aus *GoogleLine*-kodierte Wegpunkten:

```
<GoogleMaps>
  <Path><!--/maps/api/staticmap--></Path>
  <Param name="size"><!--640x640--></Param>

  <!-- für jede Stecknadel-Markierung -->
  <Param name="markers">
    <!--color:Farbe|label:Großbuchstabe|Latitude,Longitude-->
  </Param>

  <!-- für jeden Pfad; Parameter getrennt durch "|" -->
  <Param name="path">
    <!--Parametername:Parameterwert|enc:GoogleLine-kodierter Pfad-->
  </Param>
</GoogleMaps>
```

Ausgabe

Für Operation CreateGoogleURL wird ein XML- oder JSON-Dokument ausgegeben, das eine URL (siehe <http://code.google.com/intl/de-DE/apis/maps/documentation/staticmaps/>) zu einem dem Input entsprechenden *Google Maps*-Kartenausschnitt enthält (Achtung: &-Zeichen werden in XML-Dokument als Entität gespeichert):

```
<HREF>http://maps.google.de/Service-URL-Part?sensor=false
    &markers=color:Stecknadelfarbe|label:Buchstabe|Latitude,Longitude
    &markers=color:Stecknadelfarbe|label:Buchstabe|Latitude,Longitude
    &path=color:Pfadfarbe|enc:GoogleLine-kodierter Pfad&size=Bildabmessungen
</HREF>
```

6.12 Insurance

6.12.1 Audatex DV90 Converter



Audatex DV90 Converter: Liest ein *Audatex DV90*-Kalkulationsdokument ein und gibt dessen Daten als XML-Dokument aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation dieser Native Adapter durchführt Mögliche Werte: <i>ConvertToXml</i> : Audatex DV90-Textdokument einlesen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.nativeadapter.NativeAdapter</i> : Hauptklasse (Standard)
----------------	---

Statuswerte

1 (<i>successful</i>)	Das eingelesene <i>DV90</i> -Kalkulationsdokument wurde erfolgreich konvertiert.
-1 (<i>failed</i>)	Die Konvertierung ist aufgrund eines technischen Fehlers fehlgeschlagen.

6.12.2 GDV

6.12.2.1 GDV Message Plausibility Checker



GDV Message Plausibility Checker: Prüft eine GDV-Nachricht auf Plausibilität gemäß den Regeln einer Konfigurationsdatei

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>Execute</i> : GDV-Nachricht auf Plausibilität prüfen, gemäß den Regeln einer in Parameter <i>iniFile</i> festgelegten Konfigurationsdatei
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.gdvplausi.gdvPlausiPrueferWrapper</i> : Hauptklasse (Standard)
<i>iniFile</i>	Pfad zur Konfigurationsdatei (.ini-Format) Mögliche Werte: Pfad und Dateiname (z.B. <i>/Project/GDV.ini</i>)

6.12.2.2 GDV Message Reader



GDV Message Reader: Liest eine GDV-Nachricht ein, parst sie gemäß den Regeln einer Deskriptordatei und gibt ein XML-Dokument mit UTF-8-Zeichenkodierung aus. Unterstützt werden die Nachrichtentypen SN 2000 / 2003, VU/VM 2009 / 2007 / 2005 / 2002 / 1998.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Execute</i> : GDV-Nachricht einlesen und in XML konvertieren, gemäß den Regeln einer in Parameter <i>descriptorFile</i> festgelegten Deskriptor-Datei
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.ASCIIGDV</i> : Hauptklasse (Standard)
<i>DescFile</i>	Deskriptordatei, die die GDV-Nachricht in ihrer Datenstruktur beschreibt, siehe <i>Handbuch X4 Designer – ASCII-Deskriptor</i> . Die Deskriptordatei sollte im Benutzerordner des X4 Repositorys abgelegt sein (z.B. <i>X4DB/1/Project/Descr.ad</i>) Mögliche Werte: Pfad und Dateiname (z.B. <i>/Project/Descr.ad</i>)

6.12.2.3 GDV Message Writer



GDV Message Writer: Konvertiert ein XML-Dokument gemäß den Regeln einer Deskriptordatei in eine GDV-Nachricht.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Execute</i> : GDV-Nachricht aus einem XML-Dokument generieren, gemäß den Regeln einer in Parameter <i>DescFile</i> festgelegten Deskriptor-Datei
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.ASCIIGDV</i> : Hauptklasse (Standard)
<i>DescFile</i>	Name der Deskriptordatei, die die GDV-Nachricht in ihrer Datenstruktur beschreibt, siehe <i>Handbuch X4 Designer – ASCII-Deskriptor</i> . Die Deskriptordatei sollte im Benutzerordner des X4 Repositorys abgelegt sein (z.B. <i>X4DB/1/Project/Descr.ad</i>) Mögliche Werte: Pfad und Dateiname (z.B. <i>/Project/Descr.ad</i>)

6.12.2.4 GDV VU/VM

GDV VU/VM: Die GDV VU/VM-Bibliothek stellt Prozesse und Adapter bereit, mit denen GDV VU/VM-Daten in verschiedenen Versionen von und nach XML konvertiert werden können.

i GDV VU/VM wird Ihnen als X4-Bibliothek bereitgestellt, das Sie zunächst in Ihr X4 Repository importieren müssen.

Verzeichnisstruktur

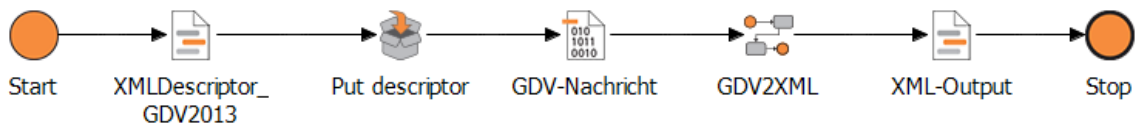
Die Bibliothek hat nach dem Import folgende Verzeichnisstruktur:

Struktur im X4 Repository	Ordner und Inhalte
<ul style="list-style-type: none"> ▼ GDV <ul style="list-style-type: none"> ▼ VU-Vermittler <ul style="list-style-type: none"> ▼ Adapter <ul style="list-style-type: none"> Collector.fa Text2Lines.fa ▼ Data <ul style="list-style-type: none"> > Descriptor > Templates ▼ Mapping <ul style="list-style-type: none"> DecodeLine.xsl GDVASCIIWriter.xsl GroupByVertrag.xsl ▼ Process <ul style="list-style-type: none"> > Example <ul style="list-style-type: none"> GDV2XML.wrf XML2GDV.wrf 	<p>Adapter</p> <ul style="list-style-type: none"> <i>X4Documents Collector (Collector.fa) und Text to Lines Converter(Text2Lines.fa)</i> <div style="border: 1px solid #f9e79f; padding: 10px; margin: 10px 0;"> <p>⚠ Beachten Sie, dass die zwei Adapter Collector.fa und Text2Lines.fa nicht verändert werden dürfen.</p> </div> <p>Data</p> <p>Ordner Descriptor:</p> <ul style="list-style-type: none"> Descriptor_*.xml: Deskriptor-Dateien zum Schreiben der GDV-Datei XMLDescriptor_*.xml: Deskriptor-Dateien zum Lesen der GDV-Datei Verzeichnisse 2002–2013: Verzeichnisse mit beschreibenden PDF-Dokumenten zum entsprechenden GDV-Release <p>Ordner Templates: Daten-Templates</p> <p>Mapping</p> <p>Die Mappings dienen zur Konvertierung der GDV VU/VM ASCII-Datensätze in lesbare XML-Datenstrukturen.</p> <div style="border: 1px solid #f9e79f; padding: 10px; margin: 10px 0;"> <p>⚠ Beachten Sie, dass die Mappings nicht verändert werden dürfen!</p> </div> <p>Process</p> <ul style="list-style-type: none"> Ordner Example: Beispielprozesse inklusive Beispieldaten für das Konvertieren zwischen dem XML-Format und GDV VU/VM ASCII-Datensätze GDV2XML.wrf: Prozess zum Konvertieren der GDV VU/VM ASCII-Datensätze nach XML XML2GDV.wrf: Prozess zum Konvertieren der XML-Daten in GDV VU/VM ASCII-Datensätze

ⓘ Achten Sie bitte beim Lesen und Schreiben des GDV-Formats darauf, ggf. notwendige Änderungen an einem Deskriptor auch am entsprechenden XML-Deskriptor vorzunehmen (und umgekehrt).

Konvertierung von GDV nach XML

1. In einem X4-Prozess den gewünschten XML-Deskriptor(XMLDescriptor_*.xml) über die Operation Read einlesen.
2. Anschließend den Deskriptor in eine Kontextvariable descriptor über die Operation Put laden.
3. Die gewünschte Datei im GDV-Format als Input für den Subprozess GDV2XML.wrf einfügen.



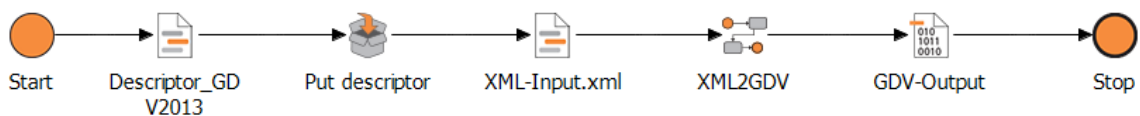
4. Das Ergebnis der Konvertierung über die Operation `write` in eine XML-Datei schreiben.
5. Prozess ausführen.

Das Ergebnis der Ausführung ist die XML-Repräsentation der Input-Datei.

- ✓ Bei Bedarf können Sie eine zusätzliche Transformation ausführen, welche die leeren Elemente entfernt.

Konvertierung von XML nach GDV

1. In einem X4-Prozess den gewünschten Deskriptor(Descriptor_GDV*.xml) über die Operation Read einlesen.
2. Anschließend den Deskriptor in eine Kontextvariable descriptor über die Operation Put laden.
3. Die gewünschte Datei im XML-Format als Input für den Subprozess XML2GDV.wrf einfügen.



4. Das Ergebnis der Konvertierung über die Operation `write` in eine Text-Datei schreiben.
5. Prozess ausführen.

Das Ergebnis der Ausführung ist die GDV-Repräsentation der Input-Datei.

- ✓ Bei Bedarf können Sie eine zusätzliche Transformation ausführen, welche die leeren Elemente entfernt.

6.12.3 SZF Reader



SZF Reader: Liest ein *SilverDAT II*-Kalkulationsdokument ein und gibt die Daten als XML-Dokument aus.

Benennung des Adapters bis Version 5.5.4: *SZF Converter*

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Read: SilverDAT II</i> -Textdokument einlesen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.szfreader.SzfReader</i> : Hauptklasse (Standard)
----------------	---

6.13 Internet of Things

6.13.1 IEC 61850 Client



IEC 61850 Client: Verbindet sich zu einem MMS-Server und liest die Geräte- und Knoteninformationen aus.

Der Connector stellt damit eine Basis-Schnittstelle für den weltweit etablierten Kommunikationsstandard IEC 61850 für die Automatisierung von Schaltanlagen zur Verfügung. Er agiert als Client und kann Nachrichten als klassische Client-Server-Kommunikation empfangen.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>RequestServerNodes</i>: Nach bestimmten Geräten und/oder Knoten suchen und die Werte auslesen • <i>RequestServerModel</i>: Beim MMS-Server das Geräte- und Knotenverzeichnis abfragen und als XML ausgeben
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.iec.IEC61850</i> : Hauptklasse (Standard)
<i>address</i>	IP-Adresse oder Host-Name des MMS-Servers Mögliche Werte: Beliebige gültige Zeichenkette (z. B. <i>192.168.1.1</i>)
<i>port</i>	Portnummer des MMS-Servers Mögliche Werte: Beliebige ganze Zahl
<i>authenticationParameter</i>	Optionalen Anmeldeparameter zur Anmeldung am MMS-Server

<i>responseTimeout</i>	Zeitbeschränkung für die Server-Anfrage in Millisekunden Mögliche Werte: <ul style="list-style-type: none"> • Beliebige ganze Zahl • 0: Keine Zeitbeschränkung • 20000: Nach 20 Sekunden die Server-Anfrage abbrechen
<i>messageFragmentTimeout</i>	Zeitbeschränkung zwischen zwei eingehenden Datenpaketen in Millisekunden Mögliche Werte: <ul style="list-style-type: none"> • Beliebige ganze Zahl • 0: Keine Zeitbeschränkung • 20000: Nach 20 Sekunden die Verbindung abbrechen
<i>maxTPDU</i>	Setzt die maximale TPDU (Transfer Protocol Data Unit) Mögliche Werte: Beliebige ganze Zahl
<i>maxMMSPDU</i>	Setzt die maximale MMS PDU Mögliche Werte: Beliebige ganze Zahl zwischen 64 und 65000

Statuswerte

<i>1(successful)</i>	Die Anfrage war erfolgreich und ein Ergebnis wurde zurückgegeben
<i>-1(failed)</i>	Die Anfrage schlug aufgrund eines technischen Fehlers fehl

Input

Operation RequestServerModel

Die Operation RequestServerModel benötigt kein Input-XML.

Operation RequestServerNodes

Die Operation RequestServerNodes erwarten ein Input-XML, das wie folgt aufgebaut ist:

```

<RootElement>
  <!-- @Fc ist optional; @ObjectReference ist erforderlich -->
  <Node ObjectReference="ied1lDevice1/LPHD1"/>      <!-- Liefert alle Knoten und
Geräte innerhalb von ied1lDevice1/LPHD1 -->
  <Node ObjectReference="ied1lDevice1/LPHD1" Fc="MX"/><!-- Liefert alle Knoten und
Geräte innerhalb von ied1lDevice1/LPHD1 mit dem Function Code MX -->
</RootElement>

```

Ausgabe

Operation RequestServerModel

Die Operation RequestServerModel gibt ein XML-Dokument aus, das wie folgt aufgebaut ist:

```
<?xml version="1.0" encoding="UTF-8"?>
<ServerModel>
  <LogicalDevice Name="ied1lDevice1" ObjectReference="ied1lDevice1">
    <LogicalNode Name="LLN0" ObjectReference="ied1lDevice1/LLN0">
      <FcDataObject Name="Mod" ObjectReference="ied1lDevice1/LLN0.Mod"
        Fc="ST">
        <BdaQuality Name="q" ObjectReference="ied1lDevice1/LLN0.Mod.q"
          Fc="ST" />
        <BdaTimestamp Name="t" ObjectReference="ied1lDevice1/LLN0.Mod.t"
          Fc="ST" />
      </FcDataObject>
      <FcDataObject Name="Beh" ObjectReference="ied1lDevice1/LLN0.Beh"
        Fc="ST">
        <BdaInt32 Name="stVal" ObjectReference="ied1lDevice1/LLN0.Beh.stVal"
          Fc="ST" />
        <BdaQuality Name="q" ObjectReference="ied1lDevice1/LLN0.Beh.q"
          Fc="ST" />
        <BdaTimestamp Name="t" ObjectReference="ied1lDevice1/LLN0.Beh.t"
          Fc="ST" />
      </FcDataObject>
    </LogicalNode>
  </LogicalDevice>
</ServerModel>
```

Operation RequestServerNodes

Die Operation RequestServerNodes gibt ein XML-Dokument aus. Dabei wird das Input-XML in den jeweiligen Knoten um die Antwort erweitert:

```

<?xml version="1.0" encoding="UTF-8"?>
<RootElement>
  <Node ObjectReference="ied1lDevice1/LPHD1" Fc="MX">
    <NoResult />
  </Node>
  <Node ObjectReference="ied1lDevice1/LPHD1" Fc="DC">
    <LogicalNode Name="LPHD1" ObjectReference="ied1lDevice1/LPHD1">
      <FcDataObject Name="PhyNam" ObjectReference="ied1lDevice1/LPHD1.PhyNam"
        Fc="DC">
        <BdaVisibleString Name="vendor"
          ObjectReference="ied1lDevice1/LPHD1.PhyNam.vendor" Fc="DC" Value="
"
          MaxLength="255" StringValue="" />
        </FcDataObject>
      </LogicalNode>
    </Node>
    <Node ObjectReference="ied1lDevice1/LPHD1">
      <LogicalNode Name="LPHD1" ObjectReference="ied1lDevice1/LPHD1">
        <FcDataObject Name="PhyHealth" ObjectReference="ied1lDevice1/
LPHD1.PhyHealth"
          Fc="ST">
            <BdaInt32 Name="stVal" ObjectReference="ied1lDevice1/
LPHD1.PhyHealth.stVal"
              Fc="ST" Value="0" />
            <BdaQuality Name="q" ObjectReference="ied1lDevice1/LPHD1.PhyHealth.q"
              Fc="ST" Value="AAA=" Overflow="false" Inconsistent="false"
              Validity="GOOD" OperatorBlocked="false" Inaccurate="false"
             OldData="false"
              Oscillatory="false" BadReference="false" Failure="false"
              Substituted="false" OutOfRange="false" Test="false" />
            <BdaTimestamp Name="t" ObjectReference="ied1lDevice1/
LPHD1.PhyHealth.t"
              Fc="ST" Value="AAAAAAAAAAAA=" Date="1970-01-01T00:00:00Z"
              LeapSecondsKnown="false" ClockNotSynchronized="false"
              TimeAccuracy="0"
              ClockFailure="false" />
            </FcDataObject>
            <FcDataObject Name="Proxy" ObjectReference="ied1lDevice1/LPHD1.Proxy"
              Fc="ST">
                <BdaBoolean Name="stVal" ObjectReference="ied1lDevice1/
LPHD1.Proxy.stVal"
                  Fc="ST" Value="false" />
                <BdaQuality Name="q" ObjectReference="ied1lDevice1/LPHD1.Proxy.q"
                  Fc="ST" Value="AAA=" Overflow="false" Inconsistent="false"
                  Validity="GOOD" OperatorBlocked="false" Inaccurate="false"
                 OldData="false"
                  Oscillatory="false" BadReference="false" Failure="false"
                  Substituted="false" OutOfRange="false" Test="false" />
                <BdaTimestamp Name="t" ObjectReference="ied1lDevice1/LPHD1.Proxy.t"
                  Fc="ST" Value="AAAAAAAAAAAA=" Date="1970-01-01T00:00:00Z"
                  LeapSecondsKnown="false" ClockNotSynchronized="false"
                  TimeAccuracy="0"
                  ClockFailure="false" />
                </FcDataObject>
                <FcDataObject Name="PhyNam" ObjectReference="ied1lDevice1/LPHD1.PhyNam"
                  Fc="DC">

```

```

        <BdaVisibleString Name="vendor"
            ObjectReference="ied1lDevice1/LPHD1.PhyNam.vendor" Fc="DC" Value="
"
            MaxLength="255" StringValue="" />
        </FcDataObject>
    </LogicalNode>
</Node>
</RootElement>

```

6.13.2 Modbus TCP Connector

ModBus TCP Connector: Verbindet sich mit Geräten über das Modbus-TCP-Protokoll.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Read: Liest Daten vom Modbus-Server aus • Write: Schreibt Daten auf dem Modbus-Server
-----------	---

Parameter

Adapter	<p>Adapter-Hauptklasse (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.modbus.ModBusConnectorAdapter</i>: Hauptklasse (Standard)</p>
ip	Server-IP
port	<p>Server-Port</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebiger Port • 502 (Standard)

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt
-1	Es kam zu einem Fehler während der Ausführung

Input

Der Adapter erwartet eine vordefinierte XML-Struktur für die verschiedenen Operationen:

- *Operation Read:*

Beispiel-Input für die Operation READ

```
<?xml version="1.0" encoding="UTF-8"
<Request>
  <InputRegisters count="2" ref="0"/>
  <InputRegisters count="1" ref="3"/>
  <HoldingRegisters count="2" ref="0"/>
  <HoldingRegisters count="1" ref="3"/>
  <Discretes count="2" ref="0"/>
  <Discretes count="1" ref="3"/>
  <Coils count="2" ref="0"/>
  <Coils count="1" ref="3"/>
</Request>
```

- *Operation Write:*

Beispiel-Input für die Operation WRITE

```
<?xml version="1.0" encoding="UTF-8"
<Request>
  <HoldingRegister ref="11">14</HoldingRegister>
  <Coil ref="20">true</Coil>
  <Coil ref="000E1">TRUE</Coil>
</Request>
```

Ausgabe

Der Adapter gibt je nach Operation verschiedene Ergebnisse aus:

- *Operation Read:*

Beispiel-Ausgabe für die Operation READ

```

<?xml version="1.0" encoding="UTF-8"
<Request>
  <InputRegisters count="2" ref="0">
    <Result>
      <InputRegister ref="0">0</InputRegister>
      <InputRegister ref="1">0</InputRegister>
    </Result>
  </InputRegisters>
  <InputRegisters count="1" ref="3">
    <Result>
      <InputRegister ref="0">0</InputRegister>
    </Result>
  </InputRegisters>
  <HoldingRegisters count="2" ref="0">
    <Result>
      <HoldingRegister ref="0">0</HoldingRegister>
      <HoldingRegister ref="1">0</HoldingRegister>
    </Result>
  </HoldingRegisters>
  <HoldingRegisters count="1" ref="3">
    <Result>
      <HoldingRegister ref="0">0</HoldingRegister>
    </Result>
  </HoldingRegisters>
  <Discretes count="2" ref="0">
    <Result>
      <Discrete ref="0">>false</Discrete>
      <Discrete ref="1">>false</Discrete>
    </Result>
  </Discretes>
  <Discretes count="1" ref="3">
    <Result>
      <Discrete ref="0">>false</Discrete>
    </Result>
  </Discretes>
  <Coils count="2" ref="0">
    <Result>
      <Coil ref="0">>false</Coil>
      <Coil ref="1">>false</Coil>
    </Result>
  </Coils>
  <Coils count="1" ref="3">
    <Result>
      <Coil ref="0">>false</Coil>
    </Result>
  </Coils>
</Request>

```

- *Operation Write:*

Beispiel-Ausgabe für die Operation WRITE

```
<?xml version="1.0" encoding="UTF-8"
<Request>
  <HoldingRegister ref="11">14</HoldingRegister>
  <Coil ref="20">true</Coil>
  <Coil ref="000E1">TRUE</Coil>
</Request>
```

Beispiel-Ausgabe im Falle eines Fehlers

```
<?xml version="1.0" encoding="UTF-8"
<listErrors description="Wrong Input values for the elements of this list">
  <HoldingRegister ref="11">xx</HoldingRegister>
  <Coil ref="xxxx">true</Coil>
  <Coil ref="000B">45</Coil>
</listErrors>
```

6.13.3 MQTT Publisher/Subscriber

- MQTT Publisher
- MQTT Subscriber

6.13.3.1 MQTT Publisher

MQTT Publisher: Ermöglicht das Veröffentlichen von Nachrichten in einem MQTT-Broker.

Das MQTT-Protokoll bietet eine einfache Methode zur Nachrichtenübermittlung über ein Publish/Subscribe-Modell. Damit eignet es sich besonders für die Nachrichtenübermittlung im Umfeld des Internet der Dinge, wie z. B. bei Sensoren mit niedriger Leistung oder mobilen Geräten wie Telefonen, eingebetteten Computern oder Mikrocontrollern.





 Der *MQTT Publisher* wird meist in Verbindung mit dem *MQTT Subscriber* verwendet.

Eigenschaften (Properties)


Operation	Vom Adapter ausgeführte Operation Mögliche Werte: <ul style="list-style-type: none"> • Publish: Nachricht in einem MQTT-Broker veröffentlichen
-----------	---

Parameter

Adapter	<p>Adapter-Hauptklasse (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.mqtt.MqttPublisherAdapter</i>: Hauptklasse (Standard)</p>
Alias	<p>Broker-Alias</p> <p>Mögliche Werte: Beliebige Zeichenkette</p> <div> <p> Der Alias kann innerhalb der Konfigurationsdatei <code>mqttconfig.xml</code> als Attribut des Elements <code>Broker</code> definiert werden.</p> </div>
ClientId	<p>Bezeichner für den Client, der im Broker veröffentlicht</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> Falls die Client ID in der Konfigurationsdatei <code>mqttconfig.xml</code> definiert ist: beliebige Zeichenkette Falls die Client ID nicht in der Konfigurationsdatei <code>mqttconfig.xml</code> definiert ist: UUID <div> <p> Die Client ID kann innerhalb der Konfigurationsdatei <code>mqttconfig.xml</code> für jeden Broker im Element <code>ClientId</code> definiert werden.</p> </div>
clientTimeout	<p>Timeout für <code>ONLINE_ON_DEMAND</code>-Verbindungen (für die Operation <code>Publish</code>). Wenn <code>ConnectionMode</code> von <code>ONLINE_ON_DEMAND</code> abweicht, wird <code>clientTimeout</code> nicht berücksichtigt.</p> <p>Mögliche Werte: Ganzzahl größer als 0</p>

connectionMode	<p>Verbindungsmodus, der zum Erstellen des Clients ausgewählt wurde. Abhängig vom gewählten Wert bleibt die Client-Verbindung erhalten oder wird nach einer bestimmten Zeit wieder getrennt (ClientTimeout).</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ONLINE_ON_DEMAND: Der Parameter ClientTimeout wird benötigt, um den Client zu konfigurieren. • ALWAYS_ONLINE_ON_FIRST_PUBLISH: Der Client bleibt verbunden, sobald eine erste Veröffentlichung erfolgt. Daher wird der Parameter ClientTimeout nicht benötigt. <div data-bbox="603 701 1433 898"> <p> Dieser Parameter wird nicht benötigt, wenn alias und clientId in der Konfigurationsdatei definiert sind oder die Werte für alias und clientId bereits verwendet wurden, um eine Verbindung herzustellen.</p> </div>
stateful	<p>Legt fest, ob der Zustand zwischen dem Client und dem Broker zustandsorientiert ist.</p> <p>Mögliche Werte: true / false</p> <div data-bbox="603 1086 1433 1283"> <p> Dieser Parameter wird nicht benötigt, wenn alias und clientId in der Konfigurationsdatei definiert sind oder die Werte für alias und clientId bereits verwendet wurden, um eine Verbindung herzustellen.</p> </div>
keystorePath	<p>Pfad zum Keystore. Wenn keine SSL/TLS-Konfiguration benötigt wird, um eine Verbindung zum Broker herzustellen, dann kann der Parameterwert leer sein.</p> <p>Mögliche Werte: Dateipfad</p> <div data-bbox="603 1496 1433 1693"> <p> Dieser Parameter wird nicht benötigt, wenn alias und clientId in der Konfigurationsdatei definiert sind oder die Werte für alias und clientId bereits verwendet wurden, um eine Verbindung herzustellen.</p> </div>
keystorePassword	<p>Passwort zum für KeystorePath angegebenen Keystore.</p> <p>Mögliche Werte: Beliebige Zeichenkette</p> <div data-bbox="603 1854 1433 2051"> <p> Dieser Parameter wird nicht benötigt, wenn alias und clientId in der Konfigurationsdatei definiert sind oder die Werte für alias und clientId bereits verwendet wurden, um eine Verbindung herzustellen.</p> </div>

Topic	<p>Name des Topics, in dem der Client veröffentlicht</p> <p>Mögliche Werte: Beliebige Zeichenkette</p> <div> <p>i Das Topic kann innerhalb der Konfigurationsdatei <code>mqttconfig.xml</code> für jeden Broker im Attribut <code>filter</code> des Elements <code>Element Topic</code> definiert werden.</p> </div>
server	<p>Protokoll, Host und Port für die Verbindung vom Client zum Broker.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>tcp://hostname:port</code>: Falls keine SSL/TLS-Verbindung verwendet wird. • <code>ssl://hostname:port</code>: Falls eine SSL/TLS-Verbindung verwendet wird. <div> <p>i Dieser Parameter wird nicht benötigt, wenn <code>alias</code> und <code>clientId</code> in der Konfigurationsdatei definiert sind oder die Werte für <code>alias</code> und <code>clientId</code> bereits verwendet wurden, um eine Verbindung herzustellen.</p> </div>
InputFormat	<p>Input-Format des Adapters</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • BYTES • BASE64_STRING • BASE64_XML
loginUsername	<p>Username, der verwendet wird, um eine Verbindung zu einem Broker herzustellen, der Zugangsdaten verlangt.</p> <p>Mögliche Werte: Benutzername</p> <div> <p>i Dieser Parameter wird nicht benötigt, wenn <code>alias</code> und <code>clientId</code> in der Konfigurationsdatei definiert sind oder die Werte für <code>alias</code> und <code>clientId</code> bereits verwendet wurden, um eine Verbindung herzustellen.</p> </div>

loginPassword	<p>Passwort, das verwendet wird, um eine Verbindung zu einem Broker herzustellen, der Zugangsdaten verlangt.</p> <p>Mögliche Werte: Passwort</p> <div>  Dieser Parameter wird nicht benötigt, wenn <code>alias</code> und <code>clientId</code> in der Konfigurationsdatei definiert sind oder die Werte für <code>alias</code> und <code>clientId</code> bereits verwendet wurden, um eine Verbindung herzustellen. </div>
---------------	---

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt.
-1	Es gab einen Fehler während des Verbindungsaufbaus, der Trennung der Verbindung, der Veröffentlichung oder dem Abonnieren

Input

Für die Operation `Publish` erwartet der Adapter ein beliebiges Input-Dokument.

Ausgabe

Die Ausgabe entspricht dem Input.


Konfigurationsdatei

Um Broker, Client, SSL/TLS-Konfigurationen usw. zu definieren, kann eine Konfigurationsdatei mit einer festgelegten Struktur verwendet werden, die beim Serverstart geladen wird. Die Konfigurationsdatei `mqttconfig.xml` muss unter dem Serverpfad `X4DB/0` abgelegt werden. Wenn Broker und Client in der Konfigurationsdatei definiert sind, dann müssen in den Adapterparametern nur die Werte für `alias` und `clientId` angegeben werden.

Die Datei enthält folgende Informationen:

KeyStore	Pfad und Passwort des Keystores, der verwendet wird, um eine SSL-Verbindung zum Broker aufzubauen. Ist dieses optionale Element nicht angegeben, wird ein Trust-Manager angelegt, der alle Zertifikate annimmt.
-----------------	---

Broker	<p data-bbox="515 275 683 309">MQTT-Server</p> <p data-bbox="515 342 1420 405">Relevante Informationen werden im Attribut <code>alias</code> eingebettet. Dieses Attribut muss eindeutig sein.</p> <p data-bbox="515 409 1005 443">Der Broker enthält folgende Elemente:</p> <ul data-bbox="568 477 1436 1603" style="list-style-type: none"><li data-bbox="568 477 1324 539">• <code>Server</code>: Broker-URL; Die Protokolle TLS und TCP werden unterstützt.<ul data-bbox="651 544 1436 831" style="list-style-type: none"><li data-bbox="651 544 1420 607">• <code>ssl://host:port</code>: Die Verbindung zum Broker findet über das TLS-Protokoll statt.<li data-bbox="651 622 1436 685">• ⓘ Host wird durch den Host Server und Port durch den Port Server ersetzt.<li data-bbox="651 701 1436 831">• <code>tcp://host:port</code>: Die Verbindung zum Broker findet über das TCP-Protokoll (ohne SSL-Verschlüsselung) statt. ⓘ Host wird durch den Host Server und Port durch den Port Server ersetzt.<li data-bbox="568 853 1324 887">• <code>clientId</code>: Eindeutige Client-ID mit folgenden Attributen:<ul data-bbox="651 891 1436 1603" style="list-style-type: none"><li data-bbox="651 891 997 925">• <code>mode</code>: Art der Verbindung<ul data-bbox="730 929 1436 1305" style="list-style-type: none"><li data-bbox="730 929 1340 1016">• <code>ALWAYS_ONLINE_ON_STARTUP</code> (Standard): Der Adapter verbindet sich beim Server-Start und hält die Verbindung die ganze Zeit aufrecht<li data-bbox="730 1021 1372 1108">• <code>ALWAYS_ONLINE_ON_FIRST_PUBLISH</code>: Die Verbindung wird bei der ersten Anfrage (Publish) aufgebaut und die ganze Zeit aufrechterhalten<li data-bbox="730 1113 1436 1305">• <code>ONLINE_ON_DEMAND</code>: Der Adapter verbindet sich, wenn noch keine Verbindung besteht und hat ein zusätzliches Attribut <code>timeout</code>, das lediglich in diesem Modus gültig ist. Das Attribut trennt die Verbindung nach der angegebenen Zeit (in Sekunden) nachdem die letzte Nachricht versendet wurde.<li data-bbox="675 1310 1396 1397">• ⓘ Hat <code>mode</code> einen ungültigen Wert oder ist leer, wird der Standard-Wert <code>ALWAYS_ONLINE_ON_STARTUP</code> angenommen.<li data-bbox="651 1413 1412 1603">• <code>timeout</code>: Timeout in Sekunden. Dieses Attribut ist nur für den Verbindungstyp <code>ONLINE_ON_DEMAND</code> gültig und trennt die Verbindung nach der angegebenen Zeit (in Sekunden) nachdem die letzte Nachricht versendet wurde. ⓘ Wenn <code>timeout</code> leer ist, wird der Standard-Wert von 20 Sekunden angenommen.
--------	--

Topic	<p>Topic das abonniert werden kann, um die veröffentlichte Nachricht zu erhalten. Es hat folgende Attribute:</p> <ul style="list-style-type: none"> • filter: Name des Topic-Filters • qos: Quality of service <ul style="list-style-type: none"> • <i>AT MOST ONCE</i>: Die Nachricht wird höchstens ein Mal oder kein einziges Mal zugestellt. Es erfolgt keine Benachrichtigung, wenn die Zustellung erfolgt ist. • <i>AT LEAST ONCE</i>: Die Nachricht wird immer mindestens ein Mal zugestellt. Erhält der Sender keine Benachrichtigung, wird die Nachricht mit dem DUP-Flag erneut zugestellt, bis eine Benachrichtigung empfangen wurde. • <i>EXACTLY ONCE</i>: Die Nachricht muss beim Sender und beim Empfänger lokal gespeichert werden, bis diese verarbeitet wurde. <p> Ist das Attribut qos nicht angegeben, ist der Standard-Wert AT_LEAST_ONCE.</p>
--------------	--


Beispiel-Konfiguration

```
<?xml version='1.0' encoding='UTF-8'?>
<RootElement>
  <KeyStore password="password">C:/Users/MaxMustermann/Documents/MQTT/Keystore/
application.keystore</KeyStore>
  <Broker alias="Mosquitto_MQTT">
    <Server>ssl://localhost:8883</Server>
    <ClientId mode="ALWAYS_ONLINE_ON_STARTUP">C1</ClientId>
    <ClientId mode="ALWAYS_ONLINE_ON_FIRST_PUBLISH">C2</ClientId>
    <ClientId timeout="40" mode="ONLINE_ON_DEMAND">C3</ClientId>
    <ClientId timeout="20" mode="ONLINE_ON_DEMAND">C4</ClientId>
    <Topic filter="topic/action1" qos="AT_LEAST_ONCE">
      <Process url="x4db://1/X4Experience/ESB/3-Reference/1-AdapterSuites/
X4MQTTAdapter/Processes/TopicProcesses/ProcessTopic1.wrf"/>
    </Topic>
    <Topic filter="topic/action2" qos="AT_MOST_ONCE">
      <Process url="x4db://1/X4Experience/ESB/3-Reference/1-AdapterSuites/
X4MQTTAdapter/Processes/TopicProcesses/ProcessTopic2.wrf"/>
    </Topic>
    <Topic filter="topic/action1" qos="EXACTLY_ONCE">
      <Process url="x4db://1/X4Experience/ESB/3-Reference/1-AdapterSuites/
X4MQTTAdapter/Processes/TopicProcesses/ProcessTopic3.wrf"/>
    </Topic>
  </Broker>
</RootElement>
```

6.13.3.2 MQTT Subscriber

MQTT Subscriber: Ermöglicht das Abonnieren eines Topics in einem MQTT-Broker.

Das MQTT-Protokoll bietet eine einfache Methode zur Nachrichtenübermittlung über ein Publish/Subscribe-Modell. Damit eignet es sich besonders für die Nachrichtenübermittlung im Umfeld des Internet der Dinge, wie z. B. bei Sensoren mit niedriger Leistung oder mobilen Geräten wie Telefonen, eingebetteten Computern oder Mikrocontrollern.





 Der *MQTT Subscriber* wird meist in Verbindung mit dem *MQTT Publisher* verwendet.




Eigenschaften (Properties)



Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> Subscribe: Mit einem Client ein Topic in einem MQTT-Broker abonnieren
-----------	--

Parameter

Adapter	<p>Adapter-Hauptklasse (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.mqtt.MqttSubscriberAdapter</i>: Hauptklasse (Standard)</p>
Alias	<p>Broker-Alias</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> Beliebige Zeichenkette <p> Der Alias kann innerhalb der Konfigurationsdatei <i>mqttconfig.xml</i> als Attribut des Elements <i>Broker</i> definiert werden.</p>
ClientId	<p>Bezeichner für den Client, der den Broker abonniert</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> Falls die Client ID in der Konfigurationsdatei <i>mqttconfig.xml</i> definiert ist: beliebige Zeichenkette Falls die Client ID nicht in der Konfigurationsdatei <i>mqttconfig.xml</i> definiert ist: UUID <p> Die Client ID kann innerhalb der Konfigurationsdatei <i>mqttconfig.xml</i> für jeden Broker im Element <i>ClientId</i> definiert werden.</p>
clientTimeout	<p>Timeout für <i>ONLINE_ON_DEMAND</i>-Verbindungen (für die Operation <i>Subscribe</i>). Wenn <i>ConnectionMode</i> von <i>ONLINE_ON_DEMAND</i> abweicht, wird <i>ClientTimeout</i> nicht berücksichtigt.</p> <p>Mögliche Werte: Ganzzahl größer als 0</p>

connectionMode	<p>Verbindungsmodus, der zum Erstellen des Clients ausgewählt wurde. Abhängig vom gewählten Wert bleibt die Client-Verbindung erhalten oder wird nach einer bestimmten Zeit wieder getrennt (ClientTimeout).</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• ONLINE_ON_DEMAND: Der Parameter ClientTimeout wird benötigt, um den Client zu konfigurieren.• ALWAYS_ONLINE_ON_FIRST_PUBLISH: Der Client bleibt verbunden, sobald eine erste Veröffentlichung erfolgt. Daher wird der Parameter ClientTimeout nicht benötigt. <div><p> Dieser Parameter wird nicht benötigt, wenn alias und clientId in der Konfigurationsdatei definiert sind oder die Werte für alias und clientId bereits verwendet wurden, um eine Verbindung herzustellen.</p></div>
stateful	<p>Legt fest, ob der Zustand zwischen dem Client und dem Broker zustandsorientiert ist.</p> <p>Mögliche Werte: true / false</p> <div><p> Dieser Parameter wird nicht benötigt, wenn alias und clientId in der Konfigurationsdatei definiert sind oder die Werte für alias und clientId bereits verwendet wurden, um eine Verbindung herzustellen.</p></div>
keystorePath	<p>Pfad zum Keystore. Wenn keine SSL/TLS-Konfiguration benötigt wird, um eine Verbindung zum Broker herzustellen, dann kann der Parameterwert leer sein.</p> <p>Mögliche Werte: Dateipfad</p> <div><p> Dieser Parameter wird nicht benötigt, wenn alias und clientId in der Konfigurationsdatei definiert sind oder die Werte für alias und clientId bereits verwendet wurden, um eine Verbindung herzustellen.</p></div>
keystorePassword	<p>Passwort zum für KeystorePath angegebenen Keystore.</p> <p>Mögliche Werte: Beliebige Zeichenkette</p> <div><p> Dieser Parameter wird nicht benötigt, wenn alias und clientId in der Konfigurationsdatei definiert sind oder die Werte für alias und clientId bereits verwendet wurden, um eine Verbindung herzustellen.</p></div>

Topic	<p>Name des Topics, das der Client abonniert</p> <p>Mögliche Werte: Beliebige Zeichenkette</p> <div> <p> Das Topic kann innerhalb der Konfigurationsdatei <code>mqttconfig.xml</code> für jeden Broker im Attribut <code>filter</code> des Elements <code>Topic</code> definiert werden.</p> </div>
server	<p>Protokoll, Host und Port für die Verbindung vom Client zum Broker.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>tcp://hostname:port</code>: Falls keine SSL/TLS-Verbindung verwendet wird. • <code>ssl://hostname:port</code>: Falls eine SSL/TLS-Verbindung verwendet wird. <div> <p> Dieser Parameter wird nicht benötigt, wenn <code>alias</code> und <code>clientId</code> in der Konfigurationsdatei definiert sind oder die Werte für <code>alias</code> und <code>clientId</code> bereits verwendet wurden, um eine Verbindung herzustellen.</p> </div>
InputFormat	<p>Input-Format des Adapters</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>BYTES</code> • <code>BASE64_STRING</code> • <code>BASE64_XML</code>
loginUsername	<p>Username, der verwendet wird, um eine Verbindung zu einem Broker herzustellen, der Zugangsdaten verlangt.</p> <p>Mögliche Werte: Benutzername</p> <div> <p> Dieser Parameter wird nicht benötigt, wenn <code>alias</code> und <code>clientId</code> in der Konfigurationsdatei definiert sind oder die Werte für <code>alias</code> und <code>clientId</code> bereits verwendet wurden, um eine Verbindung herzustellen.</p> </div>

loginPassword	<p>Passwort, das verwendet wird, um eine Verbindung zu einem Broker herzustellen, der Zugangsdaten verlangt.</p> <p>Mögliche Werte: Passwort</p> <div>  Dieser Parameter wird nicht benötigt, wenn <code>alias</code> und <code>clientId</code> in der Konfigurationsdatei definiert sind oder die Werte für <code>alias</code> und <code>clientId</code> bereits verwendet wurden, um eine Verbindung herzustellen. </div>
QoS	<p>Quality of Service, für den Nachrichtenempfang beim Abonnement eines Topics.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>AT_MOST_ONCE</code> • <code>AT_LEAST_ONCE</code> (Standard) • <code>EXACTLY_ONCE</code> <div>  Dieser Parameter wird nicht benötigt, wenn <code>alias</code> und <code>clientId</code> in der Konfigurationsdatei definiert sind oder die Werte für <code>alias</code> und <code>clientId</code> bereits verwendet wurden, um eine Verbindung herzustellen. </div>
Process	<p>Pfad zum Prozess, der jedes Mal ausgeführt wird, wenn eine Nachricht des abonnierten Topics empfangen wurde</p> <p>Mögliche Werte: Prozess-Pfad im Format <code>x4db://User/process.wrf</code></p>

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt.
-1	Es gab einen Fehler während des Verbindungsaufbaus, der Trennung der Verbindung, der Veröffentlichung oder dem Abonnieren

Input

Der Adapter erwartet kein Input-Dokument.

Ausgabe


Die Ausgabe entspricht dem Input.

Konfigurationsdatei

Um Broker, Client, SSL/TLS-Konfigurationen usw. zu definieren, kann eine Konfigurationsdatei mit einer festgelegten Struktur verwendet werden, die beim Serverstart geladen wird. Die Konfigurationsdatei `mqttconfig.xml` muss unter dem Serverpfad `X4DB/0` abgelegt werden. Wenn Broker und Client in der Konfigurationsdatei definiert sind, dann müssen in den Adapterparametern nur die Werte für `alias` und `clientId` angegeben werden.

Die Datei enthält folgende Informationen:

KeyStore	<p>Pfad und Passwort des Keystores, der verwendet wird, um eine SSL-Verbindung zum Broker aufzubauen. Ist dieses optionale Element nicht angegeben, wird ein Trust-Manager angelegt, der alle Zertifikate annimmt.</p>
Broker	<p>MQTT-Server</p> <p>Relevante Informationen werden im Attribut <code>alias</code> eingebettet. Dieses Attribut muss eindeutig sein.</p> <p>Der Broker enthält folgende Elemente:</p> <ul style="list-style-type: none"> • Server: Broker-URL; Die Protokolle TLS und TCP werden unterstützt. <ul style="list-style-type: none"> • <code>ssl://host:port</code> : Die Verbindung zum Broker findet über das TLS-Protokoll statt. • ⓘ Host wird durch den Host Server und Port durch den Port Server ersetzt. • <code>tcp://host:port</code> : Die Verbindung zum Broker findet über das TCP-Protokoll (ohne SSL-Verschlüsselung) statt. <ul style="list-style-type: none"> • ⓘ Host wird durch den Host Server und Port durch den Port Server ersetzt. • ClientId: Eindeutige Client-ID mit folgenden Attributen: <ul style="list-style-type: none"> • mode: Art der Verbindung <ul style="list-style-type: none"> • <code>ALWAYS_ONLINE_ON_STARTUP</code> (Standard): Der Adapter verbindet sich beim Server-Start und hält die Verbindung die ganze Zeit aufrecht • <code>ALWAYS_ONLINE_ON_FIRST_PUBLISH</code>: Die Verbindung wird bei der ersten Anfrage (Publish) aufgebaut und die ganze Zeit aufrechterhalten • <code>ONLINE_ON_DEMAND</code>: Der Adapter verbindet sich, wenn noch keine Verbindung besteht und hat ein zusätzliches Attribut <code>timeout</code>, das lediglich in diesem Modus gültig ist. Das Attribut trennt die Verbindung nach der angegebenen Zeit (in Sekunden) nachdem die letzte Nachricht versendet wurde. <ul style="list-style-type: none"> • ⓘ Hat <code>mode</code> einen ungültigen Wert oder ist leer, wird der Standard-Wert <code>ALWAYS_ONLINE_ON_STARTUP</code> angenommen. • timeout: Timeout in Sekunden. Dieses Attribut ist nur für den Verbindungstyp <code>ONLINE_ON_DEMAND</code> gültig und trennt die Verbindung nach der angegebenen Zeit (in Sekunden) nachdem die letzte Nachricht versendet wurde. <ul style="list-style-type: none"> • ⓘ Wenn <code>timeout</code> leer ist, wird der Standard-Wert von 20 Sekunden angenommen.

Topic	<p>Topic das abonniert werden kann, um die veröffentlichte Nachricht zu erhalten. Es hat folgende Attribute:</p> <ul style="list-style-type: none"> • filter: Name des Topic-Filters • qos: Quality of service <ul style="list-style-type: none"> • <i>AT MOST ONCE</i>: Die Nachricht wird höchstens ein Mal oder kein einziges Mal zugestellt. Es erfolgt keine Benachrichtigung, wenn die Zustellung erfolgt ist. • <i>AT LEAST ONCE</i>: Die Nachricht wird immer mindestens ein Mal zugestellt. Erhält der Sender keine Benachrichtigung, wird die Nachricht mit dem DUP-Flag erneut zugestellt, bis eine Benachrichtigung empfangen wurde. • <i>EXACTLY ONCE</i>: Die Nachricht muss beim Sender und beim Empfänger lokal gespeichert werden, bis diese verarbeitet wurde. <p> Ist das Attribut qos nicht angegeben, ist der Standard-Wert <code>AT_LEAST_ONCE</code>. Für jedes Topic wird ein Prozess definiert, der nach Eingang der Nachricht ausgeführt werden soll. Das Element Process hat folgende Attribute:</p> <ul style="list-style-type: none"> • url: Prozess-Pfad im Format <code>x4db://User/process.wrf</code>
-------	--

Beispiel-Konfiguration

```
<?xml version='1.0' encoding='UTF-8'?>
<RootElement>
  <KeyStore password="password">C:/Users/MaxMustermann/Documents/MQTT/Keystore/
application.keystore</KeyStore>
  <Broker alias="Mosquitto_MQTT">
    <Server>ssl://localhost:8883</Server>
    <ClientId mode="ALWAYS_ONLINE_ON_STARTUP">C1</ClientId>
    <ClientId mode="ALWAYS_ONLINE_ON_FIRST_PUBLISH">C2</ClientId>
    <ClientId timeout="40" mode="ONLINE_ON_DEMAND">C3</ClientId>
    <ClientId timeout="20" mode="ONLINE_ON_DEMAND">C4</ClientId>
    <Topic filter="topic/action1" qos="AT_LEAST_ONCE">
      <Process url="x4db://1/X4Experience/ESB/3-Reference/1-AdapterSuites/
X4MQTTAdapter/Processes/TopicProcesses/ProcessTopic1.wrf"/>
    </Topic>
    <Topic filter="topic/action2" qos="AT_MOST_ONCE">
      <Process url="x4db://1/X4Experience/ESB/3-Reference/1-AdapterSuites/
X4MQTTAdapter/Processes/TopicProcesses/ProcessTopic2.wrf"/>
    </Topic>
    <Topic filter="topic/action1" qos="EXACTLY_ONCE">
      <Process url="x4db://1/X4Experience/ESB/3-Reference/1-AdapterSuites/
X4MQTTAdapter/Processes/TopicProcesses/ProcessTopic3.wrf"/>
    </Topic>
  </Broker>
</RootElement>
```

6.13.4 OPC Client Connector



OPC Client Connector : Stellt die Verbindung zu einem OPC-Server her und kann Werte auslesen und setzen.

Eigenschaften (Properties)

Operation	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: execute: Adapter ausführen
-----------	--

Parameter

Adapter	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: de.softproject.integration.adapter.opc.OpcAdapter: Hauptklasse (Standard)
Host	Hostname oder IP-Adresse des OPC-Servers
Domain	Name der Windows-Domain des OPC-Servers
User	Benutzername, um auf den OPC-Server zuzugreifen
Password	Entsprechendes Passwort des Benutzers
Clsid	Class-ID der OPC-Server-Software
ProgID	Programm-ID

Statuswerte

1	Die Operation wurde ohne technische Fehler ausgeführt
-1	Die Operation wurde mit einem technischem Fehler beendet Die Operation ist eventuell teilweise ausgeführt worden; Genauere Informationen zur Fehlerursache entnehmen Sie dem Server-Log.

Konfiguration

Im Verzeichnis X4DB/0 muss eine Datei opc.xml angelegt werden, die über folgende Struktur verfügt:

```

<opc_config>
  <Connection>
    <Host></Host>
    <Domain></Domain>
    <User></User>
    <Password></Password>
    <ClsId></ClsId>
  </Connection>
  <Group name="">
    <Item name=""/>
    <Item name=""/>
  </Group>
</opc_config>

```

Für jedes Element <Group> und <Item> gilt:

- Es können beliebig viele Gruppen mit beliebig vielen Items angelegt werden.
- Eine Gruppe ist eine vom Benutzer selbst zusammengestellte Sammlung von Items, die zusammen behandelt werden sollen.
- Der Gruppenname kann selbst gewählt werden.
- Der Item-Name muss dessen Benennung auf dem OPC-Sever entsprechen.
- Lese-Operationen werden immer auf die komplette Gruppe ausgeführt.
- Ein Item darf sich in mehreren Gruppen gleichzeitig befinden.
- Beim Schreiben kann man nur auf diejenigen Items schreiben, die sich in derselben Gruppe befinden.

Input

Der Adapter erwartet spezifische Input-XML-Strukturen. Diese Input-XML-Strukturen können über XSL-Mappings in Abhängigkeit von den verarbeiteten XML-Daten dynamisch generiert werden.

```

<OPC>
  <read>
    <Group name=""/>
    <Group name=""/>
  </read>
  <write>
    <Group name="">
      <Item name="">...</Item>
      <Item name="">...</Item>
    </Group>
  </write>
  <copy>
    ...
  </copy>
</OPC>

```

Erläuterung

Für jedes Element <read> gilt:

- Auf dem read-Element können beliebige Attribute angegeben werden, die in die Ausgabe kopiert werden.

- Ein `read`-Element kann beliebig viele `Group`-Elemente enthalten.


Für jedes Element `<write>` gilt:

- Auf dem `write`-Element können beliebige Attribute angegeben werden, die in die Ausgabe kopiert werden.
- Ein `write`-Element kann beliebig viele `Group`-Elemente mit beliebig vielen `Item`-Elementen enthalten.

Für jedes Element `<copy>` gilt:

- Auf dem `copy`-Element können beliebige Attribute angegeben werden, die in die Ausgabe kopiert werden.
- Das `copy`-Element kann beliebige XML-Elemente mit beliebigem Inhalt enthalten; Diese Elemente werden in die Ausgabe kopiert.

Ausgabe

 Für alle Kommandos werden die jeweiligen Attribute entsprechend auf die Container-Elemente kopiert.

Kommando: read

```
<OPC>
  <read>
    <Group name=''>
      <Item name=''> ... </Item>
      <Item name=''> ... </Item>
    </Group>
  </read>
</OPC>
```

Kommando: write

Das Kommando wird ausgeführt, es wird jedoch keine Ausgabe generiert:

```
<OPC>
  <write/>
</OPC>
```

Kommando: copy

```
<OPC>
  <copy> <!-- XML-Struktur aus dem Input-Dokument --> </copy>
</OPC>
```

6.13.5 OPC UA Publisher/Subscriber

- OPC UA Publisher Adapter

- [OPC UA Subscriber Adapter](#)

6.13.5.1 OPC UA Publisher Adapter

OPC UA Publisher Adapter: Verbindet sich mit einem OPC UA Server zum Lesen und Schreiben von Daten von/zum einer Maschine in einem X4-Geschäftsprozess.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>GetServerConfiguration</code>: Ruft die Server-Konfiguration auf • <code>ListNodes</code>: Listet die verfügbaren Knoten auf • <code>ListNodesAttributesId</code>: Listet die Attributes für jeden Knoten auf • <code>Read</code>: Liest Werte und den Status aus • <code>ReadHistory</code>: Liest historische Werte und den Status aus • <code>Write</code>: Schreibt die Attribute eines Knotens
-----------	---

Parameter

Adapter	<p>Adapter-Hauptklasse (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.opcua.OpcUaPublisherAdapter</code>: Hauptklasse (Standard)</p>
url	<p>Server-URL</p> <p>Mögliche Werte: Beliebige gültige URL, z. B. <code>opc.tcp://SP-WS-204.softproject.local:53530/OPCUA/SimulationServer</code></p>
user	<p>Benutzername, um eine Verbindung zum Server aufzubauen</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Benutzernamen, z. B. <code>myUser</code></p>
password	<p>Passwort, um eine Verbindung zum Server aufzubauen</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Passwort, z. B. <code>myPassword</code></p>
securityMode	<p>Sicherheits-Modus und Server-Richtlinien</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>NONE</code> • <code>BASIC256_SIGN</code> • <code>BASIC256_SIGN_ENCRYPT</code> • <code>BASIC128RSA15_SIGN</code> • <code>BASIC128RSA15_SIGN_ENCRYPT</code> • <code>BASIC256SHA256_SIGN</code> • <code>BASIC256SHA256_SIGN_ENCRYPT</code>

autoReconnect	Angabe, ob sich der Client automatisch wieder mit dem Server verbinden soll oder nicht.
nodeIndex	Knoten-Index Mögliche Werte: Beliebige ganze Zahl mit dem Knoten-Index, z. B. <i>1234</i>
nodeIdentifier	Knoten-Bezeichner Mögliche Werte: Beliebige Zeichenkette mit dem Knoten-Bezeichner, z. B. <i>myNode</i>
nodeIdentifierType	Typ des Knoten-Bezeichners Mögliche Werte: <ul style="list-style-type: none"> • String • Numeric
historyOption	Optionen zum Abrufen der Historie eines bestimmten Knotens Mögliche Werte: <ul style="list-style-type: none"> • AT_TIMES • AVERAGE • COUNT • DELTA • END • MAXIMUM • MAXIMUM_ACTUAL_TIME • MINIMUM • MINIMUM_ACTUAL_TIME • RANGE • RAW_DATA • START • WORST_QUALITY
from	Beginn-Datum mit dem Format yyyy-MM-dd hh:mm:ss Mögliche Werte: Beliebiges Datum mit dem entsprechenden Format, z. B. <i>2018-01-01 00:00:00</i>
to	End-Datum mit dem Format yyyy-MM-dd hh:mm:ss Mögliche Werte: Beliebiges Datum mit dem entsprechenden Format, z. B. <i>2018-12-31 00:00:00</i>
interval	Intervall-Wert in Sekunden Mögliche Werte: Beliebige ganze Zeit mit dem Intervall in Sekunden, z. B. <i>3000</i>

Statuswerte


1	Die Adapter-Operation wurde erfolgreich ausgeführt
---	--

-1	Bei der Ausführung des Adapters kam es zu einem Fehler
----	--

Input

Für die Operationen Read und Write erwartet der Adapter ein spezifisches XML-Dokument mit folgenden Elementen:

- `nodeIndex` (Integer)
- `nodeIdentifier` (String)
- `nodeIdentifierType` (String , Numeric)
- `attributeId` (Integer)

 Verwenden Sie die Operation `ListNodesAttributtesId`, um die Knoten-Attribut-ID aufzurufen.

Beispiel-Input für die Operation READ

```
<?xml version="1.0" encoding="UTF-8"
<RootElement>
  <NodeID nodeIndex="5" nodeIdentifier="Counter1" nodeIdentifierType="String"
attributeId="13"></NodeID>
  <NodeID nodeIndex="5" nodeIdentifier="Square1" nodeIdentifierType="String"
attributeId="13"></NodeID>
  <NodeID nodeIndex="5" nodeIdentifier="Triangle1" nodeIdentifierType="String"
attributeId="13"></NodeID>
  <NodeID nodeIndex="0" nodeIdentifier="2256" nodeIdentifierType="Numeric"
attributeId="5"></NodeID>
</RootElement>
```

Beispiel-Input für die Operation WRITE

```
<?xml version="1.0" encoding="UTF-8"
<RootElement>
  <NodeID nodeIndex="5" nodeIdentifier="Counter1" nodeIdentifierType="String"
attributeId="13">999</NodeID>
  <NodeID nodeIndex="5" nodeIdentifier="Square1" nodeIdentifierType="String"
attributeId="13">9636</NodeID>
  <NodeID nodeIndex="5" nodeIdentifier="Triangle1" nodeIdentifierType="String"
attributeId="13">TestData</NodeID>
  <NodeID nodeIndex="0" nodeIdentifier="2256" nodeIdentifierType="Numeric"
attributeId="5">Testing</NodeID>
</RootElement>
```

Output

Der Adapter gibt ein spezifisches XML-Dokument mit dem Ergebnis aus:

Beispiel-Output für die Operation READ

```
<?xml version="1.0" encoding="UTF-8"
<Result>
  <NodeID statusCode="GOOD" nodeIndex="5" nodeIdIdentifier="Counter1"
nodeIdentifierType="String">999</NodeID>
  <NodeID statusCode="GOOD" nodeIndex="5" nodeIdIdentifier="Square1"
nodeIdentifierType="String">9636.0</NodeID>
  <NodeID statusCode="GOOD" nodeIndex="5" nodeIdIdentifier="Triangle1"
nodeIdentifierType="String">1.5999999955515</NodeID>
  <NodeID statusCode="GOOD" nodeIndex="0" nodeIdIdentifier="2256" nodeIdIdentifierType="
Numeric">The current status of the server</NodeID>
</Result>
```

Beispiel-Output für die Operation WRITE

```
<?xml version="1.0" encoding="UTF-8"
<RootElement>
  <NodeID nodeIndex="5" nodeIdIdentifier="Counter1" nodeIdIdentifierType="String">SUCCE
SS</NodeID>
  <NodeID nodeIndex="5" nodeIdIdentifier="Square1" nodeIdIdentifierType="String">SUCCE
S</NodeID>
  <NodeID nodeIndex="5" nodeIdIdentifier="Triangle1">XML malformed - Invalid value
format to write into the Node attribute</NodeID>
  <NodeID nodeIndex="0" nodeIdIdentifier="2256">Bad_NotWritable (0x803B0000) "The
access level does not allow writing to the Node</NodeID>
</RootElement>
```

6.13.5.2 OPC UA Subscriber Adapter

OPC UA Subscriber Adapter: Stellt eine Verbindung zu einem OPC UA Server her und abonniert oder storniert Datenänderungen oder Ereignisse.

Eigenschaften (Properties)

Operation	Vom Adapter ausgeführte Operation Mögliche Werte: <ul style="list-style-type: none"> Subscribe: Abonniert Datenänderungen oder Ereignisse entsprechend dem Input-XML-Dokument UnSubscribe: Storniert bestehende Abonnements für einen bestimmten Client
-----------	---

Parameter

Adapter	Adapter-Hauptklasse (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.opcua.OpcUaSubscriberAdapter</i> : Hauptklasse (Standard)
---------	--

clientAlias	Alias zur Kennzeichnung der Verbindungseinstellungen Mögliche Werte: Beliebige Zeichenkette mit dem Client Alias
process	X4DB-Pfad zu einem technischen Prozess, der nach jedem Abonnentenempfang ausgeführt wird Mögliche Werte: Beliebiger gültiger X4DB-Pfad, z. B. <i>x4db://1/SampleProject/SampleProcess.wrf</i>
url	Server-URL Mögliche Werte: Beliebige gültige URL, z. B. <i>opc.tcp://SP-WS-204.softproject.local:53530/OPCUA/SimulationServer</i>
user	Benutzername, um eine Verbindung zum Server aufzubauen Mögliche Werte: Beliebige Zeichenkette mit dem Benutzernamen, z. B. <i>myUser</i>
password	Passwort, um eine Verbindung zum Server aufzubauen Mögliche Werte: Beliebige Zeichenkette mit dem Passwort, z. B. <i>myPassword</i>
security mode	Sicherheits-Modus und Server-Richtlinien Mögliche Werte: <ul style="list-style-type: none"> • NONE • BASIC256_SIGN • BASIC256_SIGN_ENCRYPT • BASIC128RSA15_SIGN • BASIC128RSA15_SIGN_ENCRYPT • BASIC256SHA256_SIGN • BASIC256SHA256_SIGN_ENCRYPT
autoReconnect	Angabe, ob sich der Client automatisch wieder mit dem Server verbinden soll oder nicht.

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt
-1	Bei der Ausführung des Adapters kam es zu einem Fehler

Input

Operation Subscribe:

- Vollständiges Beispiel mit verschiedenen Abonnement-Typen:

Input für die Operation Subscribe

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <Subscriptions>
3      <!-- Subscription to a data value change -->
4      <Subscription type="DATA_CHANGE" nodeIndex="3" nodeIdIdentifier="C
ounter" nodeIdIdentifierType="String"></Subscription>
5
6      <!-- Subscription to an event, specifying a condition -->
7      <Subscription type="EVENT" nodeIndex="5" nodeIdIdentifier="MyLevel
.Alarm" nodeIdIdentifierType="String">
8          <Field name="SourceName">
9              <Literal>LIKE</Literal>
10             <Operand>Server</Operand>
11         </Field>
12     </Subscription>
13
14     <!-- Subscription to an event, specifying more than one
condition -->
15     <Subscription type="EVENT" nodeIndex="5" nodeIdIdentifier="MyLevel
.Alarm" nodeIdIdentifierType="String">
16         <Field name="ActiveState" attributeId="13">
17             <Operand>EQUALS</Operand>
18             <Literal>Active</Literal>
19         </Field>
20         <Field name="AckedState" attributeId="13">
21             <Operand>EQUALS</Operand>
22             <Literal>Unacknowledged</Literal>
23         </Field>
24         <Field name="Comment" attributeId="13"></Field>
25         <Field name="EnabledState" attributeId="13"></Field>
26         <Field name="EventType" attributeId="1">
27             <Operand>EQUALS</Operand>
28             <Literal>i=9482</Literal>
29         </Field>
30         <Field name="Quality">
31             <Operand>EQUALS</Operand>
32             <Literal>GOOD (0x00000000) ""</Literal>
33         </Field>
34         <Field name="Severity">
35             <Operand>LESS_THAN</Operand>
36             <Literal>600</Literal>
37         </Field>
38         <Field name="SourceName"></Field>
39         <Field name="Time"></Field>
40     </Subscription>
41
42     <!-- Configuring -->
43     <OnAfterCreate>x4db://1/TestingOPCUA/Processes/
OnAfterCreate.wrf</OnAfterCreate>
44     <OnAlive>x4db://1/TestingOPCUA/Processes/OnAlive.wrf</OnAlive>
45 </Subscriptions>

```

- Datenwertänderung abonnieren:

- ❗ Um eine Datenwertänderung zu abonnieren, muss ein Subscription-Knoten mit folgenden Pflichtattributen angelegt werden:
- type, um die Art des Abonnements festzulegen
 - nodeIndex, nodeIdentifier, um den OPC UA-Knoten zu identifizieren, von dem die Änderungen abonniert werden
 - nodeIdentifierType, um den Datentyp des Knotenbezeichners zu spezifizieren

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <Subscriptions>
3      <Subscription type="DATA_CHANGE" nodeIndex="3" nodeIdentifier="Counter"
4      nodeIdentifierType="String">
5  </Subscription>
6  </Subscriptions>

```

- Events abonnieren, ohne Felder zu spezifizieren:

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <Subscriptions>
3      <!-- We will receive an empty event -->
4      <Subscription type="EVENT" nodeIndex="5" nodeIdentifier="MyLevel
5      .Alarm" nodeIdentifierType="String">
6  </Subscription>
7  </Subscriptions>

```

- Events abonnieren und Felder spezifizieren:

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <Subscriptions>
3      <!-- In the Field node, we define the field we want to retrieve
4      on each event reception. -->
5      <Subscription type="EVENT" nodeIndex="5" nodeIdentifier="MyLevel
6      .Alarm" nodeIdentifierType="String">
7          <Field name="SourceName"></Field>
8          <Field name="Time"></Field>
9      </Subscription>
10     <!-- Any number of Field nodes possible. -->
11 </Subscriptions>

```

- Events abonnieren und Filter für die spezifizierten Felder definieren:

Es ist möglich, Bedingungen für die Attribute zu definieren, die bei jedem Ereignisempfang erhalten werden. Dazu müssen entsprechend viele Field-Knoten angelegt werden. Jeder Field-Knoten enthält dabei die Elemente Literal und Operand.

Event mit Filtern abonnieren

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <Subscriptions>
3      <!-- In the Field node, we define the field we want to retrieve
         on each event reception. Inside this node Operand and Literal are
         defined in order to create the condition over the field. -->
4      <Subscription type="EVENT" nodeIndex="5" nodeIdIdentifier="MyLevel
        .Alarm" nodeIdIdentifierType="String">
5          <!-- Severity less than 600 -->
6          <Field name="Severity">
7              <Operand>LESS_THAN</Operand>
8              <Literal>600</Literal>
9          </Field>
10         </Subscription>
11         <!-- Any number of Field nodes possible. If there is more than
             one Field node containing conditions, the logical operator applied
             between fields will be 'OR' -->
12     </Subscriptions>

```

ⓘ Beachten Sie:

- Das *name*-Attribut innerhalb von *Field* ist erforderlich.
- Operand kann folgende Werte haben:
 - LESS_THAN
 - LESS_THAN_OR_EQUALS
 - GREATER_THAN
 - GREATER_THAN_OR_EQUALS
 - EQUALS
 - NOT
 - LIKE
- *Literal* enthält die verschiedenen literalen Werte, die innerhalb des Ausdrucks berücksichtigt werden sollen.

Operation Unsubscribe

Die Operation *UnSubscribe* benötigt keinen Input, da eine Stornierung des Abonnements erfolgt. Daher muss im Parameter *clientAlias* lediglich der Alias des Mandanten angegeben werden.

Output**Operation Subscribe**

Die Ausgabestruktur hängt vom Typ der Empfangsnachricht ab.

- Wird ein Event mit Feldern empfangen (mit oder ohne Bedingungen), die im Abonnement konfiguriert wurden, ähnelt die Antwort-Struktur folgender Beispielausgabe:

Output für Event-Antworten

```

1  <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2  <Message type="EVENT">
3      <Node description="ns=5;s=MyLevel.Alarm"/>
4      <Fields>
5          <Field fieldName="ActiveState">Active</Field>
6          <Field fieldName="AckedState">Unacknowledged</Field>
7          <Field fieldName="Comment">EMPTY</Field>
8          <Field fieldName="EnabledState">Enabled</Field>
9          <Field fieldName="EventType">i=9482</Field>
10         <Field fieldName="Quality">GOOD</Field>
11         <Field fieldName="Severity">500</Field>
12         <Field fieldName="SourceName">MyLevel</Field>
13         <Field fieldName="Time">1576741128000</Field>
14     </Fields>
15 </Message>

```

Die Angabe von Bedingungen führt dazu, dass mehr oder weniger Events empfangen werden. Sobald jedoch ein Event empfangen wird, enthält die Antwort die Felder, die im Abonnement angegeben wurden.

- Wird ein Event ohne Feld-Spezifikationen empfangen, ähnelt die Antwort-Struktur folgender Beispielausgabe:

Output für Event-Antworten

```

1  <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2  <Message type="EVENT"/>

```

- Wird eine Datenwertänderung empfangen, ähnelt die Antwort-Struktur folgender Beispielausgabe:

Output für Datenwertänderungen

```

1  <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2  <Message type="DATA_CHANGE">
3      <Node description="ns=3;s=Counter"/>
4      <OldValue>
5          <!-- 0 means null timestamp -->
6          <sourceTimestamp>0</sourceTimestamp>
7          <!-- No value and statusCode present in 'OldValue' node
happens only just after subscribing on the first reception -->
8      </OldValue>
9      <NewValue>
10         <sourceTimestamp>13221214634996</sourceTimestamp>
11         <statusCode>GOOD</statusCode>
12         <value>2312.11</value>
13     </NewValue>
14 </Message>

```


Operation UnSubscribe

Die Operation UnSubscribe hat keine Ausgabe, da der Status 1 bedeutet, dass die Operation erfolgreich ausgeführt wurde.

Alive Listener Konfiguration

Es gibt einen Alive Listener, der es ermöglicht, Aktionen für jede Live-Benachrichtigung eines Abonnements zu konfigurieren.

Dabei lassen sich Prozesse für folgende Situationen definieren:

- **After create:** Das Abonnement wurde auf dem Server (neu) erstellt. Dies ist der Fall, wenn die Abonnieung aufgrund einer Kommunikationsunterbrechung in einen Timeout geriet und nach der Wiederverbindung neu erstellt werden muss.
- **Alive:** Der Server bestätigte, dass die Verbindung aktiv ist, obwohl es keine Änderungen zu senden gab.
- **Life timeout:** Der Server hat nicht bestätigt, dass die Verbindung aktiv ist, und der Life-Timeout wurde überschritten.
- **Timeout:** Der Server hat nicht bestätigt, dass die Verbindung aktiv ist, und die maximale Keepalive-Anzahl wurde überschritten.

Die Aktionen werden als Teil des Input-XML-Dokuments wie folgt definiert:

Input for Subscribe operation

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <Subscriptions>
3    <Subscription ...></Subscription>
4    <Subscription ...></Subscription>
5
6    <!-- Alive listener actions -->
7    <!-- We can define some of them or none, they are not required -->
8
9    <OnAfterCreate>x4db://1/TestingOPCUA/Processes/OnAfterCreate.wrf</
OnAfterCreate>
10   <OnAlive>x4db://1/TestingOPCUA/Processes/OnAlive.wrf</OnAlive>
11   <onLifetimeTimeout>x4db://1/TestingOPCUA/Processes/
onLifetimeTimeout.wrf</onLifetimeTimeout>
12   <onTimeout>x4db://1/TestingOPCUA/Processes/onTimeout.wrf</onTimeout>
13 </Subscriptions>

```

6.13.6 SIMATIC S7 Connector



SIMATIC S7 Connector : Ermöglicht das Anbinden von *Siemens SIMATIC S7*-Systemen als SPS aus technischen Prozessen heraus. Dabei lassen sich sowohl Daten aus der Steuerung auslesen als auch an diese übermitteln. Über eine XML-Input-Struktur kann bestimmt werden, welche Daten gelesen bzw. geschrieben werden sollen. Es können *SIMATIC*-Systeme der Reihe S7-400-1500 angebunden werden.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • Read: Daten aus der SPS auslesen • Write: Daten in die SPS schreiben
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.s7.S7Connector:</i> Hauptklasse (Standard)
<i>host</i>	Adresse, über welche die SPS im Netzwerk erreichbar ist Mögliche Werte: IP-Adresse oder Host-Name
<i>rack</i>	Rack, das auf der SPS angesprochen werden soll Mögliche Werte: Nummer des Racks (beim SPS-Entwickler zu erfragen)
<i>slot</i>	Slot, der auf der SPS angesprochen werden soll Mögliche Werte: Nummer des Slots (beim SPS-Entwickler zu erfragen)

Statuswerte

<i>1(ok)</i>	Erfolgreiche Kommunikation (Lesen oder Schreiben) mit der SPS
<i>-1(error)</i>	Fehler bei der Kommunikation (Lesen oder Schreiben) mit der SPS Eine Fehlermeldung wird zurückgegeben

Input

Dieser Adapter erwartet jeweils ein XML-Dokument mit folgender Input-Struktur:

Input: Operation Read

```
<Read dbnum="[DBNUM]" bytes="[LENGTH]" offset="[OFFSET]">
  <FELDNAME s7type="[TYP]" s7size="[LENGTH]" />
  ...
</Read>
```

Parameter:

- dbnum: DBNummer von der gelesen werden soll (beim SPS-Entwickler zu erfragen)

- `bytes`: Anzahl der zu lesenden Bytes
- `offset`: Offset in Bytes, ab dem gelesen werden soll
- Die Elemente `FELDDNAME` sind frei wählbar
- `s7type`: Typ des zu lesenden Feldes;
Mögliche Werte:
 - Array
 - Bit (**Hinweis:** liest nur 1 Bit, daher müssen immer 8 Felder angegeben werden!)
 - Byte
 - `DateAndTime`
 - Date (im Format `yyyy-MM-dd'T'HH:mm:ss`)
 - Integer
 - Long
 - Real
 - String
 - Time (als Long)
- `s7size`: Länge in Bytes, die für das Feld gelesen werden sollen. Nur für die Typen Array und String notwendig

Input: Operation Write

```
<Write dbnum="DBNUM" bytes="LENGTH" offset="OFFSET">
  <FELDDNAME s7type="String" s7size="12">[WERT]</AUFNR>
  ...
</Write>
```

Parameter:

- siehe Operation Read
- Der WERT des Elements muss syntaktisch zum Typ passen

Ausgabe

Dieser Adapter gibt ein XML-Dokument mit folgendem Aufbau zurück:

- Operation Read: Die Ausgabe entspricht der Eingabe, die Elemente sind bei Erfolg jeweils mit den passenden Werten gefüllt
- Operation Write: Die Ausgabe entspricht der Eingabe

6.14 Message Oriented Middleware

6.14.1 Apache Kafka

Apache Kafka ermöglicht es, Datenströme über eine verteilte Streaming-Plattform zu speichern und zu verarbeiten. Mit verschiedenen Schnittstellen können Daten in Kafka-Cluster geschrieben, Daten gelesen oder aus Drittsystemen im- oder exportiert werden. Durch die hohe Fehlertoleranz und Skalierbarkeit eignet sich Apache Kafka für große Datenmengen und Anwendungen z. B. im Big-Data-Umfeld.

① Weitere Informationen zur Verwendung von Apache Kafka in einer lokalen Umgebung finden sich [hier](#).
Weitere Informationen zur Konfiguration und Integration von TLS-Zertifikaten in Apache Kafka Cluster finden sich [hier](#).

Um die beiden Adapter [Apache Kafka Publisher](#) bzw. [Apache Kafka Subscriber](#) zu verwenden, wird eine Konfigurationsdatei in der X4 Server-Installation benötigt. Diese Konfigurationsdatei muss `apachekafka-config.xml` heißen und im Ordner `X4DB/0` liegen. Beim Start des X4 Servers werden alle Informationen aus dieser Datei geladen.

Die folgende XML-Datei zeigt alle erforderlichen Elemente und Attribute in der richtigen Struktur:

Beispieldatei apachekafka-config.xml

```
<?xml version='1.0' encoding='UTF-8'?>
<ApacheKafkaConfiguration>
  <!-- Many clusters can be defined here -->
  <Cluster id="ApacheKafka_broker_1" server="localhost:9092">

    <!-- Now the different clients that can work with this broker are defined:
      For each ClientId, we define the following things:
      - The ClientId name: alias to identify the client in X4 Processes.
      - groupId: the id of the group where the consumer/publisher belongs
to.
    -->
    <Client id = "AK_Client_1" groupId="test" autoCommitIntervalMS="100"/>
    <Client id = "AK_Client_2" groupId="test" autoCommitIntervalMS="100"/>

    <!-- Now, the different topics that clients can interact with for this broker
are defined -->
    <Topic filter="topic1"/>
    <Topic filter="topic2"/>
    <Topic filter="topic3"/>
  </Cluster>


  <Cluster id="ApacheKafka_broker_2" server="localhost:9093">

    <!-- Node that includes the SSL/TLS needed configuration for server -->
    <SslConfiguration privateKeyPassword="password">
      <!-- Server keystore password and server keystore path -->
      <KeyStore password="password" path="C:\tmp\ssl_new\server.keystore.jks"/>
      <!-- Server truststore password and server truststore path -->
      <TrustStore password="password" path="C:
\tmp\ssl_new\server.truststore.jks"/>
    </SslConfiguration>

    <!-- Now the different clients that can work with this broker are defined:
      For each ClientId, we define the following things:
      - The ClientId name: alias to identify the client in X4 Processes.
      - groupId: the id of the group where the consumer/publisher belongs
to.
    -->
    <Client groupId="test" autoCommitIntervalMS="100" id = "AK_Client_3"/>
    <Client groupId="test" autoCommitIntervalMS="100" id = "AK_Client_4"/>

    <!-- Now, the different topics that clients can interact with for this broker
are defined -->
    <Topic filter="topic2"/>
    <Topic filter="topic5"/>
    <Topic filter="topic4"/>
  </Cluster>
</ApacheKafkaConfiguration>
```

Im Folgenden sind die wichtigsten Elemente und Attribute näher erläutert:

Element / Attribut	Beschreibung
Cluster	<i>Muss eindeutig sein.</i> Definiert die erforderlichen Parameter, um die Verbindung zu einem bestimmten Kafka-Cluster zu konfigurieren. Im Attribut <code>id</code> befindet sich die Information, die für den Adapter-Parameter <code>clusterId</code> benötigt wird.
server	URL des Clusters. Unterstützt das TCP-Protokoll. Mögliche Werte. <code>hostname:port</code>
SslConfiguration	Definiert die SSL/TLS-Konfigurationsparameter.
privateKeyPassword	Passwort des Private Key in der Keystore-Datei.
KeyStore	Definiert, wo sich der zu verwendende Keystore befindet und welches Passwort hinterlegt ist.
TrustStore	Definiert, wo sich der zu verwendende Truststore befindet und welches Passwort hinterlegt ist.
ClientId	Informationen zum Client, der verwendet wird, um Nachrichten zu einem bestimmten Topic zu veröffentlichen.
groupId	Definiert die Gruppen-ID, die verwendet wird, um die Verbindungen zu gruppieren.
autoCommitIntervalMS	Definiert die Häufigkeit der Commits in Millisekunden. <div> Dieses Attribut wird für den Adapter Apache Kafka Subscriber benötigt.</div>
Topic	Topic, in dem die Nachrichten veröffentlicht werden.
filter	Name des Topics

6.14.1.1 Apache Kafka Publisher


Apache Kafka Publisher: Ermöglicht es, Nachrichten zu einem bestimmten Apache Kafka Topic zu veröffentlichen.

Eigenschaften (Properties)

Operation	Definiert die vom Adapter ausgeführte Operation. Mögliche Werte: <ul style="list-style-type: none"> <code>Publish</code>: Veröffentlicht eine Nachricht zu einem bestimmten Topic, die in einem XML-Input angegeben ist.
-----------	---

Parameter

Adapter	<p>Adapter-Hauptklasse (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.apachekafka.ApacheKafkaPublisherAdapter</i>: Hauptklasse (Standard)</p>
clusterId	<p>ID des Clusters, in dem die Nachricht veröffentlicht wird</p> <p>Mögliche Werte: Cluster-ID</p> <div><p> Die ID des Clusters kann in der Datei <code>apachekafka-config.xml</code> gefunden werden. Sie befindet sich im Attribut <code>id</code> des Elements <code>Cluster</code>.</p></div>
clientId	<p>ID des Clients, mit dem die Nachricht veröffentlicht wird</p> <p>Mögliche Werte: Client-ID</p> <div><p> Die ID des Clients kann in der Datei <code>apachekafka-config.xml</code> gefunden werden. Sie befindet sich im Attribut <code>id</code> des Elements <code>Client</code>.</p></div>
topic	<p>Name des Topics, in dem die Nachricht veröffentlicht wird</p> <p>Mögliche Werte: Zeichenkette aus alphanumerischen ASCII-Zeichen, . (Punkt), _ (Unterstrich) und - (Bindestrich)</p>

messageDataType	<p>Datentyp der Nachricht(en), die veröffentlicht werden.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • BYTES: Base64-kodierter Wert • BYTEARRAY: Base64-kodierter Wert • DOUBLE: Gleitkommazahl • FLOAT: Gleitkommazahl • INTEGER: Ganzzahl • LONG: Ganzzahl • SHORT: Ganzzahl • STRING (Standard): Zeichenkette • UUID: Java UUID <div style="border: 1px solid orange; padding: 10px; margin-top: 10px;"> <p> In einem Topic können Nachrichten mit verschiedenen Datentypen veröffentlicht werden. Um die Nachrichten mit dem Apache Kafka Subscriber Adapter zu lesen, muss der Datentyp bekannt und korrekt eingestellt sein. Beispielsweise kann eine Nachricht vom Datentyp STRING nicht mit der Datentyp-Einstellung LONG gelesen werden. Daher sollten alle Nachrichten in einem Topic vom selben Datentyp sein.</p> </div>
-----------------	--

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt.
-1	Bei der Ausführung des Adapters kam es zu einem Fehler.

Input

Der Adapter erwartet eine vordefinierte XML-Struktur als Input. Die Struktur ist abhängig vom Datentyp der Nachricht.

- Datentyp STRING:

```
<Messages>
  <Message>
    <!-- This is a message. -->
  </Message>
  <Message>
    <!-- This is another message. -->
  </Message>
</Messages>
```

- Datentyp BYTES:


```
<Messages>
  <Message>ZXN0byB1cyB1biBmaWNoZXJvIGRlIHVleHRv</Message>
</Messages>
```

- Datentyp NUMERIC:

```
<Messages>
  <Message>5428</Message>
</Messages>
```



Output

Bei einer erfolgreichen Ausführung wird der Input als Output ausgegeben.



6.14.1.2 Apache Kafka Subscriber


Apache Kafka Subscriber: Ermöglicht es, Nachrichten zu einem bestimmten Topic zu abonnieren, das Abonnement zu beenden oder die letzte Nachricht zu lesen.

Eigenschaften (Properties)

Operation	<p>Definiert die vom Adapter ausgeführte Operation.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Subscribe: Abonniert ein Topic, um Nachrichten zu erhalten. Dabei wird ein vorhandener Client aus der Datei <code>apachekafka-config.xml</code> verwendet. <div style="border: 1px solid #f0e68c; padding: 10px; margin: 10px 0;"> <p> Für ein Abonnement kann nur ein Client verwendet werden, der noch nicht für ein anderes Topic verwendet wird. Ein Client kann nicht mehrere Topics abonnieren.</p> </div> <p>Erforderliche Parameter: <code>clusterId</code>, <code>clientId</code>, <code>topic</code>, <code>process</code></p> <ul style="list-style-type: none"> • Unsubscribe: Trennt Client und Topic und beendet das Abonnement eines bestimmten Topics. <p>Erforderliche Parameter: <code>clusterId</code>, <code>clientId</code></p> <ul style="list-style-type: none"> • Last offset: Gibt den Offset eines bestimmten Clients bei dessen abonnierten Topic zurück. Mit dem Offset können vergangene Nachrichten erneut empfangen werden. <div style="border: 1px solid #f0e68c; padding: 10px; margin: 10px 0;"> <p> Wenn diese Operation für einen Client verwendet wird, der kein Topic abonniert hat, dann wird ein leeres Ergebnis zurückgegeben.</p> </div> <p>Erforderliche Parameter: <code>clusterId</code>, <code>clientId</code>, <code>topic</code></p>
-----------	--

Parameter

Adapter	<p>Adapter-Hauptklasse (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.apachekafka.ApacheKafkaSubscriberAdapter</code>: Hauptklasse (Standard)</p>
clusterId	<p>ID des Clusters, mit dem das Topic abonniert wird</p> <p>Mögliche Werte: Cluster-ID</p> <div> <p> Die ID des Clusters kann in der Datei <code>apachekafka-config.xml</code> gefunden werden. Sie befindet sich im Attribut <code>id</code> des Elements <code>Cluster</code>.</p> </div>
clientId	<p>ID des Clients, mit dem das Topic abonniert wird</p> <p>Mögliche Werte: Client-ID</p> <div> <p> Die ID des Clients kann in der Datei <code>apachekafka-config.xml</code> gefunden werden. Sie befindet sich im Attribut <code>id</code> des Elements <code>Client</code>.</p> </div>
topic	<p>Name des Topics, das abonniert wird</p> <p>Mögliche Werte: Zeichenkette aus alphanumerischen ASCII-Zeichen, <code>.</code> (Punkt), <code>_</code> (Unterstrich) und <code>-</code> (Bindestrich) <u>oder</u> kommasetrennte Liste aus mehreren Zeichenketten, um mehrere Topics zu abonnieren</p>
process	<p>Prozess, der ausgeführt wird, nachdem eine Nachricht empfangen wurde.</p> <p>Mögliche Werte: Zeichenkette aus alphanumerischen ASCII-Zeichen, <code>.</code> (Punkt), <code>_</code> (Unterstrich) und <code>-</code> (Bindestrich)</p>
consumerPosition	<p>Offset, mit dem Nachrichten empfangen werden, wenn ein Topic abonniert wird.</p> <p>Mögliche Werte: Ganzzahl größer als 0</p> <div> <p> Dieser Parameter hat nur einen Effekt, wenn ein einzelnes Topic abonniert wird. Wenn mehrere Topics abonniert werden, hat dieser Parameter keine Auswirkungen.</p> </div>

messageDataType	<p>Datentyp der Nachricht(en), die veröffentlicht werden.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • BYTES: Base64-kodierter Wert • BYTEARRAY: Base64-kodierter Wert • DOUBLE: Gleitkommazahl • FLOAT: Gleitkommazahl • INTEGER: Ganzzahl • LONG: Ganzzahl • SHORT: Ganzzahl • STRING (Standard): Zeichenkette • UUID: Java UUID <div style="border: 1px solid orange; padding: 10px; margin-top: 10px;"> <p> In einem Topic können Nachrichten mit verschiedenen Datentypen veröffentlicht werden. Um die Nachrichten mit dem Apache Kafka Subscriber Adapter zu lesen, muss der Datentyp bekannt und korrekt eingestellt sein. Beispielsweise kann eine Nachricht vom Datentyp STRING nicht mit der Datentyp-Einstellung LONG gelesen werden. Daher sollten alle Nachrichten in einem Topic vom selben Datentyp sein.</p> </div>
-----------------	--

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt.
-1	Bei der Ausführung des Adapters kam es zu einem Fehler.

Input

Dieser Adapter erwartet keinen Input.

Output

Der Adapter gibt abhängig von der ausgeführten Operation folgenden Output aus:

- Operation `Subscribe`: Für diese Operation wird kein direkter Output ausgegeben. Wenn in einem abonnierten Topic eine neue Nachricht empfangen wird, dann wird ein Prozess mit der empfangenen Nachricht als Input gestartet. Diese Nachricht wird in folgender Struktur übergeben:

Beispiel Output Operation `Subscribe`


```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Message>
  <ClusterId>ApacheKafka_broker_1</ClusterId>
  <Topic>topicString</Topic>
  <Offset>20</Offset>
  <Content encoding="base64">This is a message.</Content>
</Message>
```

- Operation `Unsubscribe`: Kein Output

- Operation Last offset:

Beispiel Output Operation Last offset

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Offsets>
  <Offset partitionId="0" value="17"/>
</Offsets>
```

 Im Output dieser Adapteroperation zeigt der Wert von `partitionId` die Partition, in der sich das Topic befindet, und der Wert von `value` den Offset zur letzten Nachricht, die der angegebene Client gelesen hat.

6.14.2 AMQP Publisher/Subscriber

6.14.2.1 AMQP Publisher


AMQP Publisher: Ermöglicht das Veröffentlichen von Nachrichten in einem AMQP-Topic.

Eigenschaften (Properties)

Operation	Vom Adapter ausgeführte Operation Mögliche Werte: • Publish: Nachrichten in einem AMQP-Topic veröffentlichen
-----------	---

Parameter

Adapter	Adapter-Hauptklasse (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.amqp.AmqpPublisherAdapter</i> : Hauptklasse (Standard)
storedConfig	Bezeichner der Konfiguration, bereitgestellt in der Datei <i>amqpconfig.xml</i> Mögliche Werte: Beliebige Zeichenkette mit dem Bezeichner, z. B. <i>queueEmployeesConfig</i>
host	AMQP Server-Host Mögliche Werte: Server-Host, z. B. <i>localhost</i>

port	<p>Dedizierter AMQP-Port auf dem Server</p> <p>Mögliche Werte: AMQP-Port-Nummer, z. B. <i>5672</i></p>
username	<p>Name des Benutzers, der die Nachricht veröffentlicht</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Benutzernamen, z. B. <i>user</i></p>
password	<p>Password des Benutzers</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Passwort, z. B. <i>p4ssw0rd</i></p>
queue	<p>Warteschlange, in der die Nachricht veröffentlicht wird</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Namen der Warteschlange, z. B. <i>employees</i></p>
container	<p>Container, in dem sich die Warteschlange befindet</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Container-Namen z. B. <i>employeesContainer</i></p>
qos	<p>Quality of Service der Nachricht</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • AT_MOST_ONCE: Die Nachricht wird höchstens ein Mal oder kein einziges Mal zugestellt. • AT_LEAST_ONCE: Die Nachricht wird mindestens ein Mal zugestellt. • EXACTLY_ONCE: Die Nachricht wird nur ein Mal zugestellt. <p> Ist der Parameter qos nicht angegeben, ist der Standard-Wert AT_MOST_ONCE.</p>
durable	<p>Nachricht als dauerhaft definieren</p> <p>Mögliche Werte: true (Standard) / false</p>
priority	<p>Prioritätsstufe der Nachricht</p> <p>Mögliche Werte: Beliebige Ganzzahl zwischen 0 und 255, z. B. <i>3</i></p>

ttl	Zeitspanne in Millisekunden, nach der die Nachricht verworfen werden soll Mögliche Werte: Beliebige Ganzzahl, die die Zeitspanne in Millisekunden angibt, z. B. 4500
-----	--

Statuswerte

0	Die Operation wurde erfolgreich ausgeführt (es wird keine Ausgabe generiert)
-1	Während der Ausführung der Operation kam es zu einem Fehler (es wird keine Ausgabe generiert)

Konfiguration

Um die benötigten Informationen bereitzustellen, kann eine Konfigurationsdatei verwendet werden, die beim Serverstart geladen wird. Die Konfigurationsdatei `amqpconfig.xml` muss unter dem Pfad `X4DB/0` bereitgestellt werden.

Die Datei hat folgende Struktur:

amqpconfig.xml

```
<?xml version='1.0' encoding='UTF-8'?>
<RootElement>
  <AMQPConfig name="queueEmployeesConfig">
    <Server host="localhost" port="5672"/>
    <Client user="user" password="p4ssw0rd"/>
    <Queue container="employeesContainer" name="employees" qos="AT_LEAST_ONCE"/>
  </AMQPConfig>
</RootElement>
```

Input

Der AMQP Publisher erwartet folgende Input-Struktur, wobei nur ein Element pro Nachricht erlaubt ist:

Einfache Typen:

Beispiel-Input: Einfache Typen

```
<Messages>
  <Message>
    <Element type="String" value="Good morning!!"/>
  </Message>
  <Message>
    <Element type="Int" value="18"/>
  </Message>
</Messages>
```

- ❗ Folgende einfache Typen sind erlaubt: Binary, Boolean, Byte, Char, Decimal32, Decimal64, Decimal128, Double, Float, Long, Short, String, Symbol, Timestamp (nach dem Muster `yyyy-[m]m-[d]d hh:mm:ss`), UByte, UInt, ULong, UShort, Uuid.

Komplexe Typen:

- Liste:

Beispiel-Input: Liste

```
<Messages>
  <Message>
    <Element type="List">
      <Element type="Int" value="1"/>
      <Element type="Short" value="3"/>
      <Element type="Boolean" value="false"/>
    </Element>
  </Message>
</Messages>
```

- Map:



Beispiel-Input: Map

```
<Messages>
  <Message>
    <Element type="Map">
      <Entries>
        <Entry>
          <Key type="Int" value="2"/>
          <Value type="String" value="Hello!!"/>
        </Entry>
        <Entry>
          <Key type="Int" value="5"/>
          <Value type="String" value="Goodbye!!"/>
        </Entry>
      </Entries>
    </Element>
  </Message>
</Messages>
```

6.14.2.2 AMQP Subscriber


AMQP Publisher: Verwaltet Abonnements für eine AMQP-Warteschlange und definiert den auszuführenden Prozess.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Subscribe: AMQP-Warteschlange abonnieren und den auszuführenden Prozess definieren  Erforderliche Parameter sind <code>host</code>, <code>port</code>, <code>username</code>, <code>password</code>, <code>queue</code> und <code>process</code>. Die Parameter können auch über eine Konfigurationsdatei bereitgestellt werden, die im Parameter <code>storedConfig</code> referenziert wird. • Unsubscribe: AMQP-Warteschlange abkündigen  Erforderliche Parameter sind <code>username</code> und <code>queue</code>. Die Parameter können auch über eine Konfigurationsdatei bereitgestellt werden, die im Parameter <code>storedConfig</code> referenziert wird.
-----------	--

Parameter

Adapter	<p>Adapter-Hauptklasse (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.amqp.AmqpSubscriberAdapter</code>: Hauptklasse (Standard)</p>
storedConfig	<p>Bezeichner der Konfiguration, bereitgestellt in der Datei <code>amqpconfig.xml</code></p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Bezeichner, z. B. <code>queueEmployeesConfig</code></p>
host	<p>AMQP Server-Host</p> <p>Mögliche Werte: Server-Host, z. B. <code>localhost</code></p>
port	<p>Dedizierter AMQP-Port auf dem Server</p> <p>Mögliche Werte: AMQP-Port-Nummer, z. B. <code>5672</code></p>
username	<p>Name des Benutzers, der die Nachricht abonniert bzw. abkündigt</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Benutzernamen, z. B. <code>user</code></p>
password	<p>Password des Benutzers</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Passwort, z. B. <code>p4ssw0rd</code></p>
queue	<p>Warteschlange, in der die Nachricht abonniert bzw. abkündigt wird</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Namen der Warteschlange, z. B. <code>employees</code></p>

container	<p>Container, in dem sich die Warteschlange befindet</p> <p>Mögliche Werte: Beliebige Zeichenkette mit dem Container-Namen z. B. <i>employeesContainer</i></p>
qos	<p>Quality of Service der Nachricht</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • AT_MOST_ONCE: Die Nachricht wird höchstens ein Mal oder kein einziges Mal zugestellt. • AT_LEAST_ONCE: Die Nachricht wird mindestens ein Mal zugestellt. • EXACTLY_ONCE: Die Nachricht wird nur ein Mal zugestellt. <p> Ist der Parameter qos nicht angegeben, ist der Standard-Wert AT_MOST_ONCE.</p>
process	<p>Pfad des technischen Prozesses, der ausgeführt wird, sobald eine Nachricht empfangen wird</p> <p>Mögliche Werte: Prozesspfad, z. B. <i>x4db://1/Project/Processes/process.wrf</i></p>

Statuswerte

0	Die Operation wurde erfolgreich ausgeführt (es wird keine Ausgabe generiert)
-1	Während der Ausführung der Operation kam es zu einem Fehler (es wird keine Ausgabe generiert)

Konfiguration

Um die benötigten Informationen bereitzustellen, kann eine Konfigurationsdatei verwendet werden, die beim Serverstart geladen wird. Die Konfigurationsdatei `amqpconfig.xml` muss unter dem Pfad `X4DB/0` bereitgestellt werden.

Die Datei hat folgende Struktur:

amqpconfig.xml

```
<?xml version='1.0' encoding='UTF-8'?>
<RootElement>
  <AMQPConfig name="queueEmployeesConfig">
    <Server host="localhost" port="5672"/>
    <Client user="user" password="p4ssw0rd"/>
    <Queue container="employeesContainer" name="employees" qos="AT_LEAST_ONCE"/>
  </AMQPConfig>
</RootElement>
```

6.14.3 JMS



JMS JMS: Tauscht Nachrichten über die *Java Message Service*-Warteschlange eines nachrichtenorientierten Middleware-Produkts aus, das als JMS-Provider dient. Der Adapter unterstützt das Senden und Empfangen von Nachrichten von und zur JMS-Warteschlange und kann Nachrichten in JMS-Topics publizieren und JMS-Topics abonnieren.

Ehemalige Bezeichnung bis Version 4.5.1: *JMS Transfer Adapter*

Ehemalige Bezeichnung bis Version 5.5.4: *JMS Connector*

Eigenschaften (Properties)

<i>Delete</i>	(keine Funktion)
<i>File name</i>	(keine Funktion)
<i>I/O Mode</i>	(keine Funktion)
<i>Operation</i>	<p>Bestimmt, welche Operation mit den Nachrichten der JMS-Warteschlange durchgeführt wird.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Send: Sendet Daten aus dem vorangegangenen Prozess-Schritt an die JMS-Warteschlange • Receive: Liest JMS-Nachricht in den Prozess ein • List: Gibt die Zahl der Nachrichten in der Warteschlange als XML-Dokument aus (falls Parameter <code>toXML</code> gesetzt ist, werden stattdessen die Header-Informationen der Nachrichten ausgegeben).

Parameter

<i>object_name</i>	<p>Name der JMS-Warteschlange oder des JMS-Topics, unter dem es im JNDI-Provider (Java Naming and Directory Interface) registriert ist</p> <p>Mögliche Werte: Beliebige Zeichenkette (z. B. <i>queue/A</i>)</p>
<i>connection_factory</i>	<p>Name der <i>JMS Connection Factory</i>, unter der die Nachrichtenwarteschlange im JNDI (Java Naming and Directory Interface) registriert ist</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette (z. B. <i>java:/JmsXA</i>)</p>

<i>JNDIProperties</i>	<p>Zusätzliche Eigenschaften des JNDI-Providers</p> <p>Mögliche Werte: Beliebige Zeichenkette mit Eigenschaften, bei mehreren Eigenschaften durch Semikolon getrennt (z. B. <code>java.naming.factory.initial=org.jnp.interfaces.NamingContextFactory;java.naming.provider.url=localhost:1099</code>)</p>
<i>approach</i>	<p>Verwendeter Nachrichtenkanaltyp</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>topic</code>: Punkt-zu-Punkt-Verbindung zwischen einem Sender und einem Empfänger, welcher nach Erhalt die Nachricht normalerweise aus dem Nachrichtenkanal entfernt • <code>queue</code>: Themenorientierte Publizieren/Abonnieren-Verbindung, bei der Nachrichten im Nachrichtenkanal erhalten bleiben, um auch von anderen Empfängern abgerufen werden zu können, bis die Nachrichten ersetzt werden oder verfallen (Standard)
<i>timeout</i>	<p>Zeitbeschränkung für den Verbindungsaufbau in Millisekunden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl • 0: keine Beschränkung; Vorsicht: Wenn die Warteschlange leer ist, dann "steht" der Prozess mit dieser Einstellung! • 10000: Maximal 10 Sekunden warten (Standard)
<i>filter</i>	<p>Filtert beim Empfangen von Nachrichten über JMS-Nachrichtenselektoren</p> <p>Mögliche Werte: Beliebige Zeichenkette mit Eigenschaften nach dem Muster <code><Eigenschaft1>=<Wert1>;<Eigenschaft2>=<Wert2></code> (z. B. <code>JMSPriority=5</code>)</p>
<i>msg_property</i>	<p>Header-Attribute der JMS-Nachricht beim Senden von Nachrichten</p> <p>Mögliche Werte: Beliebige Zeichenkette mit Eigenschaften nach dem Muster <code><Eigenschaft1>=<Wert1>;<Eigenschaft2>=<Wert2></code></p>
<i>headers_from_XML</i>	<p>Header-Informationen des Input-Dokuments verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>ja</code>: Header-Informationen verwenden • <code>nein</code>: Keine Header-Informationen verwenden (Standard)

<i>transacted</i>	<p>Diesen Prozess-Schritt in einer lokalen Transaktion ausführen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ja: Als lokale Transaktion ausführen • nein: Keine lokale Transaktion ausführen (Standard)
<i>ack_mode</i>	<p>Acknowledgement-Modus (dieser regelt den Umgang mit Empfangsbestätigungen)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • AUTO_ACKNOWLEDGE Auto acknowledge: Automatisch den Empfang von Nachrichten bestätigen • CLIENT_ACKNOWLEDGE Client acknowledge: Empfang durch einen Client bestätigen • DUPS_OK_ACKNOWLEDGE Dups ok acknowledge: Mehrfaches Empfangen einer Nachricht erlauben
<i>msg_priority</i>	<p>Nachrichten-Priorität (9 hat die höchste Priorität)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahlen von 0–9 • 4: Priorität 4 (Standard)
<i>msg_expiration_time</i>	<p>Gültigkeitsdauer von Nachrichten (Lebensdauer) in Millisekunden; nach dem Ablauf der Gültigkeit ist die Nachricht obsolet und wird automatisch aus der Warteschlange entfernt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige positive ganze Zahl • 0: keine Gültigkeitsbeschränkung (Standard)
<i>msg_type</i>	<p>JMS-Nachrichtentyp</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • text: Daten als Zeichenketten (etwa als XML-Dokumente) austauschen (Standard) • byte: Byte-Feldgruppe für Datenaustausch in nativen/ binären Formaten • map: JMS-MapMessages (Key-Value-Paare) versenden

<i>subscription_name</i>	<p>Name des Abonnements für dauerhafte Nachrichten-Abonnements (wenn Parameter approach auf <i>topic</i> gesetzt wurde)</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>durable_subscription</i>	<p>Dauer des JMS-Abonnements</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ja: dauerhaft beziehen • nein: temporär beziehen
<i>ClientID</i>	<p>Name des JMS-Clients</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>toXML</i>	<p>JMS-Nachricht inklusive Header-Informationen als XML-Dokument ausgeben (wenn Eigenschaft Operation auf <i>Receive</i> gesetzt wurde, dann wird der Zeichensatz unverändert übernommen; als MIME-Type wird dann <i>Text/XML</i> verwendet). Wenn Sie den Parameter toXML nicht setzen, dann werden Byte-, Stream- oder Object-Nachrichten als MIME-Typ <i>Application/Octet-stream</i> deklariert; Text-Nachrichten werden hingegen UTF-8-kodiert und mit MIME-Typ <i>Text/Plain</i> deklariert.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ja: XML-Dokument ausgeben • nein: Kein XML-Dokument ausgeben (Standard)
<i>base64</i>	<p>JMS-Nachricht Base64-kodiert ausgeben (wenn Parameter toXML gesetzt ist)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ja: Nachricht Base64-kodiert ausgeben • nein: Keine Base64-kodierte Nachricht ausgeben (Standard)
<i>JMSReplyTo</i>	Ziel-Warteschlange der Antwort-Nachricht
<i>JMSCorrelationID</i>	ID der Nachricht, auf die sich die Antwort bezieht (verschiedene Teile einer Nachricht besitzen dieselbe JMSCorrelationID)
<i>login</i>	Benutzername für den JMS
<i>password</i>	Passwort für den JMS

❗ Die im Applikations-Server standardmäßig verfügbaren JMS-Warteschlangen sind wegen des verwendeten Datenbanksystems (HSQLDB, Derby) nicht für den Produktiveinsatz geeignet. Trotz transaktionaler Verarbeitung können mit diesen Daten verloren gehen. Verwenden Sie ausschließlich produktiv einsetzbare Datenbanksysteme (Oracle, MSSQL etc.) zur Nachrichten-Persistierung von JMS-Warteschlangen; siehe http://docs.jboss.org/jbossas/docs/Server_Configuration_Guide/4/html/apbs04.html.

Input: msg:type map

```
<?xml version="1.0" encoding="UTF-8"?>
<RootElement>
  <Entry name="firstKey">Blub</Entry>
  <Entry name="b" type="byte">1</Entry>
  <Entry name="s" type="short">2</Entry>
  <Entry name="i" type="integer">3</Entry>
  <Entry name="l" type="long">4</Entry>
  <Entry name="f" type="float">5.5</Entry>
  <Entry name="d" type="double">6.6</Entry>
  <Entry name="bool" type="boolean">true</Entry>
  <Entry name="c" type="character">z</Entry>
  <Entry name="blob" type="bytearray">aaaa</Entry>
</RootElement>
```

Input: msg:type map und headers_from_XML

```
<?xml version="1.0" encoding="UTF-8"?>
<messages>
  <message type="map">
    <headers>
      <header>
        <name>myFirstHeader</name>
        <value>the first</value>
      </header>
      <header>
        <name>mySecondHeader</name>
        <value>the second</value>
      </header>
    </headers>
    <body>
      <Entry name="firstKey">Blub</Entry>
      <Entry type="byte" name="b">1</Entry>
      <Entry type="short" name="s">2</Entry>
      <Entry type="integer" name="i">3</Entry>
      <Entry type="long" name="l">4</Entry>
      <Entry type="float" name="f">5.5</Entry>
      <Entry type="double" name="d">6.6</Entry>
      <Entry type="boolean" name="bool">true</Entry>
      <Entry type="character" name="c">z</Entry>
      <Entry type="bytearray" name="blob">aaaa</Entry>
    </body>
  </message>
</messages>
```

Output: msg:type map

```
<?xml version="1.0" encoding="UTF-8"?>
<MapMessage>
  <Entry name="b" type="Byte">1</Entry>
  <Entry name="blob" encoding="base64" type="ByteArray">aaaa</Entry>
  <Entry name="s" type="Short">2</Entry>
  <Entry name="c" type="Character">z</Entry>
  <Entry name="bool" type="Boolean">true</Entry>
  <Entry name="d" type="Double">6.6</Entry>
  <Entry name="f" type="Float">5.5</Entry>
  <Entry name="firstKey" type="String">Blub</Entry>
  <Entry name="i" type="Integer">3</Entry>
  <Entry name="l" type="Long">4</Entry>
</MapMessage>
```

Input: msg:type map und headers_to_XML

```

<?xml version="1.0" encoding="UTF-8"?>
<document>
  <headers>
    <header>
      <name>mySecondHeader</name>
      <value>the second</value>
    </header>
    <header>
      <name>JMSMessageID</name>
      <value>ID:4bedf9a6-05a1-11e7-bf59-eba4f4962570</value>
    </header>
    <header>
      <name>JMSXDeliveryCount</name>
      <value>1</value>
    </header>
    <header>
      <name>myFirstHeader</name>
      <value>the first</value>
    </header>
  </headers>
  <body>
    <MapMessage>
      <Entry name="b" type="Byte">1</Entry>
      <Entry name="blob" encoding="base64" type="ByteArray">aaaa</Entry>
      <Entry name="s" type="Short">2</Entry>
      <Entry name="c" type="Character">z</Entry>
      <Entry name="bool" type="Boolean">true</Entry>
      <Entry name="d" type="Double">6.6</Entry>
      <Entry name="f" type="Float">5.5</Entry>
      <Entry name="firstKey" type="String">Blub</Entry>
      <Entry name="i" type="Integer">3</Entry>
      <Entry name="l" type="Long">4</Entry>
    </MapMessage>
  </body>
</document>

```

6.14.4 Websphere MQ

6.14.4.1 IBM MQ Monitoring



IBM MQ Monitoring: Überwacht einen *IBM Websphere MQ Server* mittels übergebener Kommandos.



Abhängigkeiten

Der Adapter benötigt die kostenpflichtige Bibliothek `com.ibm.mq.allclient.jar` im Classpath.

Eigenschaften

Operation	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <ul style="list-style-type: none">• <i>Query</i>: Führt die im XML-Dokument übergebenen Kommandos aus und gibt diese in einem XML-Dokument aus
-----------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.ibmmq.MQChannelChecker</i> : Hauptklasse (Standard)
<i>MonitorIp</i>	IP-Adresse des zu überwachenden MQ Servers Mögliche Werte: eine beliebige Zeichenkette (z. B. <i>127.0.0.1</i>)
<i>MonitorPort</i>	Port des zu überwachenden MQ Servers Mögliche Werte: eine beliebige positive Zahl (z. B. <i>4711</i>)
<i>MonitorChannel</i>	Name des Channels, der auf dem MQ Server überwacht werden soll Mögliche Werte: eine beliebige Zeichenkette (z. B. <i>Channel1</i>)

Statuswerte

1 (<i>successful</i>)	Alle Befehle wurden erfolgreich durchgeführt
-1 (<i>failed</i>)	Die Anfrage schlug aufgrund eines technischen Fehlers fehl

Input

Dieser Adapter erwartet ein XML-Dokument als Input, das wie folgt aufgebaut ist:

```
<Commands>
  <{Command} />
</Commands>
```

Folgende Kommandos werden unterstützt:

- *InquireQueue*
- *InquireChannelNames*

•

*InquireChannelStatus***Beispiel für die drei Kommandos:**

```
<Commands>
  <InquireQueue queue="{Wert}" />
  <InquireChannelNames channel="{Wert}" />
  <InquireChannelStatus channel="{Wert}" />
</Commands>
```

Output

Der Adapter erweitert im ausgegebenen XML-Dokument das Kommando-Element um das jeweilige Ergebnis.

Beispiel

```
<Commands>
  <InquireQueue queue="{Wert}">
    <Message>
      <Value name="{Wert}">{Wert}</Value>
    </Message>
  </InquireQueue>
  <InquireChannelNames channel="{Wert}">
    <NoResult />
  </InquireChannelNames>
  <InquireChannelStatus channel="{Wert}">
    <Message>
      <Value name="{Wert}">{Wert}</Value>
      <Value name="{Wert}">{Wert}</Value>
    </Message>
  </InquireChannelStatus>
</Commands>
```

•

6.14.4.2 MQ Series Transfer

MQ Series Transfer: Sendet und empfängt Nachrichten über eine *IBM Websphere MQ* (MQ Series) Nachrichten-Warteschlange und unterstützt MQ-spezifische Funktionen wie z. B. *Dynamic Queues*.


Ehemalige Bezeichnung bis Version 4.5.1: *MQ Series Transfer Adapter*

Eigenschaften (Properties)


<i>Delete</i>	(keine Funktion)
<i>File name</i>	(keine Funktion)
<i>I/O Mode</i>	(keine Funktion)
<i>Operation</i>	<p>Bestimmt, welche Operation mit der Nachrichtenwarteschlange durchgeführt wird.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Send</i>: Daten aus dem vorangegangenen Prozess-Schritt an die <i>MQ Series</i>-Warteschlange senden • <i>Receive</i>: Nachricht in den Prozess einlesen • <i>Commit</i>: Transaktion abschließen • <i>Rollback</i>: Transaktion abbrechen und sämtliche Änderungen rückgängig machen • <i>Create Queue</i>: Warteschlange erstellen • <i>Destroy Queue</i>: Warteschlange entfernen • <i>Clear</i>: Verbindung zu MQ Series zurücksetzen

Parameter

<i>alias</i>	Name der Parametergruppe zum externen Konfigurieren des Adapters über die Datei <code>QueueConfig.xml</code> (in <code>x4.ear</code> im Unterordner <code>conf</code>). Dort angegebene Werte besitzen eine höhere Priorität als die Parameter des Transfer-Adapter-Prozessbausteins.
<i>toXML</i>	<p>Wenn Eigenschaft <i>Operation</i> auf <i>Receive</i> gesetzt wurde: Ergebnis des Prozess-Schritts als XML-Dokument ausgeben; Wenn zudem Parameter <code>MQMessage.format</code> auf <i>NONE</i> gesetzt wurde, dann wird die Nachricht Base64-kodiert</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: XML-Dokument generieren, das Header und Body der Nachricht mit sämtlichen Eigenschaften enthält (siehe Beispiel) • <i>false</i>: Als Ergebnis des Prozess-Schritts eine MQ-Nachricht generieren (Standard)

<i>ccdtUrl</i>	<p>URL zu einer <i>Client Channel Definition Table (CCDT)</i></p> <p> Es gibt vier verschiedene Möglichkeiten, die <i>ccdtUrl</i> zu definieren:</p> <ol style="list-style-type: none"> Über die Java-Argumente <code>-DMQCHLLIB</code> und <code>-DMQCHLTAB</code> beim Start des <i>X4 Servers</i> Mögliche Werte: <ul style="list-style-type: none"> <code>-DMQCHLLIB</code>: Pfad zur Table-Datei <code>-DMQCHLTAB</code>: <i>AMQCLCHL.TAB</i> (IBM Default-Name) Durch das Java-Argument <code>-Dx4.mqseries.ccdtUrl</code> Beispiel: <code>-Dx4.mqseries.ccdtUrl=file:///Pfad_zur_Tabelle/AMQCLCHL.TAB</code> Über die Eigenschaft <i>ccdtUrl</i> in der <i>QueueConfig.xml</i> Über den Parameter <i>ccdtUrl</i> im Adapter selbst <p>Dabei wird die oben genannte Priorität angewendet.</p>
<i>RemoteQueueManager.name</i>	<p>Name des Queue-Managers</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> Beliebige gültige Zeichenkette <code>{EMPTY}</code>: Explizit kein Wert (leere Zeichenkette)
<i>MQEnviroment.hostname</i>	<p>Hostname des angesprochenen MQ-Servers; gilt systemweit innerhalb der gesamten Java-VM für alle folgenden Operationen (der Parameter wird nur für lokale Tests auf Entwicklungssystemen benötigt, sonst leer lassen!)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> Beliebige Zeichenkette (z. B. <i>MQSR2402</i>) <code>{EMPTY}</code>: Explizit kein Wert (leere Zeichenkette)
<i>MQEnviroment.channel</i>	<p>Name des Channels; gilt systemweit innerhalb der gesamten Java-VM für alle folgenden Operationen (nur für lokale Tests auf Entwicklungssystemen benötigt, sonst leer lassen!)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> Beliebige Zeichenkette (z. B. <i>MQSR2402.CLS25740.DT</i>) <code>{EMPTY}</code>: Explizit kein Wert (leere Zeichenkette)

<i>MQEnviroment.CCSID</i>	<p>CCSID (Charset) aller Nachrichten für die gesamte Java-VM; gilt systemweit innerhalb der gesamten Java-VM für alle folgenden Operationen, deshalb Vorsicht bei der Benutzung (sollte nicht benutzt werden!)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige CCSID (z. B. <i>1208</i>) • <i>{EMPTY}</i>: Explizit kein Wert (leere Zeichenkette)
<i>MQEnviroment.userID</i>	<p>Benutzer-ID zur Authentifizierung des <i>WebSphere MQ Clients</i>; gilt systemweit innerhalb der gesamten Java-VM für alle folgenden Operationen (nur für lokale Tests auf Entwicklungssystemen benötigt, sonst leer lassen!)</p>
<i>MQEnviroment.localAddressSetting</i>	<p>MQ-spezifischer Parameter; gilt systemweit innerhalb der gesamten Java-VM für alle folgenden Operationen (nur für lokale Tests auf Entwicklungssystemen benötigt, sonst leer lassen!)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige gültige Zeichenkette • <i>{EMPTY}</i>: Explizit kein Wert (leere Zeichenkette)
<i>MQEnviroment.password</i>	<p>Passwort zur Authentifizierung des <i>WebSphere MQ Clients</i>; gilt systemweit innerhalb der gesamten Java-VM für alle folgenden Operationen (nur für lokale Tests auf Entwicklungssystemen benötigt, sonst leer lassen!)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige gültige Zeichenkette • <i>{EMPTY}</i>: Explizit kein Wert (leere Zeichenkette)
<i>MQEnviroment.port</i>	<p>Port des MQServers; gilt systemweit innerhalb der gesamten Java-VM für alle folgenden Operationen (nur für lokale Tests auf Entwicklungssystemen benötigt, sonst leer)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl (z. B. <i>1414</i>) • <i>{EMPTY}</i>: Explizit kein Wert (leere Zeichenkette)

<p><i>MQEnvironment. properties.MQC.SSL_ CIPHER_SUITE_PROPERTY</i></p>	<p>CipherSuite-Konfiguration für die aktuelle Verbindung. Die eingetragene CipherSuite muss dabei der verwendeten CipherSpec entsprechen:</p> <p>CipherSpec und die entsprechende CipherSuite</p> <ul style="list-style-type: none"> • <i>NULL_MD5: SSL_RSA_WITH_NULL_MD5</i> • <i>NULL_SHA: SSL_RSA_WITH_NULL_SHA</i> • <i>RC4_MD5_EXPORT: SSL_RSA_EXPORT_WITH_RC4_40_MD5</i> • <i>RC4_MD5_US: SSL_RSA_WITH_RC4_128_MD5</i> • <i>RC4_SHA_US: SSL_RSA_WITH_RC4_128_SHA</i> • <i>RC2_MD5_EXPORT: SSL_RSA_EXPORT_WITH_RC2_CBC_40_MD5</i> • <i>DES_SHA_EXPORT: SSL_RSA_WITH_DES_CBC_SHA</i> • <i>RC4_56_SHA_EXPORT1024: SSL_RSA_EXPORT1024_WITH_RC4_56_SHA</i> • <i>DES_SHA_EXPORT1024: SSL_RSA_EXPORT1024_WITH_DES_CBC_SHA</i> • <i>TRIPLE_DES_SHA_US: SSL_RSA_WITH_3DES_EDE_CBC_SHA</i> • <i>TLS_RSA_WITH_AES_128_CBC_SHA: SSL_RSA_WITH_AES_128_CBC_SHA</i> • <i>TLS_RSA_WITH_AES_256_CBC_SHA: SSL_RSA_WITH_AES_256_CBC_SHA</i> • <i>AES_SHA_US</i> • <i>TLS_RSA_WITH_DES_CBC_SHA: SSL_RSA_WITH_DES_CBC_SHA</i> • <i>TLS_RSA_WITH_3DES_EDE_CBC_SHA: SSL_RSA_WITH_3DES_EDE_CBC_SHA</i> • <i>FIPS_WITH_DES_CBC_SHA: SSL_RSA_FIPS_WITH_DES_CBC_SHA</i> • <i>FIPS_WITH_3DES_EDE_CBC_SHA: SSL_RSA_FIPS_WITH_3DES_EDE_CBC_SHA</i> <div style="border: 1px solid #add8e6; padding: 10px; margin-top: 10px;"> <p> Zusätzliche Informationen zur CipherSuite-Konfiguration erhalten Sie auch hier (Kapitel 4)</p> </div> <p>Mögliche Werte: CipherSuite als gültige Zeichenkette, s.o.</p>
<p><i>MQEnvironment. properties.MQC. TRANSPORT_PROPERTY</i></p>	<p>Einstellung für die aktuelle Verbindung</p> <p>Mögliche Werte: <i>MQC.TRANSPORT_MQSERIES</i>: Einziger zulässiger Wert</p>
<p><i>MQEnvironment. properties.MQC. CCSID_PROPERTY</i></p>	<p>CCSID (Charset) für die aktuelle Verbindung</p> <p>Mögliche Werte: Beliebige ganze Zahl (z. B. 1208)</p>
<p><i>MQEnvironment. properties.MQC. USER_ID_PROPERTY</i></p>	<p>Benutzer-ID für die aktuelle Verbindung</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>

<i>MQEnviroment. properties.MQC. CHANNEL_PROPERTY</i>	Name des Channels für die aktuelle Verbindung Mögliche Werte: Beliebige Zeichenkette (z. B. <i>MQSR2402.CLS25740.DT</i>)
<i>MQEnviroment. properties.MQC. CONNECT_OPTIONS_ PROPERTY</i>	Verbindungsoptionen für die aktuelle Verbindung Mögliche Werte: Beliebige Zeichenkette
<i>MQEnviroment. properties.MQC. HOST_NAME_PROPERTY</i>	Hostname des angesprochenen MQ-Servers für die aktuelle Verbindung Mögliche Werte: Beliebige Zeichenkette (z. B. <i>MQSR2402</i>)
<i>MQEnviroment. properties.MQC. LOCAL_ADDRESS_ PROPERTY</i>	Lokale Adresse für die aktuelle Verbindung Mögliche Werte: Beliebige Zeichenkette
<i>MQEnviroment. properties.MQC. ORB_PROPERTY</i>	ORB-Eigenschaft für die aktuelle Verbindung Mögliche Werte: Beliebige Zeichenkette
<i>MQEnviroment. properties.MQC. PASSWORD_PROPERTY</i>	Passwort-Eigenschaft für den aktuellen Verbindungsaufbau, um den <i>WebSphere MQ Client</i> zu authentifizieren (entspricht Parameter <i>MQEnviroment.password</i>) Mögliche Werte: Beliebige Zeichenkette
<i>MQEnviroment. properties.MQC. PORT_PROPERTY</i>	Port für die aktuelle Verbindung Mögliche Werte: Beliebige Zeichenkette
<i>MQEnviroment. properties.MQC. RECEIVE_EXIT_ PROPERTY</i>	<i>Receive Exit</i> -Eigenschaft für die aktuelle Verbindung Mögliche Werte: Beliebige Zeichenkette
<i>MQEnviroment. properties.MQC. SECURITY_EXIT_ PROPERTY</i>	<i>Security Exit</i> -Eigenschaft für die aktuelle Verbindung Mögliche Werte: Beliebige Zeichenkette
<i>MQEnviroment. properties.MQC. SEND_EXIT_PROPERTY</i>	<i>Send Exit</i> -Eigenschaft für die aktuelle Verbindung Mögliche Werte: Beliebige Zeichenkette
<i>MQQueueManager.name</i>	Name des QueueManagers. Nur für lokale Tests auf Entwicklungssystemen notwendig, sonst leer.
<i>MQQueue.name</i>	Name der Queue

<i>MQQueue.dynamic QueueName</i>	Name der Session Queue für diese Nachricht (Sollte zukünftig nicht mehr verwendet werden)
<i>MQQueue.alternate UserId</i>	Alternative User-ID um den Zugriff auf die Warteschlange zu prüfen, wenn diese geöffnet ist; dieser Parameter kann nicht gesetzt werden, während das Objekt geöffnet ist
<i>MQQueue.openOptions. MQC.MQOO_OUTPUT</i>	(Muss bei schreibendem Zugriff gesetzt werden)
<i>MQQueue.openOptions. MQC.MQOO_FAIL_ IF QUIESCING</i>	Erlaubt dem MQ-Manager, den Listener zu trennen (z. B. beim Herunterfahren)
<i>MQQueue.openOptions. MQC.MQDL_SUPPORTED</i>	MQ-spezifischer Parameter
<i>MQQueue.openOptions. MQC.MQOO_ALTERNATE_ USER_AUTHORITY</i>	MQ-spezifischer Parameter
<i>MQQueue.openOptions. MQC.MQOO_BIND_AS_ QDEF</i>	Queue-Einstellungen zur Persistenz etc. übernehmen (sollte immer gesetzt werden) Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: Queue-Einstellungen übernehmen • <i>nein</i>: Keine Queue-Einstellungen übernehmen (Standard)
<i>MQQueue.openOptions. MQC.MQOO_BIND_ NOT_FIXED</i>	MQ-spezifischer Parameter
<i>MQQueue.openOptions. MQC.MQOO_BIND_ON_ OPEN</i>	MQ-spezifischer Parameter
<i>MQQueue.openOptions. MQC.MQOO_INPUT_AS_ Q_DEF</i>	Einstellungen der Queue zur Persistenz etc. übernehmen (muss beim Lesen immer gesetzt werden) Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: Queue-Einstellungen übernehmen • <i>nein</i>: Keine Queue-Einstellungen übernehmen (Standard)
<i>MQQueue.openOptions. MQC.MQOO_INPUT_SHARED</i>	MQ-spezifischer Parameter
<i>MQQueue.openOptions. MQC.MQOO_INPUT_ EXCLUSIVE</i>	MQ-spezifischer Parameter
<i>MQQueue.openOptions. MQC.MQOO_INQUIRE</i>	MQ-spezifischer Parameter
<i>MQQueue.openOptions. MQC.MQOO_PASS_ALL_ CONTEXT</i>	MQ-spezifischer Parameter

<i>MQQueue.openOptions. MQC.MQOO_PASS_ IDENTITY_CONTEXT</i>	MQ-spezifischer Parameter
<i>MQQueue.openOptions. MQC.MQOO_SAVE_ALL_ CONTEXT</i>	MQ-spezifischer Parameter
<i>MQQueue.openOptions. MQC.MQOO_SET</i>	MQ-spezifischer Parameter
<i>MQQueue.openOptions. MQC.MQOO_SET_ALL_ CONTEXT</i>	MQ-spezifischer Parameter
<i>MQQueue.openOptions. MQC.MQOO_SET_ IDENTITY_CONTEXT</i>	MQ-spezifischer Parameter
<i>MQPutMessageOptions. options.MQC.MQPMO_ NEW_MSG_ID</i>	MQ-spezifischer Parameter
<i>MQPutMessageOptions. options.MQC. MQPMO_SYNCPOINT</i>	<p>Syncpoint beim Schreiben verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Syncpoint beim Schreiben verwenden • <i>nein</i>: Keinen Syncpoint beim Schreiben verwenden (Standard)
<i>MQQueue. closeOptions.MQC. MQCOO_DELETE</i>	MQ-spezifischer Parameter
<i>MQQueue. closeOptions.MQC. MQCOO_DELETE_ PURGE</i>	MQ-spezifischer Parameter

<i>MQMessage.format</i>	<p>Nachrichtenformat</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>MQC.MQFMT_STRING</i>: Nachricht besteht komplett aus Zeichen • <i>MQC.MQFMT_ADMIN</i>: Befehl-Server-Anfrage/Antwort • <i>MQC.MQFMT_COMMAND1</i>: 1 Befehl-Server-Antwortnachricht vom Typ 1 • <i>MQC.MQFMT_COMMAND2</i>: 2 Befehl-Server-Antwortnachricht vom Typ 2 • <i>MQC.MQFMT_DEAD_LETTER_HEADER</i>: Dead-letter Header • <i>MQC.MQFMT_EVENT</i>: Event • <i>MQC.MQFMT_NONE</i>: Kein Formatname, es findet keine Konvertierung statt (d. h. die Bytes werden original geliefert); wenn zudem Parameter toXML aktiviert ist, dann wird die Nachricht Base64-kodiert • <i>MQC.MQFMT_PCF</i>: Benutzerdefinierte Nachricht im <i>Programmable Command Format</i> • <i>MQC.MQFMT_TRIGGER</i>: Trigger-Nachricht • <i>MQC.MQFMT_XMIT_Q_HEADER</i>: Transmission Queue Header • <i>(kein Wert)</i>: Nicht konvertieren (entspricht <i>MQC.MQFMT_NONE</i>) (Standard)
<i>MQMessage.applicationIdData</i>	MQ-spezifischer Parameter
<i>MQMessage.applicationOriginData</i>	MQ-spezifischer Parameter
<i>MQMessage.backoutCount</i>	MQ-spezifischer Parameter
<i>MQMessage.messageFlags.MQC.MQMT_REQUEST</i>	MQ-spezifischer Parameter
<i>MQMessage.replyToQueueName</i>	Name der ReplyToQueue für diese Nachricht
<i>MQMessage.replyToQueueManagerName</i>	<p>Name des ReplyToQueueManagers für diese Nachricht</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>MQMessage.characterSet</i>	<p>Charset (CCSID) für die Nachricht; beim Auslesen: das erwartete Charset; Beim Schreiben: das Charset, das die Nachricht bekommt</p> <p>Mögliche Werte: Beliebige CCSID (z. B. <i>1208</i>)</p>

<i>MQMessage.correlationId</i>	<p>CorrelationID mit 24 Byte Länge (diese ist ebenso wie die zurückgegebene MessageID Base64-kodiert)</p> <p>Mögliche Werte: Beliebige Zeichenkette mit 32 Zeichen, die eine Base64-kodierte CorrelationID mit 24 Byte Länge enthält</p>
<i>MQMessage.encoding</i>	Zeichenkodierung der Nachricht
<i>MQMessage.expiry</i>	<p>Zeit in Zehntelsekunden, nach der die Nachricht ungültig wird und aus der Queue gelöscht wird.</p> <p>Mögliche Werte: Beliebige ganze Zahl</p>
<i>MQMessage.feedback</i>	MQ-spezifischer Parameter
<i>MQMessage.groupId</i>	MQ-spezifischer Parameter
<i>MQMessage.messageId</i>	MQ-spezifischer Parameter
<i>MQMessage.messageSequenceNumber</i>	MQ-spezifischer Parameter
<i>MQMessage.messageType</i>	MQ-spezifischer Parameter
<i>MQMessage.offset</i>	MQ-spezifischer Parameter
<i>MQMessage.originalLength</i>	MQ-spezifischer Parameter
<i>MQMessage.persistence</i>	MQ-spezifischer Parameter
<i>MQMessage.priority</i>	MQ-spezifischer Parameter
<i>MQMessage.putApplicationType</i>	MQ-spezifischer Parameter
<i>MQMessage.report</i>	MQ-spezifischer Parameter
<i>MQMessage.userId</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions.options.MQC.MQGMO_SYNCPOINT</i>	<p>Syncpoint beim Empfangen von Nachrichten verwenden (erforderlich für die Operationen Commit und Rollback)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Syncpoint beim Empfangen verwenden • <i>nein</i>: Keinen Syncpoint beim Empfangen verwenden (Standard)

<i>MQGetMessageOptions. options.MQC. MQGMO_WAIT</i>	Beim Auslesen warten, wenn sich keine Nachricht in der Queue befindet
<i>MQGetMessageOptions. options.MQC. MQGMO_NO_WAIT</i>	<p>Beim Auslesen nicht warten, wenn sich keine Nachricht in der Queue befindet, und sofort abbrechen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Nicht warten und sofort abbrechen • <i>nein</i>: Warten (Standard)
<i>MQGetMessageOptions. options.MQC. MQGMO_ACCEPT_ TRUNCATED_MSG</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. options.MQC. MQGMO_BROWSE_FIRST</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. options.MQC. MQGMO_BROWSE_MSG_ _UNDER_CURSOR</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. options.MQC. MQGMO_BROWSE_NEXT</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. options.MQC. MQGMO_CONVERT</i>	<p>Automatische Konvertierung in MQ aktivieren</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Konvertierung aktivieren • <i>nein</i>: Keine Konvertierung (Standard)
<i>MQGetMessageOptions. options.MQC.MQGMO_ FAIL_IF QUIESCING</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. options.MQC. MQGMO_LOCK</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. options.MQC. MQGMO_MSG_UNDER_ CURSOR</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. options.MQC. MQGMO_NONE</i>	<p>Alle anderen <i>GetOptions</i> ignorieren</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Alle anderen <i>GetOptions</i> ignorieren • <i>nein</i>: Andere <i>GetOptions</i> nicht ignorieren (Standard)


<i>MQGetMessageOptions. options.MQC.MQGMO_ MARK_SKIP_BACKOUT</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. options.MQC. MQGMO_NO_SYNCPOINT</i>	<p>Syncpoint deaktivieren</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Syncpoint deaktivieren (damit ist keine Transaktion möglich) • <i>nein</i>: Syncpoint nicht deaktivieren (Standard)
<i>MQGetMessageOptions. options.MQC.MQGMO_ SYNCPOINT_IF_ PERSISTENT</i>	<p>Syncpoint nutzen, wenn die Nachricht als persistent gekennzeichnet ist</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Syncpoint bei persistenten Nachrichten nutzen • <i>nein</i>: Keinen Syncpoint bei persistenten Nachrichten nutzen (Standard)
<i>MQGetMessageOptions. options.MQC.MQGMO_ UNLOCK</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. options.MQC. MQGMO_ALL_MSGS_ AVAILABLE</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. options.MQC. MQGMO_ALL_SEGMENTS_ AVAILABLE</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. options.MQC. MQGMO_COMPLETE_MSG</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. options.MQC. MQGMO_LOGICAL_ORDER</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. matchOptions.MQC. MQGMO_MATCH_ CORREL_ID</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. matchOptions.MQC. MQGMO_MATCHGROUP_ID</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. matchOptions.MQC. MQGMO_MATCH_MSG_ID</i>	MQ-spezifischer Parameter

<i>MQGetMessageOptions. matchOptions.MQC. MQGMO_MATCH_MSG_ SEQ_NUMBER</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. matchOptions. MQC_MQGMO_NONE</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions. waitInterval</i>	Zeit in Millisekunden, die beim Auslesen gewartet wird, wenn sich keine Nachrichten in der Queue befinden (siehe Parameter <i>MQGetMessageOptions.options.MQC.MQGMO_WAIT</i>). Der Nachrichtenempfang wird nicht abgebrochen, wenn er während des Warte-Intervalls begonnen wurde. Mögliche Werte: Beliebige positive ganze Zahl
<i>MQGetMessageOptions. resolvedQueueName</i>	MQ-spezifischer Parameter

Beispiel

Das erzeugte XML-Dokument, wenn Eigenschaft *Operation* auf *Receive* gesetzt wurde, besitzt folgende Struktur:

```
<document>
<header>
  <CorrelationID>AAAAAAAAAAAAAAAAAAAA</CorrelationID>
  <MessageID>QU1RIFFX3NwX2Nvbml5YxvAdUQgC9sH</MessageID>
  <MessageSequenceNr>1</MessageSequenceNr>
  <MessageFormat>MQSTR</MessageFormat>
  <ReplyToQueue/>
  <ReplyToQueueMngr>QM_sp_connect</ReplyToQueueMngr>
  <backoutCount>0</backoutCount>
</header>
<body>Message</body>
</document>
```

 Weitere Informationen zu den *MQ Series*-spezifischen Parametern erhalten Sie von dessen Hersteller *IBM* in der englischsprachigen Dokumentation *WebSphere MQ – Using Java* in Kapitel 9 *The WebSphere MQ base Java classes and interfaces*.

6.14.4.3 Websphere MQ Connector



WebSphere MQ Connector: Sendet, empfängt und zählt Nachrichten einer *IBM Websphere MQ*-Nachrichten-Warteschlange und unterstützt dabei die *MQ*-Version 7.

Ehemalige Bezeichnung bis Version 4.5.1: *IBM Websphere MQ Adapter*


i Beachten Sie, dass dieser Adapter unabhängig von der Warteschlangen-Konfiguration des *MQ Series Transfer Adapters* (innerhalb von *x4.ear* erfolgt dies in der Datei *conf/QueueConfig.xml*) arbeitet.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation mit der Nachrichtenwarteschlange durchgeführt wird.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Send</i>: Daten aus dem vorangegangenen Prozess-Schritt an die <i>MQ Series</i>-Warteschlange senden • <i>Receive</i>: Nachricht in den technischen Prozess einlesen • <i>Commit</i>: Transaktion abschließen • <i>Rollback</i>: Transaktion abbrechen und sämtliche Änderungen rückgängig machen • <i>Count</i>: Anzahl der Nachrichten in der Warteschlange in einem XML-Dokument ausgeben
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.ibmmq.MQAdapter</i>: Hauptklasse (Standard)</p>
<i>toXML</i>	<p>Wenn Eigenschaft <i>Operation</i> auf <i>Receive</i> gesetzt wurde: Ergebnis des Prozess-Schritts als XML-Dokument ausgeben; Wenn zudem Parameter <i>MQMessage.format</i> auf <i>NONE</i> gesetzt wurde, dann wird die Nachricht Base64-kodiert</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: XML-Dokument generieren, das Header und Body der Nachricht mit sämtlichen Eigenschaften enthält (siehe Beispiel) • <i>false</i>: Als Ergebnis des Prozess-Schritts eine MQ-Nachricht generieren (Standard)

<i>ccdtUrl</i>	<p>URL zu einer <i>Client Channel Definition Table (CCDT)</i></p> <p></p> <p>Es gibt drei verschiedene Möglichkeiten, die <i>ccdtUrl</i> zu definieren:</p> <ol style="list-style-type: none"> Über die Java-Argumente <i>-DMQCHLLIB</i> und <i>-DMQCHLTAB</i> beim Start des <i>X4 Servers</i> Mögliche Werte: <i>-DMQCHLLIB</i>: Pfad zur <i>Table</i>-Datei <i>-DMQCHLTAB</i>: <i>AMQCLCHL.TAB</i> (IBM Default-Name) Durch das Java-Argument <i>-Dx4.mqseries.ccdtUrl</i> <i>Beispiel</i>: <i>-Dx4.mqseries.ccdtUrl=file:///Pfad_zur_Tabelle/AMQCLCHL.TAB</i> Über den Parameter <i>ccdtUrl</i> im Adapter selbst Dabei wird die oben genannte Priorität angewendet.
<i>RemoteQueueManager_name</i>	Name des Queue-Managers
<i>MQEnvironment_hostname</i>	<p>Hostname des angesprochenen MQ-Servers; gilt Java-VM-weit (der Parameter wird nur für lokale Tests auf Entwicklungssystemen benötigt, sonst bleibt er leer)</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <i>MQSR2402</i>)</p>
<i>MQEnvironment_channel</i>	<p>Name des Channels; gilt Java-VM-weit (nur für lokale Tests auf Entwicklungssystemen benötigt, sonst leer)</p> <p>Mögliche Werte: Beliebige Zeichenkette (z.B. <i>MQSR2402.CLS25740.DT</i>)</p>
<i>MQEnvironment_CCSID</i>	<p>CCSID (Charset) aller Nachrichten für die gesamte Java-VM; deshalb Vorsicht bei der Benutzung (sollte nicht benutzt werden!)</p> <p>Mögliche Werte: Beliebige CCSID (z.B. <i>1208</i>)</p>
<i>MQEnvironment_userID</i>	Benutzer-ID zur Authentifizierung des <i>WebSphere MQ Clients</i>
<i>MQEnvironment_localAddressSetting</i>	MQ-spezifischer Parameter
<i>MQEnvironment_password</i>	Passwort zur Authentifizierung des <i>WebSphere MQ Clients</i>
<i>MQEnvironment_port</i>	<p>Port des MQ-Servers; gilt Java-VM-weit (nur für lokale Tests auf Entwicklungssystemen benötigt, sonst leer)</p> <p>Mögliche Werte: Beliebige ganze Zahl (z.B. <i>1414</i>)</p>

<i>MQEnvironment_ properties_MQC_ TRANSPORT_PROPERTY</i>	Einstellung für die Verbindung Mögliche Werte: <i>MQC.TRANSPORT_MQSERIES</i> : Einziger zulässiger Wert
<i>MQEnvironment_ properties_MQC_ CCSID_PROPERTY</i>	CCSID (Charset) für die Verbindung Mögliche Werte: Beliebige ganze Zahl (z.B. 1208)
<i>MQEnvironment_ properties_MQC_ USER_ID_PROPERTY</i>	Benutzer-ID für die Verbindung Mögliche Werte: Beliebige Zeichenkette
<i>MQEnvironment_ properties_MQC_ CHANNEL_PROPERTY</i>	Name des Kanals für die Verbindung Mögliche Werte: Beliebige Zeichenkette (z.B. <i>MQSR2402.CLS25740.DT</i>)
<i>MQEnvironment_ properties_MQC_ CONNECT_OPTIONS_PROPERTY</i>	Verbindungsoptionen Mögliche Werte: Beliebige Zeichenkette
<i>MQEnvironment_ properties_MQC_ HOST_NAME_PROPERTY</i>	Hostname des angesprochenen MQ-Servers für die Verbindung Mögliche Werte: Beliebige Zeichenkette (z.B. <i>MQSR2402</i>)
<i>MQEnvironment_ properties_MQC_ LOCAL__ADDRESS_PROPERTY</i>	Lokale Adresse für die Verbindung Mögliche Werte: Beliebige Zeichenkette
<i>MQEnvironment_ properties_MQC_ ORB_PROPERTY</i>	ORB-Eigenschaft für die Verbindung Mögliche Werte: Beliebige Zeichenkette
<i>MQEnvironment_ properties_MQC_ PASSWORD_PROPERTY</i>	Passwort-Eigenschaft für den Verbindungsaufbau, um den <i>WebSphere MQ Client</i> zu authentifizieren (entspricht Parameter <i>MQEnvironment_password</i>) Mögliche Werte: Beliebige Zeichenkette
<i>MQEnvironment_ properties_MQC_ PORT_PROPERTY</i>	Port für die Verbindung Mögliche Werte: Beliebige Zeichenkette
<i>MQEnvironment_ properties_MQC_ RECEIVE_EXIT_PROPERTY</i>	<i>Receive Exit</i> -Eigenschaft für die Verbindung Mögliche Werte: Beliebige Zeichenkette

<i>MQEnvironment_</i> <i>properties_MQC_</i> <i>SECURITY_EXIT_PROPERTY</i>	<i>Security Exit</i> -Eigenschaft für die Verbindung Mögliche Werte: Beliebige Zeichenkette
<i>MQEnvironment_</i> <i>properties_MQC_</i> <i>SEND_EXIT_PROPERTY</i>	<i>Send Exit</i> -Eigenschaft für die Verbindung Mögliche Werte: Beliebige Zeichenkette
<i>MQQueueManager_name</i>	Name des QueueManagers (nur für lokale Tests auf Entwicklungssystemen notwendig, sonst leer)
<i>MQQueue_name</i>	Name der Queue Mögliche Werte: Beliebige gültige Zeichenkette
<i>MQQueue_dynamicQueue</i> <i>Name</i>	Name der <i>Session Queue</i> für diese Nachricht (Sollte zukünftig nicht mehr verwendet werden) Mögliche Werte: Beliebige gültige Zeichenkette
<i>MQQueue_alternateUserId</i>	Alternative User-ID um den Zugriff auf die Warteschlange zu prüfen, wenn diese geöffnet ist; dieser Parameter kann nicht gesetzt werden, während das Objekt geöffnet ist
<i>MQQueue_openOptions_</i> <i>MQC_MQOO_OUTPUT</i>	(Muss bei schreibendem Zugriff gesetzt werden)
<i>MQQueue_openOptions_</i> <i>MQC_MQOO_FAIL_IF_</i> <i>QUIESCING</i>	Erlaubt dem MQ-Manager, den Listener zu trennen (z.B. beim Herunterfahren)
<i>MQQueue_openOptions_</i> <i>MQC_MQDL_SUPPORTED</i>	MQ-spezifischer Parameter
<i>MQQueue_openOptions_</i> <i>MQC_MQOO_ALTERNATE_</i> <i>USER_AUTHORITY</i>	MQ-spezifischer Parameter
<i>MQQueue_openOptions_</i> <i>MQC_MQOO_BIND_AS_QDEF</i>	Queue-Einstellungen zur Persistenz etc. übernehmen (sollte immer gesetzt werden) Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: Queue-Einstellungen übernehmen • <i>nein</i>: Keine Queue-Einstellungen übernehmen (Standard)
<i>MQQueue_openOptions_</i> <i>MQC_MQOO_BIND_NOT_</i> <i>FIXED</i>	MQ-spezifischer Parameter
<i>MQQueue_openOptions_</i> <i>MQC_MQOO_BIND_ON_OPEN</i>	MQ-spezifischer Parameter

<i>MQQueue_openOptions_ MQC_MQOO_INPUT_AS_Q_DEF</i>	<p>Einstellungen der Queue zur Persistenz etc. übernehmen (Muss beim Lesen immer gesetzt werden)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Queue-Einstellungen übernehmen • <i>nein</i>: Keine Queue-Einstellungen übernehmen (Standard)
<i>MQQueue_openOptions_ MQC_MQOO_INPUT_SHARED</i>	MQ-spezifischer Parameter
<i>MQQueue_openOptions_ MQC_MQOO_INPUT_EXCLUSIVE</i>	MQ-spezifischer Parameter
<i>MQQueue_openOptions_ MQC_MQOO_INQUIRE</i>	MQ-spezifischer Parameter
<i>MQQueue_openOptions_ MQC_MQOO_PASS_ALL_CONTEXT</i>	MQ-spezifischer Parameter
<i>MQQueue_openOptions_ MQC_MQOO_PASS_IDENTITY_CONTEXT</i>	MQ-spezifischer Parameter
<i>MQQueue_openOptions_ MQC_MQOO_SAVE_ALL_CONTEXT</i>	MQ-spezifischer Parameter
<i>MQQueue_openOptions_ MQC_MQOO_SET</i>	MQ-spezifischer Parameter
<i>MQQueue_openOptions_ MQC_MQOO_SET_ALL_CONTEXT</i>	MQ-spezifischer Parameter
<i>MQQueue_openOptions_ MQC_MQOO_SET_IDENTITY_CONTEXT</i>	MQ-spezifischer Parameter
<i>MQPutMessageOptions_ options_MQC_MQPMO_NEW_MSG_ID</i>	MQ-spezifischer Parameter
<i>MQPutMessageOptions_ options_MQC_MQPMO_SYNCPOINT</i>	<p>Syncpoint beim Schreiben von Nachrichten verwenden (erforderlich für die Operationen Commit und Rollback)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Syncpoint beim Schreiben verwenden • <i>nein</i>: Keinen Syncpoint beim Schreiben verwenden (Standard)
<i>MQQueue_closeOptions_ MQC_MQCOO_DELETE</i>	MQ-spezifischer Parameter
<i>MQQueue_closeOptions_ MQC_MQCOO_DELETE_PURGE</i>	MQ-spezifischer Parameter


<i>MQMessage_format</i>	<p>Nachrichtenformat</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>MQC.MQFMT_STRING</i>: Nachricht besteht komplett aus Zeichen • <i>MQC.MQFMT_ADMIN</i>: Befehl-Server-Anfrage/Antwort • <i>MQC.MQFMT_COMMAND1</i>: 1 Befehl-Server-Antwortnachricht vom Typ 1 • <i>MQC.MQFMT_COMMAND2</i>: 2 Befehl-Server-Antwortnachricht vom Typ2 • <i>MQC.MQFMT_DEAD_LETTER_HEADER</i>: Dead-letter Header • <i>MQC.MQFMT_EVENT</i>: Event • <i>MQC.MQFMT_NONE</i>: Kein Formatname, es findet keine Konvertierung statt (d. h. die Bytes werden original geliefert); wenn zudem Parameter toXML aktiviert ist, dann wird die Nachricht Base64-kodiert • <i>MQC.MQFMT_PCF</i>: Benutzerdefinierte Nachricht im <i>Programmable Command Format</i> • <i>MQC.MQFMT_TRIGGER</i>: Trigger-Nachricht • <i>MQC.MQFMT_XMIT_Q_HEADER</i>: Transmission Queue Header • <i>(kein Wert)</i>: Nicht konvertieren (entspricht <i>MQC.MQFMT_NONE</i>)(Standard)
<i>MQMessage_applicationIdData</i>	MQ-spezifischer Parameter
<i>MQMessage_applicationOriginData</i>	MQ-spezifischer Parameter
<i>MQMessage_backoutCount</i>	MQ-spezifischer Parameter
<i>MQMessage_messageFlags</i> – <i>MQC_MQMT_REQUEST</i>	MQ-spezifischer Parameter
<i>MQMessage_replyToQueueName</i>	Name der ReplyToQueue für diese Nachricht
<i>MQMessage_replyToQueueManagerName</i>	<p>Name des ReplyToQueueManagers für diese Nachricht</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>MQMessage_characterSet</i>	<p>Charset (CCSID) für die Nachricht; beim Auslesen: das erwartete Charset; Beim Schreiben: das Charset, das die Nachricht bekommt</p> <p>Mögliche Werte: Beliebige CCSID (z.B. <i>1208</i>)</p>

<i>MQMessage_correlationId</i>	CorrelationID mit 24 Byte Länge (diese ist ebenso wie die zurückgegebene MessageID Base64-kodiert) Mögliche Werte: Beliebige Zeichenkette mit 32 Zeichen, die eine Base64-kodierte CorrelationID mit 24 Byte Länge enthält
<i>MQMessage_encoding</i>	Zeichenkodierung der Nachricht
<i>MQMessage_expiry</i>	Zeit in Zehntelsekunden, nach der die Nachricht ungültig wird und aus der Queue gelöscht wird Mögliche Werte: Beliebige ganze Zahl
<i>MQMessage_feedback</i>	MQ-spezifischer Parameter
<i>MQMessage_groupId</i>	MQ-spezifischer Parameter
<i>MQMessage_messageId</i>	MQ-spezifischer Parameter
<i>MQMessage_messageSequenceNumber</i>	MQ-spezifischer Parameter
<i>MQMessage_messageType</i>	MQ-spezifischer Parameter
<i>MQMessage_offset</i>	MQ-spezifischer Parameter
<i>MQMessage_originalLength</i>	MQ-spezifischer Parameter
<i>MQMessage_persistence</i>	MQ-spezifischer Parameter
<i>MQMessage_priority</i>	MQ-spezifischer Parameter
<i>MQMessage_putApplicationType</i>	MQ-spezifischer Parameter
<i>MQMessage_report</i>	MQ-spezifischer Parameter
<i>MQMessage_userId</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_options_MQC_MQGM0_SYNCPOINT</i>	Syncpoint beim Empfangen von Nachrichten verwenden (erforderlich für die Operationen Commit und Rollback) Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: Syncpoint beim Empfangen verwenden • <i>nein</i>: Keinen Syncpoint beim Empfangen verwenden (Standard)
<i>MQGetMessageOptions_options_MQC_MQGM0_WAIT</i>	Beim Auslesen warten, wenn sich keine Nachricht in der Queue befindet

<i>MQGetMessageOptions_options_MQC_MQGM0_NO_WAIT</i>	<p>Beim Auslesen nicht warten, wenn sich keine Nachricht in der Queue befindet, und sofort abbrechen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Nicht warten und sofort abbrechen • <i>nein</i>: Warten (Standard)
<i>MQGetMessageOptions_options_MQC_MQGM0_ACCEPT_TRUNCATED_MSG</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_options_MQC_MQGM0_BROWSE_FIRST</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_options_MQC_MQGM0_BROWSE_MSG_UNDER_CURSOR</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_options_MQC_MQGM0_BROWSE_NEXT</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_options_MQC_MQGM0_CONVERT</i>	<p>Automatische Konvertierung in MQ aktivieren</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Konvertierung aktivieren • <i>nein</i>: Keine Konvertierung (Standard)
<i>MQGetMessageOptions_options_MQC_MQGM0_FAIL_IF QUIESCING</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_options_MQC_MQGM0_LOCK</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_options_MQC_MQGM0_MSG_UNDER_CURSOR</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_options_MQC_MQGM0_NONE</i>	<p>Alle anderen <i>GetOptions</i> ignorieren</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Alle anderen <i>GetOptions</i> ignorieren • <i>nein</i>: Andere <i>GetOptions</i> nicht ignorieren (Standard)
<i>MQGetMessageOptions_options_MQC_MQGM0_MARK_SKIP_BACKOUT</i>	MQ-spezifischer Parameter

<i>MQGetMessageOptions_options_MQC_MQGM0_NO_SYNCPOINT</i>	<p>Syncpoint deaktivieren</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Syncpoint deaktivieren (damit ist keine Transaktion möglich) • <i>nein</i>: Syncpoint nicht deaktivieren (Standard)
<i>MQGetMessageOptions_options_MQC_MQGM0_SYNCPOINT_IF_PERSISTENT</i>	<p>Syncpoint nutzen, wenn die Nachricht als persistent gekennzeichnet ist</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Syncpoint bei persistenten Nachrichten nutzen • <i>nein</i>: Keinen Syncpoint bei persistenten Nachrichten nutzen (Standard)
<i>MQGetMessageOptions_options_MQC_MQGM0_UNLOCK</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_options_MQC_MQGM0_ALL_MSGS_AVAILABLE</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_options_MQC_MQGM0_ALL_SEGMENTS_AVAILABLE</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_options_MQC_MQGM0_COMPLETE_MSG</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_options_MQC_MQGM0_LOGICAL_ORDER</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_matchOptions_MQC_MQGM0_MATCH_CORREL_ID</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_matchOptions_MQC_MQGM0_MATCH_GROUP_ID</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_matchOptions_MQC_MQGM0_MATCH_MSG_ID</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_matchOptions_MQC_MQGM0_MATCH_MSG_SEQ_NUMBER</i>	MQ-spezifischer Parameter
<i>MQGetMessageOptions_matchOptions_MQC_MQGM0_NONE</i>	MQ-spezifischer Parameter

<i>MQGetMessageOptions_waitInterval</i>	Zeit in Millisekunden, die beim Auslesen gewartet wird, wenn sich keine Nachrichten in der Queue befinden (siehe Parameter <i>MQGetMessageOptions_options_MQC_MQGM0_WAIT</i>). Der Nachrichtenempfang wird nicht abgebrochen, wenn er während des Warte-Intervalls begonnen wurde. Mögliche Werte: Beliebige positive ganze Zahl
<i>MQGetMessageOptions_resolvedQueueName</i>	MQ-spezifischer Parameter

 Weitere Informationen zu den *MQ Series* spezifischen Parametern erhalten Sie von dessen Hersteller *IBM* in der englischsprachigen Dokumentation *WebSphere MQ – Using Java* In Kapitel 9 *The WebSphere MQ base Java classes and interfaces*.

Operation: Receive

Mit Operation *Receive* lassen sich Nachrichten aus der Warteschlange in den technischen Prozess einlesen.

Input

Für Operation *Receive* erwartet der Adapter kein Input-Dokument.

Ausgabe

Für Operation *Receive* und wenn Eigenschaft *Operation* auf *Receive* gesetzt wurde, gibt der Adapter beispielsweise folgendes XML-Dokument aus:

```
<document>
<header>
  <CorrelationID>AAAAAAAAAAAAAAAAAAAA</CorrelationID>
  <MessageID>QU1RIFFNX3NwX2Nvb5lYxvAdUQgC9sH</MessageID>
  <MessageSequenceNr>1</MessageSequenceNr>
  <MessageFormat>MQSTR</MessageFormat>
  <ReplyToQueue/>
  <ReplyToQueueMngr>QM_sp_connect</ReplyToQueueMngr>
  <backoutCount>0</backoutCount>
</header>
<body>Message</body>
</document>
```

Operation: Count

Input

Für Operation *Count* erwartet der Adapter kein Input-Dokument.

Ausgabe

Für Operation *Count* gibt der Adapter ein XML-Dokument aus, das im Wurzel-Element *<Count>* die Anzahl der Nachrichten in der Warteschlange enthält.


```
<Count><!--Nachrichtenzahl--><Count>
```

6.15 Office

6.15.1 Microsoft Excel

6.15.1.1 Microsoft Excel Reader



Microsoft Excel Reader: Liest ein *Microsoft Excel*-Dokument (im *.xls*-Format *Excel 9-2003* oder im XML-basierten *OOXML*-Format *.xlsx*) ein und gibt dessen Inhalt als XML-Dokument aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>Read Spreadsheet</i> : Excel-Dokument lesen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.excel.ExcelReader</i> : Hauptklasse (Standard)
<i>Formatter</i>	Verwendeter Formatierer (hat Einfluss auf die XML-Ausgabe, siehe unten) Mögliche Werte: <ul style="list-style-type: none"> • <i>de.softproject.integration.adapter.excel.DefaultFormatter</i>: Jeden Zelleninhalt des Excel-Dokuments in einem Element <code><Cell/></code> ausgeben (Standard) • <i>de.softproject.integration.adapter.excel.NumberedCellFormatter</i>: Jeden Zelleninhalt der Excel-Datei in einem XML-Element mit Zellennummer ausgeben (z. B. <code><Cell_0/></code>) • <i>de.softproject.integration.adapter.excel.LabeledCellFormatter</i>: Jeden Zelleninhalt der Excel-Datei in einem XML-Element mit Zellenbuchstaben ausgeben (z. B. <code><Cell_A/></code>)

Ausgabe

Mit dem Standard-Formatierer

de.softproject.integration.adapter.excel.DefaultFormatter wird folgende XML-Struktur ausgegeben:

```
<Workbook>
  <Sheet name="Name des Tabellenblatts">
    <Row rowId="Zeilennummer">
      <Cell colId="Spalten-ID" type="Datentyp" format="Formatierung"
        formula="Formel">!--Zelleninhalt--></Cell>
    </Row>
  </Sheet>
</Workbook>
```

Mit dem Formatierer

`de.softproject.integration.adapter.excelWriter.NumberedCellFormatter` wird folgende XML-Struktur ausgegeben:

```
<Workbook>
  <Sheet name="Name des Tabellenblatts">
    <Row rowId="Zeilennummer">
      <Cell_0 colId="Zellen-ID" type="Datentyp" format="Formatierung"
        formula="Formel">!--Zelleninhalt--></Cell_0>
    </Row>
  </Sheet>
</Workbook>
```

Mit dem Formatierer `de.softproject.integration.adapter.excel.LabeledCellFormatter` wird folgende XML-Struktur ausgegeben:

```
<Workbook>
  <Sheet name="Name des Tabellenblatts">
    <Row rowId="Zeilennummer">
      <Cell_A colId="Zellen-ID" type="Datentyp" format="Formatierung"
        formula="Formel">!--Zelleninhalt--></Cell_A>
    </Row>
  </Sheet>
</Workbook>
```

6.15.1.2 Microsoft Excel Writer



Microsoft Excel Writer: Konvertiert ein XML-Dokument mit einer bestimmten Tabellenstruktur in eine *Microsoft Excel*-Datei (im `.xls`-Format *Excel 97-2003* oder im XML-basierten *OOXML*-Format `.xlsx`) oder füllt eine bestehende Exceldatei (im `.xls`-Format *Excel 97-2003* oder im XML-basierten *OOXML*-Format `.xlsx`) mit Daten aus einem XML-Dokument mit einer bestimmten Tabellenstruktur.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Create Spreadsheet: Excel-Dokument mit einem Tabellenblatt ausgeben, die den MIME-Typ <i>Application/Vnd.ms-excel</i> besitzt • Fill Template: Bestehende Excel-Vorlage, die in Parameter <i>TemplateURL</i> angegeben wird und beliebig viele Tabellenblätter besitzen kann, mit Daten füllen
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.excel.ExcelWriter</i>: Hauptklasse (Standard)</p>
<i>generateOOXML</i>	<p>Excel-Dateiformat (Microsoft Excel 97 – 2003 oder OOXML) bestimmen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: XML-basiertes Excel-Dokument (Office Open XML; .xlsx) erzeugen • <i>false</i>: Excel-Dokument im Format Microsoft Excel 97 – 2003 (.xls) erzeugen (Standard)
<i>TemplateUrl</i>	<p>URL zur Excel-Vorlage (wenn die Eigenschaft <i>Operation</i> auf <i>Fill Template</i> gesetzt ist). Je nach dem in Parameter <i>generateOOXML</i> gewählten Dateiformat, muss die angegebene Vorlage in demselben Format vorliegen; z.B. muss eine .xlsx-Vorlage verwendet werden, wenn der Parameter <i>generateOOXML</i> gesetzt wurde. Der Wert des Parameters <i>TemplateUrl</i> kann durch das Attribut <i>templateUrl=".."</i> im Input-XML überschrieben werden (siehe unten).</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • URL mit <i>file://</i>-Protokollangabe (z.B. <i>file:/// %USER_COLLECTION%/Project/Template.xls</i>) • URL mit <i>xstore://</i>-Protokollangabe (z.B. <i>xstore:// Project/Folder/Template.xlsx</i>)

<i>recalc</i>	<p>Formeln vor dem Speichern des Dokuments neu berechnen und Feldwerte aktualisieren</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Formeln neu berechnen • <i>false</i>: Werte von Feldern mit Formeln unverändert lassen (Standard)
---------------	--

Input

Je nach gewählter Operation, erwartet der Adapter eine spezifische Input-Struktur:

• Operation Create Spreadsheet:

Für diese Operation erwartet der Adapter folgende XML-Struktur, die dem Aufbau von (X)HTML-Tabellen entspricht. Hierbei erzeugt das Element `<table>` ein Tabellenblatt, jedes Element `<tr>` eine neue Tabellenzeile und jedes Element `<td>` eine neue Zelle oder `<th>` eine Kopfzeile. Wenn mehrere Tabellenblätter erzeugt werden sollen, müssen die `<table>`-Elemente von einem Wurzel-Element mit beliebigem Namen umschlossen werden.

i Die Formatierung der einzelnen Zellen lässt sich über verschiedene Attribute steuern. Diese werden im Abschnitt **Mögliche Attribute** erläutert.


Beispiel-Input für die Operation Create Spreadsheet

```
<root>
  <table name = "Name of the first spreadsheet">
    <tr autofit="true" height="10">
      <td forceString = "true" fontHeight = "12" wrapText="true" width="20">
<!-- Cell content -> </ td>
    </tr>
  </table>
  <table name = "Name of the second spreadsheet">
    <tr>
      <td fontName = "Font name" formatNumber = "Format type" wrapText="true"
autofit="true"> <!-- Cell content -> </td>
    </tr>
  </table>
</root>
```

i Mit dem Attribut `forceString="true"` wird der Inhalt nicht als Zahl oder Datum interpretiert, sondern direkt in das Excel-Dokument reingeschrieben.
Beispiel: `008.07300` wird ohne das Attribut `forceString="true"` als Zahl `8.073` in das Excel-Dokument geschrieben, mit dem Attribut `forceString="true"` wird es als Zeichenkette `008.07300` in das Excel-Dokument geschrieben.

• Operation Fill Template:

Für diese Operation erwartet der Adapter folgende XML-Struktur. In optionalen Attribut `templateUrl=".."` können Sie hierbei eine gültige URL zu einer Vorlagendatei angeben, z. B. `templateUrl="file:///C:/Template.xlsx"`. Diese Angabe überschreibt den Wert des Adapter-Parameters `TemplateURL`. Jedes Element `<cell>` entspricht einer Zelle der Tabelle. Für Zeilen- und ggf. Spaltennummern beginnt die Zählung bei 0.


 Die Formatierung der einzelnen Zellen lässt sich über verschiedene Attribute steuern. Diese werden im Abschnitt **Mögliche Attribute** erläutert.

Beispiel-Input für Operation FillTemplat

```
<TemplateFill templateUrl="URL to template file">
  <cell sheet = "Name of the spreadsheet"
        row = "Row number"
        col = "Column ID "
        value = "Cell content"
        formatNumber = "Format number"
        wrapText="true"/>
</TemplateFill>
```

Mögliche Attribute

Die Formatierung der einzelnen Zellen lässt sich über folgende Attribute steuern:

Attribut	Beschreibung
formatNumber	<p>Zahlenformat</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • number1: Zahl 1 • number2: Zahl 2 • number3: Zahl 3 • number4: Zahl 4 • currency1: Währung 1 • currency2: Währung 2 • currency3: Währung 3 • currency4: Währung 4 • accounting: Buchhaltung • date: Datum • time: Uhrzeit • percentage: Prozent • fraction: Bruch • scientific: Wissenschaft • text: Text • postalCode: Postleitzahl • telephone: Telefonnummer <p> Bei Bedarf lassen sich auch benutzerdefinierte Zahlenformate hinterlegen</p>

fontName	Schriftart, z.B. Calibri
horizontalAlignment	Horizontale Ausrichtung des Textes innerhalb der Zelle Mögliche Werte: <ul style="list-style-type: none"> • CENTER • CENTER_SELECTION • DISTRIBUTED • FILL • GENERAL • JUSTIFY • LEFT • RIGHT
verticalAlignment	Vertikale Ausrichtung des Textes innerhalb der Zelle Mögliche Werte: <ul style="list-style-type: none"> • CENTER • BOTTOM • TOP • DISTRIBUTED • JUSTIFY
fontHeight	Schriftgrad
fontColor	Schriftfarbe Mögliche Werte: <ul style="list-style-type: none"> • CSS-Farangaben, z.B. Red, Blue, Green etc. • RGB-Farbcode, z.B. 240, 80, 0 • HEX-Farbcode, z.B. #124361
bold	Fette Schrift Mögliche Werte: <ul style="list-style-type: none"> • true: fette Schrift • false: normale Schrift
italic	Kursive Schrift Mögliche Werte: <ul style="list-style-type: none"> • true: kursive Schrift • false: normale Schrift
underline	Art des Unterstrichs Mögliche Werte: <ul style="list-style-type: none"> • NONE • DOUBLE • DOUBLE_ACCOUNTING • SINGLE • SINGLE_ACCOUNTING

cellColor	Füllfarbe der Zelle Mögliche Werte: <ul style="list-style-type: none"> • CSS-Farangaben, z.B. Red, Blue, Green etc. • RGB-Farbcode, z.B. 240, 80, 0 • HEX-Farbcode, z.B. #124361
border borderLeft borderRight borderBottom	Art des Zellrahmens Mögliche Werte: <ul style="list-style-type: none"> • DASH_DOT • DASH_DOT_DOT • DASHED • DOTTED • DOUBLE • HAIR • MEDIUM • MEDIUM_DASH_DOT • MEDIUM_DOT_DOT • MEDIUM_DASHED • SLANTED_DASH_DOT • THICK • THIN • NONE
borderColor borderColorLeft borderColorRight borderColorBottom borderColorTop	Farbe des Zellrahmens Mögliche Werte: <ul style="list-style-type: none"> • Farbname, z.B. Red, Blue, Green etc. • RGB-Farbcode, z.B. 240, 80, 0 • HEX-Farbcode, z.B. #124361
orientation	Text-Ausrichtung Mögliche Werte: <ul style="list-style-type: none"> • Zahl zwischen 90 und -90 • HORIZONTAL • VERTICAL
wrapText	<i>Optionales Attribut:</i> Automatischen Zeilenumbruch in Zellen erlauben Mögliche Werte: <ul style="list-style-type: none"> • true: Zeilenumbruch aktivieren • false: Zeilenumbruch deaktivieren

outofit *Optionales Attribut:* Spaltenbreite automatisch anpassen


Mögliche Werte:

- true: Automatische Spaltenbreite aktivieren
- false: Automatische Spaltenbreite deaktivieren

⚠ Setzen des Attributs

Bitte beachten Sie:

- Für die Operation Create Spreadsheet muss das Attribut innerhalb der ersten Zeile im Element <td> oder <th> gesetzt werden.
- Für die Operation Fill Template muss zunächst ein neues Element columnProperties eingefügt werden, in dem die Konfiguration der einzelnen Spalten – jeweils in einem Knoten column – vorgenommen wird.

columnProperties	<p>Gruppirt die Spalten eines Blattes.</p> <p>Mögliche Werte: sheet: Name des Excel-Blattes für die die Konfiguration vorgenommen wird.</p>
column	<p>Spalte innerhalb eines Blattes</p> <p>Mögliche Werte: id: Index der Spalte.  Der erste Index ist 0. Wird ein negativer Wert angegeben, wird eine Fehlermeldung ausgegeben.</p>

```
<columnProperties sheet="Week">
  <column id="1" autofit="true" width="256" />
  <column id="11" width="500" />
  <column id="9" width="255" />
  <column id="7" width="22" />
  <column id="22" autofit="true" />
</columnProperties>
```


width *Optionales Attribut:* Feste Spaltenbreite definieren

 Ist das Attribut `autofit` gesetzt, wird das Attribut `width` ignoriert.


Mögliche Werte: Zahl zwischen 0 und 255. Ist die angegebene Zahl größer 255, wird 255 als maximale Spaltenbreite gesetzt.

Setzen des Attributs

Bitte beachten Sie:


- Für die Operation `Create Spreadsheet` muss das Attribut innerhalb der ersten Zeile im Element `<td>` oder `<th>` gesetzt werden.
- Für die Operation `Fill Template` muss zunächst ein neues Element `columnProperties` eingefügt werden, in dem die Konfiguration der einzelnen Spalten – jeweils in einem Knoten `column` – vorgenommen wird.

<code>columnProperties</code>	Gruppieren die Spalten eines Blattes. Mögliche Werte: <code>sheet</code> : Name des Excel-Blattes für die die Konfiguration vorgenommen wird.
-------------------------------	---

<code>column</code>	Spalte innerhalb eines Blattes Mögliche Werte: <code>id</code> : Index der Spalte.  Der erste Index ist 0. Wird ein negativer Wert angegeben, wird eine Fehlermeldung ausgegeben.
---------------------	---

```
<columnProperties sheet="Week">
  <column id="1" autofit="true" width="256" />
  <column id="11" width="500" />
  <column id="9" width="255" />
  <column id="7" width="22" />
  <column id="22" autofit="true" />
</columnProperties>
```

height *Optionales Attribut:* Zeilenhöhe definieren


-  Excel wendet die automatische Anpassung der Höhe mit dem Attribut `autofit` nur an, wenn zuvor eine feste Zeilenhöhe über das Attribut `height` definiert wurde. Ist das Attribut `autofit` für die Zeile gesetzt, wird das Attribut `height` ignoriert. Zusätzlich wird die automatische Zeilenhöhe auch über das Attribut `wrapText` geregelt.

Mögliche Werte: Zahl zwischen 0 und 409. Ist die angegebene Zahl größer 409, wird 409 als maximale Zeilenhöhe gesetzt.

Setzen des Attributs

Bitte beachten Sie:

- Für die Operation `Create Spreadsheet` muss das Attribut innerhalb des Element `<tr>` gesetzt werden.
- Für die Operation `Fill Template` muss zunächst ein neues Element `rowProperties` eingefügt werden, in dem die Konfiguration der einzelnen Zeilen – jeweils in einem Knoten `row` – vorgenommen wird.

<code>rowProperties</code>	<p>Gruppiert die Zeilen eines Blattes.</p> <p>Mögliche Werte: <code>sheet</code>: Name des Excel-Blattes für die die Konfiguration vorgenommen wird.</p>
<code>row</code>	<p>Zeile innerhalb eines Blattes</p> <p>Mögliche Werte: <code>id</code>: Index der Zeile.</p> <p> Der erste Index ist 0. Wird ein negativer Wert angegeben, wird eine Fehlermeldung ausgegeben.</p>

```
<rowProperties sheet="Week">
  <row id="7" height="50"/>
  <row id="6" height="409"/>
  <row id="8" autofit="true" height="250"/>
</rowProperties>
```

6.15.2 Microsoft Word Properties



Microsoft Word Properties: Liest Metadaten (Dokumenteigenschaften wie Ersteller, letzte Speicherung) eines Microsoft Word-Dokuments (.doc und .docx) und gibt diese in einem UTF-8-kodierten XML-Dokument aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt
	Mögliche Werte: <code>Get</code> : Dokumenteigenschaften aus einem MS-Word-Dokument lesen

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.wordheaderadapter.wordHeaderAdapter:</i> Hauptklasse (Standard)
----------------	---

Statuswerte

<i>1(successful)</i>	Der Funktions-Adapter wurde erfolgreich ausgeführt und die ausgelesenen Dokumenteigenschaften wurden als XML-Dokument ausgegeben
<i>0</i>	Das verarbeitete Dokument hat ein falsches Dateiformat
<i>-1(error)</i>	Das verarbeitete Dokument ist keine gültige .doc bzw .docx-Datei oder ein anderer Fehler ist aufgetreten (siehe Server-Protokoll).

Input

Der Adapter erwartet als Input ein Microsoft Word-Dokument mit der Endung .doc oder .docx.

Ausgabe

Der Adapter gibt ein XML-Dokument mit folgender Struktur aus:

```

<WordHeader>
  <Properties>
    <Author>Max Mustermann</Author>
    <Template>Normal.dotm</Template>
    <LastSavedBy>Max Mustermann</LastSavedBy>
    <LastPrinted>2017-12-07T09:24:00Z</LastPrinted>
    <ModifiedOn>2017-12-07T08:24:00Z</ModifiedOn>
    <Revision>3</Revision>
    <EditTime>60</EditTime> <!-- one hour edited in word -->
    <CreatedOn>2017-12-07T08:22:00Z</CreatedOn>
    <Title>Titelproperty</Title>
    <Subject>Subjectproperty</Subject>
    <Keywords>Keyword</Keywords>
    <Comments>Kommentar</Comments>
    <Company>Max Mustermann GmbH</Company>
  </Properties>
  <Statistics>
    <Pages>1</Pages>
    <Words>6</Words>
    <Characters>40</Characters>
    <Application>Microsoft Office Word</Application>
    <Lines>1</Lines>
    <Paragraphs>1</Paragraphs>
    <CharactersWithSpaces>45</CharactersWithSpaces>
  </Statistics>
  <UserProperties />
</WordHeader>

```

Erläuterung: Das Element `UserProperties` ist nur bei Dateien mit der Endung `.doc` mit Daten befüllt.

Änderungen am Datumsformat

Bitte beachten Sie, dass sich bei der Ausgabe das Datumsformat geändert hat. Das zuvor deutsche Format `dd.mm.yyyy hh:mm:ss` wird nun nach ISO-8601 in der Zeitzone UTC formatiert `yyyy-mm-ddThh:mm:ssZ`.

6.15.3 Microsoft Word to Text Converter



Microsoft Word to Text Converter: Liest ein *Microsoft Word*-Dokument (`.doc` im Format *Office 97-2003*) ein, extrahiert dessen Textinhalte und gibt ein Textdokument aus. Bei der Konvertierung werden alle Word-spezifischen Formatierungsparameter entfernt, so dass der reine Inhalt zur Weiterverarbeitung zur Verfügung steht.

Eigenschaften (Properties)


<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt
	Mögliche Werte: <i>Convert</i> : Text aus einem MS-Word-Dokument extrahieren

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <code>de.softproject.integration.adapter.Word2XML</code> : Hauptklasse (Standard)
----------------	---

6.15.4 Working Time Calculator

Working Time Calculator: Berechnet die Arbeitszeit.

 Der *Working Time Calculator* ersetzt mit Version 6.0 der X4 Suite die veralteten Arbeitszeit-Berechnungsfunktionen in der Java-Klasse `de.softproject.xml.WorkingTimeCalculator`, welche zur Verwendung innerhalb von XSL-Mappings bereitstellt wurden.

Eigenschaften (Properties)

Operationen	Vom Adapter ausgeführte Operation Mögliche Werte: <ul style="list-style-type: none"> • <code>AddWorkingTimeMinutes</code>: Addiert zu einem Start-Datum eine bestimmte Anzahl an Arbeitsminuten und gibt das End-Datum und die Uhrzeit aus • <code>GetWorkingTimeMinutes</code>: Berechnet die Arbeitszeit zwischen einem Start-Datum und einem End-Datum und gibt die Arbeitsminuten aus • <code>GetEaster</code>: Berechnet den Ostersonntag anhand einer Jahreszahl
-------------	---

Parameter

Adapter	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <code>de.softproject.x4.adapter.workingTimeCalculator</code> : Hauptklasse (Standard)
---------	---


Statuswerte


1	Die Operation wurde erfolgreich ausgeführt und in der Ausgabe befindet sich ein XML-Dokument mit dem Resultat zur jeweiligen Operation
-1	Es kam zu einem Fehler während der Ausführung: <ul style="list-style-type: none"> • Das erforderliche Datums-Format (yyyy-MM-dd bzw. yyyy-MM-dd hh:mm) wurde nicht eingehalten • Die Wochentage wurden nicht richtig eingetragen. Erforderlich ist ein korrekter deutscher oder englischer Wochentag. • Es ist kein Input vorhanden

Input

Der Adapter erwartet eine vordefinierte Input-XML-Struktur für die verschiedenen Operationen

- Feiertage bzw. Urlaubstage (Arbeitsfreie Tage) werden im Datumsformat yyyy-MM-dd innerhalb beliebig vieler <Holiday>-Elemente definiert. Diese werden als Kind-Elemente von <Holidays> angelegt.
- Arbeitstage (Wochentage) werden innerhalb von <WorkingDay>-Elementen definiert. Diese werden als Kind-Elemente von <WorkingDays> angelegt. Der Wert von <WorkingDay> ist der jeweilige Wochentag auf deutsch oder auf englisch, z.B. *Montag* oder *Friday*. Dabei ist die Groß- und Kleinschreibung egal.
- Spezielle Tage werden im Datumsformat yyyy-MM-dd hh:mm-hh:mm innerhalb beliebig vieler <SpecialDay>-Elemente definiert. Diese werden als Kind-Elemente von <SpecialDays> angelegt.

 **SpecialDays** gelten als spezieller Arbeitstag unabhängig davon, ob es sich um einen normalen Arbeitstag oder arbeitsfreien Tag handelt. Als Arbeitszeit zählt nur die Zeit, die innerhalb von <SpecialDay> definiert wurde.

 Die Arbeitszeit-Berechnungsfunktionen `dateDiff` oder `dateDiffWorkingMinutes` der veralteten Java-Klasse `de.softproject.xml.WorkingTimeCalculator` wurden durch folgende Funktionen ersetzt:

- `dateDiffWorkingMinutes` kann durch die Operation `GetWorkingTimeMinutes` umgesetzt werden
- `dateDiff` lässt sich umsetzen, indem alle Wochentage als `WorkingDay` definiert und die Arbeitszeiten von 00:00 bis 23:59 eingestellt werden. Dabei dürfen weder `Holidays` noch `SpecialDays` definiert werden.

- Operation `AddWorkingTimeMinutes`:

Beispiel-Input für die Operation AddWorkingTimeMinutes

```
<WorkingTimeCalculator>
  <Holidays>
    <Holiday>2019-01-01</Holiday>
    <Holiday>2019-01-06</Holiday>
    <Holiday>2019-04-19</Holiday>
    <Holiday>2019-04-21</Holiday>
    <Holiday>2019-04-22</Holiday>
  </Holidays>
  <SpecialDays>
    <SpecialDay>2019-12-24 21:00-23:00</SpecialDay>
  </SpecialDays>
  <!--You can use either the German or the English word-->
  <WorkingDays>
    <WorkingDay>Monday</WorkingDay>
    <WorkingDay>Tuesday</WorkingDay>
    <WorkingDay>Wednesday</WorkingDay>
    <WorkingDay>Thursday</WorkingDay>
    <WorkingDay>Friday</WorkingDay>
  </WorkingDays>
  <setWorkingDayBegin>09:30</setWorkingDayBegin>
  <setWorkingDayEnd>17:30</setWorkingDayEnd>
  <StartDate>2019-12-23 10:00</StartDate> <!-- Format yyyy-MM-dd or yyyy-MM-
dd hh:mm -->
  <addMinutesToWorkingTime>500</addMinutesToWorkingTime>
</WorkingTimeCalculator>
```

- Operation GetWorkingTimeMinutes:

Beispiel-Input für die Operation GetWorkingTimeMinutes

```

<WorkingTimeCalculator>
  <Holidays>
    <Holiday>2019-01-01</Holiday>
    <Holiday>2019-01-06</Holiday>
    <Holiday>2019-04-19</Holiday>
    <Holiday>2019-04-21</Holiday>
    <Holiday>2019-04-22</Holiday>
  </Holidays>
  <SpecialDays>
    <SpecialDay>2019-12-24 21:00-23:00</SpecialDay>
  </SpecialDays>
  <!--You can use either the German or the English word-->
  <WorkingDays>
    <WorkingDay>Monday</WorkingDay>
    <WorkingDay>Tuesday</WorkingDay>
    <WorkingDay>Wednesday</WorkingDay>
    <WorkingDay>Thursday</WorkingDay>
    <WorkingDay>Friday</WorkingDay>
  </WorkingDays>
  <StartDate>2019-12-23 10:00</StartDate> <!-- Format yyyy-MM-dd or yyyy-MM-
dd hh:mm -->
  <EndDate>2020-01-01 10:00</EndDate> <!-- Format yyyy-MM-dd or yyyy-MM-dd
hh:mm -->
  <setWorkingDayBegin>09:30</setWorkingDayBegin>
  <setWorkingDayEnd>17:30</setWorkingDayEnd>
</WorkingTimeCalculator>

```

- Operation GetEaster:

Beispiel-Input für die Operation GetEaster

```

<WorkingTimeCalculator>
  <GetEaster>2019</GetEaster>
</WorkingTimeCalculator>

```

Ausgabe

Der Adapter gibt je nach Operation ein XML-Dokument aus.

- Operation AddWorkingTimeMinutes:

Beispiel-Output für die Operation AddWorkingTimeMinutes

```

<WorkingTimeCalculator>
  <addWorkingTimeMinutes>2019-12-24T21:50:00</addWorkingTimeMinutes>
</WorkingTimeCalculator>

```

- Operation GetWorkingTimeMinutes:

Beispiel-Output für die Operation GetWorkingTimeMinutes

```
<WorkingTimeCalculator>
  <getWorkingTimeMinutes>2010.0</getWorkingTimeMinutes>
</WorkingTimeCalculator>
```

- Operation GetEaster:

Beispiel-Output für die Operation GetEaster

```
<EasterHolidays>
  <EasterSunday>2019-04-21</EasterSunday>
</EasterHolidays>
```

6.16 PDF Processing

6.16.1 PDF Data Processing

6.16.1.1 PDF Barcode Adder



PDF Barcode Adder: Erzeugt Strichcodes und fügt diese einem bestehenden nicht geschützten PDF-Dokument hinzu. Die Strichcode-Grafiken können auf jeder PDF-Seite positioniert, skaliert und gedreht werden. Unterstützt werden die Strichcode-Typen *Code 39*, *Code 128*, *Codabar*, *EAN 13*, *Code 25-Interleaved 2 of 5* sowie *POSTNET*.

In einem adapterspezifisch aufgebauten Input-XML-Dokument werden die Daten der Strichcodes hinterlegt. Der Adapter gibt ein PDF-Dokument aus, das um die angegebenen Strichcodes ergänzt wurde.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>AddBarcodes</i> : Dem PDF-Dokument Strichcode(s) hinzufügen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.pdf.PDFBarcode</i> : Hauptklasse (Standard)
----------------	--

<i>ReadPartially</i>	<p>PDF-Dokument in mehreren Teilen in den Arbeitsspeicher laden (kann Performance-Verbesserungen bringen)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: In mehreren Teilen nacheinander in den Arbeitsspeicher laden • <i>nein</i>: PDF-Dokument In einem Stück in den Arbeitsspeicher laden (Standard)
<i>ToXML</i>	<p>PDF-Dokument Base64-kodiert in einem XML-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: XML-Dokument ausgeben • <i>nein</i>: PDF-Dokument als Binärdatenstrom ausgeben (Standard)

Input

Dieser Funktions-Adapter erwartet eine adapterspezifische Input-XML-Struktur, in der sämtliche Daten jeder Strichcode-Grafik hinterlegt sind (siehe unten). Wenn ein PDF aus dem Repository als Quelldokument verwendet werden soll, in Element `<PDF>` im Attribut `url` dessen `xstore`-URL angeben. Alternativ, wenn das Quelldokument als Base64-kodierte Zeichenkette vorliegt, können Sie diese Zeichenkette als Inhalt des Elements `<PDF encoding="base64">` hinterlegen (siehe auskommentierte Zeile).

```
<pdfBarcode>
  <PDF url="xstore-URL des Input-PDF-Dokuments im Repository"/>
  <!-- <PDF encoding="base64">(Wenn Input-PDF Base64-Zeichenkette)</PDF> -->
  <Barcode height="Höhe" width="Breite"
    rad="Drehung (Bogenmaß)" degree="Drehung (Grad)"
    xPos="Abstand vom linken Seitenrand" yPos="Abstand vom unteren Seitenrand"
    type="Strichcode-Typ" page="Seite">Strichcode-Daten</Barcode>
</pdfBarcode>
```

Attribute im Input-XML-Dokument

Der *PDF Barcode Adder* erwartet eine adapterspezifische Input-XML-Struktur nach oben genanntem Muster. Folgende Attribute können gesetzt werden:

<i>url</i>	xstore-Pfadangabe zum Input-PDF-Dokument im Repository (z. B. <code>xstore://Project/Data/InputData.xml</code> ; erforderlich)
<i>height</i>	Höhe der Strichcode-Grafik in <i>cm</i> oder <i>pt</i> (erforderlich)

<i>width</i>	Breite der Strichcode-Grafik in <i>cm</i> oder <i>pt</i> (erforderlich)
<i>rad</i>	Drehung der Strichcode-Grafik im Bogenmaß
<i>degree</i>	Drehung der Strichcode-Grafik in Grad (erforderlich)
<i>xPos</i>	Abstand der Strichcode-Grafik vom linken Seitenrand in <i>cm</i> oder <i>pt</i> (erforderlich)
<i>yPos</i>	Abstand der Strichcode-Grafik vom unteren Seitenrand in <i>cm</i> oder <i>pt</i> (erforderlich)
<i>type</i>	Strichcode-Typ (erforderlich) Mögliche Werte: <ul style="list-style-type: none"> • <i>39</i>: Code 39-Format • <i>128</i>: Code 128-Format • <i>Codabar</i>: Codabar-Format • <i>EAN</i>: EAN 13-Format • <i>Inter25</i>: Code 25 – Interleaved 2 of 5-Format • <i>Postnet</i>: POSTNET-Format
<i>page</i>	Seitenzahl im PDF-Dokument, auf der die Strichcode-Grafik positioniert wird (erforderlich)

Beispiel

Folgendes Input-XML-Dokument enthält die Daten zum Erzeugen einer *EAN 13*-Strichcode-Grafik mit 3 cm Breite und 20° Neigung auf der ersten Seite eines PDF-Dokuments im Repository.

```
<pdfBarcode>
  <PDF url="xstore://Project/Folder/InputPDF.pdf"/>
  <!-- <PDF encoding="base64">base_64_encoded_PDF_document</PDF> -->
  <Barcode height="3cm" width="3cm" rad="" degree="20"
    xPos="120pt" yPos="1cm" type="EAN"
    page="1">1234567891234</Barcode>
</pdfBarcode>
```

6.16.1.2 PDF Form



PDF Form: Erstellt aus FDF/XFDF-Formulardaten ein PDF-Dokument mit oder ohne Formularfelder, extrahiert eine FDF-Datei aus einem PDF-Formulardokument, setzt Formularfeldwerte oder wandelt ein PDF-Formulardokument in ein flaches PDF-Dokument um. Ehemalige Bezeichnung bis Version 4.5.1: *PDF Form Processing Adapter*

- ❗ Die FDF/XFDF-Formulardaten werden in einem Text- bzw. XML-Dokument gespeichert. Dieses Formulardatendokument muss jeweils eine absolute Pfadangabe zu einem PDF-Formular-Dokument enthalten, das als Vorlage für das zu erstellende PDF-Dokument dient (nicht relevant für die Operationen `ExportFDF` und `Flatten`).

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>FlattenXFDF</i>: XFDF-Daten in ein PDF-Formulardokument schreiben (Formularfelder in Text umwandeln) • <i>FlattenFDF</i>: FDF-Daten in ein PDF-Formulardokument schreiben (Formularfelder in Text umwandeln) • <i>ExportFDF</i>: FDF-Daten aus einem PDF-Formulardokument extrahieren • <i>ImportXFDF</i>: XFDF-Daten in ein PDF-Formulardokument schreiben (Formularfelder erhalten) • <i>ImportFDF</i>: FDF-Daten in ein PDF-Formulardokument schreiben (Formularfelder erhalten) • <i>SetFields</i>: Formularfelderwerte setzen und ggf. im ausgegebenen PDF-Dokument die Formularfelder in Text umwandeln • <i>Flatten</i>: PDF-Formulardokument in ein flaches PDF-Dokument umwandeln (Formularfelder in Text umwandeln)
------------------	---

Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.pdf.PDFFormular</code>: Hauptklasse (Standard)</p>
PreserveExtendedAttributes	<p>Beim Verwenden der Operation <code>SetFields</code> die erweiterten Attribute im PDF-Dokument erhalten, wenn für ein Formularfeld im Input-XML über <code>flatten="false"</code> definiert wurde, dass Formularfelder editierbar bleiben sollen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Nach dem Befüllen der Formularfelder die erweiterten Attribute erhalten (Standard) • <i>nein</i>: Erweiterte Attribute für die Formularfelder im PDF-Dokument nicht erhalten (dies entspricht dem Verhalten in vorherigen Versionen dieses Adapters und ist nur bei Kompatibilitätsproblemen relevant)

Input

Für Operation `SetFields` erwartet der Adapter ein adapterspezifisches XML-Dokument, das den absoluten Pfad zum zugehörigen PDF-Formular in Element `<Document>` und die zu setzenden Formularfelddaten in beliebig vielen Elementen `<Field>` angibt. Mit Attribut `flatten="true"` werden im erzeugten PDF-Dokument die Formularfelder in flachen Text umgewandelt, mit `flatten="false"` bleiben die Formularfelder editierbar.

```
<Input>
  <Document flatten="true/false"><!--Absoluter Pfad zur PDF-Formulardatei--></
Document>
  <Field name="Feldname"><!--Feldwert--></Field>
  <!--...-->
</Input>
```

6.16.1.3 PDF Metadata



PDF Metadata: Liest Metadatenstrukturen aus PDF-Dokumenten aus oder schreibt Metadatenstrukturen in PDF-Dokumente.

Eigenschaften (Properties)

<i>Custom Parameter</i>	Benutzerdefinierte Parameter setzen; siehe <i>Handbuch X4 Designer – Benutzerdefinierte Parameter verwenden</i>
<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>AddMetadata</i>: Metadaten hinzufügen • <i>GetMetadata</i>: Metadaten auslesen • <i>SetMetadata</i>: Metadaten ersetzen

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.pdf.PDFMetadata</i> : Hauptklasse (Standard)
<i>ReadPartially</i>	PDF-Dokument in mehreren Teilen in den Arbeitsspeicher laden (kann Performance-Verbesserungen bringen) Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: In mehreren Teilen nacheinander in den Arbeitsspeicher laden • <i>nein</i>: PDF-Dokument(e) in einem Stück in den Arbeitsspeicher laden (Standard)

<i>ToXML</i>	<p>PDF-Dokument Base64-kodiert in einem XML-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: XML-Dokument ausgeben • <i>nein</i>: PDF-Dokument als Binärdatenstrom ausgeben (Standard)
--------------	---

Input

Dieser Funktions-Adapter erwartet eine adapterspezifische Input-XML-Struktur, die sämtliche Verarbeitungs-Informationen für das erzeugte PDF-Dokument enthält.

```
<pdfMetadata>
  <PDF url="xstore-URL eines Input-PDF-Dokuments im Repository" />

  <!-- <PDF encoding="base64"> base64-String (Wenn Input-PDF Base64-Zeichenkette)
  </PDF> -->

  <Metadata>
    <!-- beliebig viele Elemente, wobei ElementName=Name des Eintrags und
    ElementInhalt=Wert des Eintrags -->
    <ElementName><!--ElementInhalt--></ElementName>
  </Metadata>

</pdfMetadata>
```

Output

Je nach ausgewähltem Wert im Parameter *ToXML*, können folgende Output-Strukturen ausgegeben werden:

- *nein*: das PDF-Dokument wird als Binärdatenstrom ausgegeben
- *ja*: Ein XML-Dokument mit den Metadaten in folgender Struktur ausgeben:

Output beim Parameter ToXML: ja

```
<PDF encoding="base64"> base64-String </PDF>
```

6.16.1.4 PDF to Text Converter



PDF to Text Converter: Liest ein PDF-Dokument (auch passwortgeschützt) ein, extrahiert die Textinhalte eines bestimmten Seitenbereichs oder des gesamten Dokuments und gibt die extrahierten Zeichenketten in einem XML- oder Textdokument mit einer frei wählbaren Zeichenkodierung aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Extract</i> : Text aus dem Input-PDF-Dokument extrahieren
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.pdf.PDF2Text</i> : Hauptklasse (Standard)
<i>password</i>	Passwort (bei einem geschützten PDF-Dokument) Mögliche Werte: Beliebige Zeichenkette
<i>startPage</i>	Erste Seitenzahl, ab der die Texte extrahiert werden sollen Mögliche Werte: <ul style="list-style-type: none"> • Beliebige positive ganze Zahl oder \emptyset • \emptyset: Ab der ersten Seite beginnen (Standard)
<i>endPage</i>	Letzte Seitenzahl, bis zu der die Textextraktion durchgeführt werden soll Mögliche Werte: <ul style="list-style-type: none"> • Beliebige ganze Zahl oder \emptyset • \emptyset: Bis zur letzten Seite Text extrahieren (Standard)
<i>encoding</i>	Zeichenkodierung des Ergebnisdokuments Mögliche Werte: Beliebige gültige Zeichenkodierung (z.B. <i>UTF-8</i>)
<i>force</i>	Auch auf ungültigen PDF-Seiten versuchen, Text zu extrahieren Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: Ungültige PDF-Seiten verarbeiten • <i>nein</i>: Ungültige PDF-Seiten ignorieren (Standard)
<i>toXML</i>	Textinhalte in einem XML-Dokument ausgeben Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: XML-Dokument ausgeben • <i>nein</i>: Textdokument ausgeben (Standard)

6.16.2 PDF Image Processing

6.16.2.1 PDF Image Adder



PDF Image Adder: Fügt einem bestehenden nicht geschützten PDF-Dokument eine oder mehrere Grafiken hinzu oder erstellt ein neues PDF-Dokument mit den angegebenen Grafiken. Diese können sowohl als URL als auch als Base64-kodierte Zeichenkette angegeben werden.

Unterstützte Grafikformate sind BMP, EPS, GIF, JPEG/JPG, PNG, TIFF und WMF. Die Grafiken können auf jeder PDF-Seite positioniert, skaliert und gedreht werden. In einem adapterspezifisch aufgebauten Input-XML-Dokument werden die Informationen zu den Grafiken hinterlegt. Der Adapter gibt entweder ein neues PDF-Dokument oder ein PDF-Dokument, das um die angegebenen Grafiken ergänzt wurde, aus.

Enthält das Input-Dokument kein PDF-Element, wird zunächst ein PDF-Dokument im DIN A4-Format mit der definierten Anzahl an Seiten angelegt, bevor die Grafiken hinzugefügt werden.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>AddImages</i> : Dem PDF-Dokument Grafiken hinzufügen
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.pdf.PDFImage</i> : Hauptklasse (Standard)
<i>ReadPartially</i>	PDF-Dokument in mehreren Teilen in den Arbeitsspeicher laden (kann Performance-Verbesserungen bringen) Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: In mehreren Teilen nacheinander in den Arbeitsspeicher laden • <i>nein</i>: PDF-Dokument in einem Stück in den Arbeitsspeicher laden (Standard)

<i>ToXML</i>	<p>PDF-Dokument Base64-kodiert in einem XML-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: XML-Dokument ausgeben • <i>nein</i>: PDF-Dokument als Binärdatenstrom ausgeben (Standard)
--------------	---

Input

Dieser Funktions-Adapter erwartet eine adapterspezifische Input-XML-Struktur, in der Informationen zu jeder Grafik hinterlegt sind. Hierzu zählen Breite und Höhe (in *cm* oder *pt*), Drehung, x- und y-Position (Abstände vom linken unteren Seitenrand in *cm* oder *pt*), der URL zur Grafik sowie die Seite, auf der die Grafik positioniert werden soll. Das Input-PDF-Dokument kann entweder als URL oder als Base64-kodierte Zeichenkette angegeben werden.

```
<pdfImage>
  <!-- <PDF url="xstore-URL des Input-PDF-Dokuments im Repository" (Wenn Input-PDF
als URL angegeben wird) /> -->
  <!-- <PDF encoding="base64"> Base64-Zeichenkette (Wenn Input-PDF Base64-
Zeichenkette)</PDF> -->
  <Image height="Höhe" width="Breite"
    rad="Drehung (Bogenmaß)" degree="Drehung (Grad)"
    xPos="Abstand vom linken Seitenrand" yPos="Abstand vom unteren Seitenrand"
    url="xstore-URL der Grafik im Repository" page="Seite im PDF"/>
</pdfImage>
```

Folgendes Input-XML-Dokument enthält Informationen für eine PNG-Bilddatei, die auf der ersten Seite eines bereits bestehenden PDF-Dokuments (Angabe als URL oder Base64-Zeichenkette) eingefügt werden soll.

```
<pdfImage>
  <!-- <PDF url="xstore://Project/Folder/InputPDF.pdf"/> -->
  <!-- <PDF encoding="base64">base_64_encoded_PDF_document</PDF> -->
  <Image height="90pt" width="60pt" rad="" degree=""
    xPos="400pt" yPos="100pt"
    url="xstore://Project/Folder/Image.png" page="1"/>
</pdfImage>
```

Folgendes Input-XML-Dokument enthält Informationen für drei Grafiken, die in einem neuen PDF-Dokument zusammengeführt werden sollen.

```
<?xml version="1.0" encoding="UTF-8"?>
<Root>
  <Image page="1" height="100pt" xPos="10pt" yPos="500pt" url="https://
www.softproject.de/assets/images/a/Logo-SoftProject_www-ee6b7fba.png"/>
  <Image page="3" height="100pt" xPos="10pt" yPos="500pt" url="https://
www.softproject.de/assets/images/a/Logo-SoftProject_www-ee6b7fba.png"/>
  <Image page="3" height="120pt" xPos="50pt" yPos="100pt" encoding="base64">

iVBORw0KGgoAAAANSUHEUgAAAEsAAAAWCAyAAACIXmHDAAXNSR0IArs4c6QAAARnQU1BAACxjwv8YQUAA
AAJcEhZcwAAEnQAABJ0Ad5mH3gAAAHDSURBVfH7ZSxSgNBEIZ9gjyZjS/
hY9jbijW9rWBlaWNpqVUqEQRBwSBRE4gxWf3X/
MfvuHvcB066HfjZy+z07Mx3k9t53huFqm6qsByqsByqsBz6B+v1YDesJo9hvVyEj7PDxj+/
PA0wrHp+erwf1vP3KDzrnu6nbPkwbvKq6d2L26uN99e4h9iU4a7Z+VHswRpiNCd/U/
AjDgzUTyUniw0ykABtcohF52CpUEzXl5DaQx1fT/d/4vGMnMhNH+vNNU5YtpYiWBATYoVSMNAE/
Mu7m0Fgpc5blcJaTV+idACiVwQL6TZZjh98MemBpysttzFsH72ZhcMTfrKYYFsdhUArwR+vuApa5EENL3d
EGyxrPYGXt6IPTpX7mUnWCLWpQfbhkW1hY9azKNq9n22DlGlcMX4zXcWw0DiS8HukSfhRT5L+A6xKYVEEo7V
sCwu/+S/5HF9nY6AsLCZAIJK3NRQhDDhZKtSld/UBq3kJbxM/
LAazAV6eAzIULJzXSW2mXXx9wIKQE+aClSoIYLpWD/
UByxr80G9NoUBtsKwRBM5aKMzjglWV4XlUIXlUIXlUIXlUIXlUIXVWaPwDd5omEQbL+NTAAAAAE1FTkSuQmC
C</Image>
</Root>
```

6.16.2.2 PDF Multipage TIFF Adder



PDF Multipage TIFF Adder: Fügt einem bestehenden nicht geschützten PDF-Dokument eine oder mehrere Grafiken aus mehrseitigen TIFF-Dateien (*Multipage Tagged Image File Format*) hinzu.

Die Grafiken können auf jeder PDF-Seite positioniert, skaliert und gedreht werden. In einem adapterspezifisch aufgebauten Input-XML-Dokument werden die Informationen zu den Grafiken hinterlegt. Der Adapter gibt ein PDF-Dokument aus, das um die angegebenen Grafiken ergänzt wurde, als Binärdatenstrom oder als XML-Dokument.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>AddImages</i> : Dem PDF-Dokument Grafiken aus mehrseitigen TIFF-Dateien hinzufügen
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.pdf.PDFTiffImage</i> : Hauptklasse (Standard)
----------------	---

<i>ReadPartially</i>	<p>PDF-Dokument in mehreren Teilen in den Arbeitsspeicher laden (kann Performance-Verbesserungen bringen)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: In mehreren Teilen nacheinander in den Arbeitsspeicher laden • <i>nein</i>: PDF-Dokument(e) In einem Stück in den Arbeitsspeicher laden (Standard)
<i>ToXML</i>	<p>PDF-Dokument Base64-kodiert in einem XML-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: XML-Dokument ausgeben • <i>nein</i>: PDF-Dokument als Binärdatenstrom ausgeben (Standard)

Input

Dieser Funktions-Adapter erwartet eine adapterspezifische Input-XML-Struktur, in der Informationen zu jeder Grafik hinterlegt sind. Hierzu zählen Breite und Höhe (in *cm* oder *pt*), Drehung, x- und y-Position (Abstände vom unteren linken Seitenrand in *cm* oder *pt*), die Seite im Multipage-TIFF, der URL zur Grafik sowie die Seite im PDF-Dokument, auf der die Grafik positioniert werden soll.

Jedes Element `<TIFF/>` referenziert eine TIFF-Datei und jedes Element `<image/>` platziert eine Abbildung im PDF. Wenn es sich bei einem Input-PDF-Dokument um eine Base64-kodierte Daten handelt, muss stattdessen der auskommentierte Code angepasst und einkommentiert werden.

```
<pdfTiffImage>
  <PDF url="xstore-URL des Input-PDF-Dokuments im Repository"/>
  <!-- <PDF encoding="base64">(Wenn Input-PDF Base64-Zeichenkette)</PDF> -->

  <TIFF url="xstore-URL der TIFF-Grafik im Repository">
    <image index="Seite im Multipage-TIFF" page="Seite im PDF"
      height="Höhe" width="Breite" rad="Drehung (Bogenmaß)"
      degree="Drehung (Grad)" xPos="Abstand vom linken Seitenrand"
      yPos="Abstand vom unteren Seitenrand"/>
  </TIFF>
</pdfTiffImage>
```

Beispiel

Folgendes Input-XML-Dokument enthält Informationen zu zwei Seiten einer mehrseitigen TIF-Grafik, die in ein neues PDF-Dokument eingefügt werden sollen.

```

<pdfTiffImage>
  <PDF url="xstore://Project/Folder/Input.pdf"/>
  <!-- <PDF encoding="base64">base_64_encoded_PDF_document</PDF> -->

  <TIFF url="xstore://Project/Folder/Image.tif">
    <image index="1" page="1" height="100pt" width="300pt" rad=""
      degree="" xPos="9cm" yPos="26cm"/>
    <image index="3" page="2" height="" width="100pt" rad=""
      degree="" xPos="2.6cm" yPos="740pt"/>
  </TIFF>
</pdfTiffImage>

```

6.16.2.3 PDF to Image Converter



PDF to Image Converter: Konvertiert und skaliert die erste Seite eines bestehenden, nicht geschützten PDF-Dokuments in ein Bildformat.

Folgende Formate können erzeugt werden: BMP, GIF, JPEG, JPEG 2000, PNG, PNM, RAW, TIFF und WBPM. Der Adapter gibt die Grafik im gewünschten Dateiformat als Binärdatenstrom aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>Transform</i> : PDF in ein Bildformat konvertieren
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.pdf.PDF2Image</i> : Hauptklasse (Standard)
<i>width</i>	Breite des ausgegebenen Bildes in Pixel (wird ignoriert, wenn Parameter <i>maxSize</i> gesetzt ist) Mögliche Werte: <ul style="list-style-type: none"> • Beliebige positive ganze Zahl, z.B. <i>800</i> • <i>-1</i> (in Kombination mit einer positiven Pixelzahl in Parameter <i>height</i>): beim Skalieren des Bilds das ursprüngliche Seitenverhältnis beibehalten; in Kombination mit dem Wert <i>-1</i> bei Parameter <i>height</i> werden die Originalbildabmessungen beibehalten

<i>height</i>	<p>Höhe des ausgegebenen Bildes in Pixel (wird ignoriert, wenn Parameter <code>maxSize</code> gesetzt ist)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• Beliebige positive ganze Zahl, z.B. 600• -1 (in Kombination mit einer positiven Pixelzahl in Parameter <code>width</code>): beim Skalieren des Bilds das ursprüngliche Seitenverhältnis beibehalten; in Kombination mit dem Wert -1 bei Parameter <code>width</code> werden die Originalbildabmessungen beibehalten
<i>page</i>	<p>Seite innerhalb des PDF-Dokuments, die als Bild ausgegeben werden soll</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• Beliebige ganze Zahl
<i>maxSize</i>	<p>Maximale Breite oder Höhe des ausgegebenen Bildes in Pixel (Parameter <code>width</code> und <code>height</code> werden ignoriert)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• Beliebige ganze Zahl• 0: Parameter <code>width</code> und <code>height</code> zur Skalierung verwenden (Standard)
<i>scaleAlgorithm</i>	<p>Verwendeter Skalierungs-Algorithmus</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <code>SCALE_DEFAULT</code>: Standard-Algorithmus verwenden• <code>SCALE_FAST</code>: Schnellen Algorithmus verwenden• <code>SCALE_SMOOTH</code>: Kantenglättungs-Algorithmus verwenden• <code>SCALE_REPLICATE</code>: Pixelwiederholungs-Algorithmus verwenden• <code>SCALE_AREA-AVERAGING</code>: Flächen-Durchschnittsberechnungs-Algorithmus verwenden

<i>outputFormat</i>	<p>Ausgegebenes Bildformat</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>BMP</i>: Windows-Bitmap-Grafik erzeugen • <i>GIF</i>: GIF-komprimiertes Bild erzeugen • <i>JPEG</i>: JPEG-komprimiertes Bild erzeugen • <i>JPEG 2000</i>: Bild im <i>JPEG 2000</i>-Format erzeugen • <i>PNG</i>: PNG-Bild erzeugen • <i>PNM</i>: Bild im <i>Portable Anymap</i>-Format erzeugen • <i>RAW</i>: Rohdaten-Bild erzeugen • <i>TIFF</i>: Bild im <i>Tagged Image File Format</i> erzeugen • <i>WBPM</i>: Bild im <i>Wireless Bitmap</i>-Format erzeugen
---------------------	--

6.16.2.4 PDF to TIFF Converter



PDF to TIFF Converter: Konvertiert ein PDF-Dokument in eine TIFF-Grafik (*Tagged Image File Format*) und gibt die Grafik als Binärdatenstrom aus. Mehrseitige PDF-Dokumente werden in eine mehrseitige TIFF-Grafik (*Multipage TIFF*) konvertiert.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Adapter durchführt</p> <p>Mögliche Werte: <i>Transformation</i>: PDF-Dokument in eine TIFF-Grafik konvertieren</p>
------------------	---

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.pdf2tiff.PDFToTiffAdapter</i>: Hauptklasse (Standard)</p>
<i>compression</i>	<p>Kompressionstyp</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Deflate (1)</i>: Per Deflate verlustfrei komprimieren • <i>JPEG (2)</i>: JPEG-Komprimierung (verlustbehaftet) verwenden • <i>None (3)</i>: Nicht komprimieren (Standard) • <i>Packbits (4)</i>: Per PackBits-Lauflängenkodierung verlustfrei komprimieren

<i>deflateLvl</i>	<p>Deflate-Level (wenn Kompressionstyp <i>Deflate</i> gewählt wurde und die <i>Deflate</i>-Komprimierung auf die gesamte TIFF-Datei angewendet werden soll)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl von 1–9 (komprimiert bei 1 am schnellsten, bei 9 am ausführlichsten) • 0: Keine <i>Deflate</i>-Komprimierung auf die gesamte TIFF-Grafik anwenden (Standard)
<i>quality</i>	<p>Bei JPEG-Kompression: Qualitätsstufe der (verlustbehafteten) JPEG-Kompression</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl von 0–100 (bei 0 geringste Qualität, bei 100 maximale Qualität) • 0: Geringste Qualität, d.h. maximale Kompression (Standard)
<i>resX</i>	<p>Horizontaler Bildschirm-/Druck-Auflösungswert</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl (z. B. 72) • 0: Keinen Auflösungswert setzen
<i>resY</i>	<p>Vertikaler Bildschirm-/Druck-Auflösungswert</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl (z. B. 72) • 0: Keinen Auflösungswert setzen
<i>resUnit</i>	<p>Einheit für die Bildschirm-/Druck-Auflösung</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>No absolute unit</i>: Keine absolute Einheit setzen • <i>Inch</i>: Zoll als Einheit verwenden (DPI bzw. PPI) • <i>Centimeter</i>: Zentimeter als Einheit verwenden

- ❗ Eine (optionale) CPU-Typ-/betriebssystemabhängige *JAI MediaLib*-Bibliothek wird nicht mitgeliefert. Der Adapter gibt deshalb folgende Fehlermeldung im Server-Protokoll aus, die für die Adapter-Ausführung jedoch nicht relevant ist:

```
ERROR [STDERR] Could not find mediaLib accelerator wrapper classes.  
Continuing in pure Java mode."
```

Um die *JAI MediaLib*-Bibliothek abzuschalten, folgende Java-VM-Option setzen (z. B. im Start-Skript):

```
@set JAVA_OPTS=%JAVA_OPTS% -Dcom.sun.media.jai.disableMediaLib=true
```

6.16.2.5 TIFF to PDF Converter



TIFF to PDF Converter: Erzeugt aus einer mehrseitigen TIFF-Datei (*Multipage Tagged Image File Format*) ein PDF-Dokument im Format 210 x 297 mm (DIN A4), wobei jede Seite in der TIFF-Datei einer Seite im PDF-Dokument entspricht. Die Grafiken können im PDF-Dokument positioniert und anhand der Breite der Grafik proportional skaliert werden.

⚠ Beachten Sie, dass .tif-Dateien mit Alphakanälen nicht unterstützt werden.

Eigenschaften

Operation	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: Convert: PDF-Dokument aus einer mehrseitigen TIFF-Datei erstellen
-----------	---

Parameter

Adapter	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.pdf.Tiff2Pdf</i> : Hauptklasse (Standard)
ToXML	PDF-Dokument Base64-kodiert in einem XML-Dokument ausgeben Mögliche Werte: <ul style="list-style-type: none"> ja: XML-Dokument ausgeben nein: PDF-Dokument als Binärdatenstrom ausgeben (Standard)
Header	Text, der als Überschrift im PDF auf jeder Seite steht Mögliche Werte: Beliebige Zeichenkette

xPos	Abstand vom linken Seitenrand in pt Mögliche Werte: Beliebige ganze Zahl
yPos	Abstand vom unteren Seitenrand in pt Mögliche Werte: Beliebige ganze Zahl
scale	Grafiken skalieren und in die PDF-Seite einpassen Mögliche Werte: <ul style="list-style-type: none"> • ja: Grafiken skalieren • nein: Grafiken nicht skalieren, d.h. die in der TIFF-Datei festgelegte Auflösung verwenden (Standard)

Ausgabe

Der Adapter gibt ein PDF-Dokument als Binärdatenstrom oder als XML-Dokument aus, das alle Seiten der mehrseitigen TIFF-Datei enthält.

6.16.3 PDF Split and Merge

6.16.3.1 PDF Merger



PDF Merger: Fügt mehrere nicht geschützte PDF-Dokumente zu einem PDF-Dokument zusammen.

Dieser Adapter erwartet eine adapterspezifische Input-XML-Struktur, die mehrere Base64-kodierte Zeichenketten mit den Daten von PDF-Dokumenten enthält. Der Adapter gibt ein XML-Dokument aus, das eine Base64-kodierte Zeichenkette mit den Daten des zusammengefügt PDF-Dokuments enthält.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>Merge</i> : Mehrere PDF-Dokumente zu einem PDF-Dokument zusammenfügen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.PDFMerger</i> : Hauptklasse (Standard)
----------------	--

Input

Dieser Funktions-Adapter erwartet eine adapterspezifische Input-XML-Struktur, in der sämtliche Daten der PDF-Dokumente als Base64-kodierte Zeichenketten in der gewünschten Reihenfolge (festgelegt im Attribut `order`) hinterlegt sind.

```
<Data>
  <input order="1">
    <!--Base64-kodierte Zeichenkette mit Daten des ersten PDF-Dokuments-->
  </input>
  <input order="2">
    <!--Base64-kodierte Zeichenkette mit Daten des zweiten PDF-Dokuments-->
  </input>
</Data>
```

Ausgabe

Der erfolgreich ausgeführte Adapter gibt ein XML-Dokument mit folgendem Muster zurück:

```
<Data>
  <output mimetype="application/pdf" encoding="base64">
    <!--Base64-kodierte Zeichenkette mit Daten des zusammengeführten PDF-Dokuments-->
  </output>
</Data>
```

6.16.3.2 PDF Page Counter



PDF Page Counter: Ermittelt die Anzahl der Seiten eines PDF-Dokuments, das als Input-Dokument verarbeitet wird, und gibt ein XML-Dokument mit der Seitenzahl aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>GetPageCount</i> : Anzahl der Seiten im PDF ermitteln (Standard)
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.pdf.PDFPageCount</i> : Hauptklasse (Standard)
----------------	---

Ausgabe

Der Funktions-Adapter gibt ein XML-Dokument mit folgender Struktur aus:

```
<Pages><!--Anzahl der Seiten--></Pages>
```

6.16.3.3 PDF Recombiner



PDF Recombiner: Fügt Seiten von mehreren nicht geschützten PDF-Dokumenten in einem neuen PDF-Dokument zusammen.

Dieser Adapter erwartet eine adapterspezifische Input-XML-Struktur, die sämtliche Verarbeitungs-Informationen für die PDF-Dokumente enthält. Der Adapter gibt die Daten eines zusammengeführten PDF-Dokuments als Binärdatenstrom oder als XML-Dokument aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>RecombinePDF</i> : Seiten von verschiedenen PDF-Dokumenten zu einem neuen PDF-Dokument zusammenfügen
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.pdf.PDFRecombine</i> <i>r</i> : Hauptklasse (Standard)
<i>ReadPartially</i>	PDF-Dokument in mehreren Teilen in den Arbeitsspeicher laden (kann Performance-Verbesserungen bringen) Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: In mehreren Teilen nacheinander in den Arbeitsspeicher laden • <i>nein</i>: PDF-Dokument(e) In einem Stück in den Arbeitsspeicher laden (Standard)
<i>ToXML</i>	PDF-Dokument Base64-kodiert in einem XML-Dokument ausgeben Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: XML-Dokument ausgeben • <i>nein</i>: PDF-Dokument als Binärdatenstrom ausgeben (Standard)

Input

Dieser Adapter erwartet eine adapterspezifische Input-XML-Struktur, die sämtliche Verarbeitungs-Informationen für das erzeugte PDF-Dokument enthält.

```
<pdfRecombiner>
  <PDF name="Deklaration eines Input-Dokumentbezeichners"
    url="xstore-URL eines Input-PDF-Dokuments im Repository"/>
  <!-- <PDF name="inputname"
    encoding="base64">(Wenn Input-PDF Base64-Zeichenkette)
  </PDF> -->

  <Range pdf="Input-Dokumentbezeichner" from="Startseite"
    to="Endseite" odd="ungerade Seiten (true/false)"
    even="gerade Seiten (true/false)"/>
  <Page pdf="Input-Dokumentbezeichner"
    pageNr="Einzelne Seitenzahl"/>
  <Empty count="Anzahl leerer Seiten"/>
  <All pdf="Input-Dokumentbezeichner"/>
</pdfRecombiner>
```

Erläuterung:

- `<PDF/>`: referenziert je ein Input-Dokument.
Wenn es sich bei einem Input-PDF-Dokument um Base64-kodierte Daten handelt, muss stattdessen der auskommentierte Code angepasst und einkommentiert werden.
- `<Range/>` definiert einen Seitenbereich von Start- bis Endseite.
- `<Page/>` definiert eine Einzelseite.
- `<All/>` definiert sämtliche Seiten eines Input-PDF-Dokuments.
- `<Empty/>` definiert leere Seiten im erzeugten PDF-Dokument.

Beispiel

Folgendes Input-XML-Dokument enthält Informationen zu zwei PDF-Dokumenten, die mit den definierten Seiten und insgesamt drei Leerseiten in einem neuen PDF-Dokument zusammengefügt werden sollen.

```
<pdfRecombiner>
  <PDF name="input1" url="xstore://Project/Folder/InputPDF1.pdf"/>
  <PDF name="input2" url="xstore://Project/Folder/InputPDF2.pdf"/>
  <!-- <PDF name="inputname"
    encoding="base64">base_64_encoded_PDF_document</PDF> -->

  <Range pdf="input1" from="1" to="10" odd="false" even="true"/>
  <Page pdf="input1" pageNr="1"/>
  <Empty count="2"/>
  <Range pdf="input2" from="5" to="10"/>
  <Empty count="1"/>
  <All pdf="input1"/>
</pdfRecombiner>
```

6.16.3.4 PDF Splitter Iterator



PDF Splitter Iterator: Teilt ein nicht geschütztes PDF-Dokument in mehrere neue PDF-Dokumente.

Dieser Adapter erwartet eine adapterspezifische Input-XML-Struktur, die den Pfad zum verarbeiteten PDF-Dokument im Repository enthält. Der Adapter gibt je Iteration die Daten eines PDF-Teildokuments als Binärdatenstrom oder als XML-Dokument aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>SplitIterator</i>: PDF-Dokument in mehrere PDF-Dokumente teilen • <i>Clear</i>: Iterator auf Anfangswert zurücksetzen
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.pdf.PDFSplitterIteratorAdapter</i> : Hauptklasse (Standard)
<i>Start</i>	Seitenzahl, von der gestartet werden soll Mögliche Werte: Beliebige positive ganze Zahl
<i>Step</i>	Anzahl der Seiten, die verarbeitet werden sollen Mögliche Werte: Beliebige positive ganze Zahl
<i>Count</i>	Anzahl der Seiten, die pro Verarbeitungsschritt als separates PDF-Dokument ausgegeben werden sollen Mögliche Werte: Beliebige positive ganze Zahl
<i>ReadPartially</i>	PDF-Dokument in mehreren Teilen in den Arbeitsspeicher laden (kann Performance-Verbesserungen bringen) Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: In mehreren Teilen nacheinander in den Arbeitsspeicher laden • <i>nein</i>: PDF-Dokument(e) in einem Stück in den Arbeitsspeicher laden (Standard)

<i>ToXML</i>	<p>PDF-Dokument Base64-kodiert in einem XML-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: XML-Dokument ausgeben • <i>nein</i>: PDF-Dokument als Binärdatenstrom ausgeben (Standard)
--------------	---

Input

Dieser Funktions-Adapter erwartet eine adapterspezifische Input-XML-Struktur, die den Pfad zum verarbeiteten PDF-Dokument im Repository enthält. Wenn es sich bei einem Input-PDF-Dokument um eine Base64-kodierte Daten handelt, muss stattdessen der auskommentierte Code angepasst und einkommentiert werden.

```
<PDF url="xstore://Project/Folder/Input.pdf"/>
<!-- <PDF encoding="base64">(Wenn Input-PDF Base64-Zeichenkette)</PDF> -->
```

Beispiel

Folgendes Input-XML-Dokument enthält den Pfad zu einem PDF-Dokument im Repository.

```
<PDF url="xstore://Project/Folder/Input.pdf"/>
<!-- <PDF encoding="base64">base_64_encoded_PDF_document</PDF> -->
```

6.17 Security

In Kategorie *Security* werden Adapter und Konnektoren zur Verfügung gestellt, die es ermöglichen, Daten in technischen Prozessen zu verschlüsseln und zu entschlüsseln sowie Signaturen zu erzeugen und zu validieren.

6.17.1 Antivirus



Antivirus: Stellt eine Verbindung zu einem Server der ClamAV, einer Open Source (GPL) Antivirus-Engine, her. Eingabedateien werden zur Laufzeit des Adapters auf Viren überprüft.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Adapter durchführt</p> <p>Mögliche Werte:</p> <p><i>Scan</i>: Eingabedateien auf Viren überprüfen.</p>
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <code>de.softproject.integration.adapter.antivirus.AntiVirusAdapter</code> : Hauptklasse (Standard)
<i>host</i>	IP-Adresse oder Host-Name des ClamAV-Servers Mögliche Werte: <ul style="list-style-type: none"> • Beliebige gültige Zeichenkette, z. B. <code>localhost</code>, <code>127.0.0.1</code> • <code>localhost</code>: ClamAV Host (Standard)
<i>port</i>	Portnummer des ClamAV-Servers Mögliche Werte: <ul style="list-style-type: none"> • Jede beliebige Ganzzahl (Integer) • <code>3310</code>: FTP-Port (Standard)

Statuswerte

1	Der Adapter gibt ein Ergebnisdokument aus. Die Operation war erfolgreich, es wurden keine Viren gefunden <i>oder</i> die Eingabedatei ist leer.
-1	Der Adapter gibt die Eingabedatei aus. Die Operation konnte nicht ausgeführt werden. Details zur Fehlermeldung können den Server-Logs entnommen werden.
400	Der Adapter gibt ein Ergebnisdokument aus. Die Operation war erfolgreich, es wurden Viren in der Eingabedatei gefunden.

Input

Der Adapter erwartet jede Eingabedatei, sowohl Binärdateien als auch Base64-kodierte Dokumente (Text- oder XML-Dateien).

Ausgabe

Wenn die Operation fehlschlug, gibt der Adapter die Eingabedatei aus.

Wenn die Operation erfolgreich war, gibt der Adapter ein UTF-8-kodiertes XML-Ergebnisdokument der nachfolgenden Struktur aus. Wurden Viren gefunden, wird der Wert auf `true` gesetzt, sonst wird der Wert `false` ausgegeben.

```
<?xml version="1.0" encoding="UTF-8"?>
<ScanResult>
  <!--Der Wert true wird ausgegeben, wenn in der Eingabedatei Viren gefunden wurden-->
  <!--Der Wert false wird ausgegeben, wenn in der Eingabedatei keine Viren gefunden wurden-->
  <VirusFound><!--Wert--></VirusFound>
</ScanResult>
```

Abhängigkeiten und Installationshinweise

Dieser Adapter verwendet folgende Bibliotheken: ClamAV Bibliothek `clamav-client-1.0.1.jar`

6.17.1.1 Installation ClamAV

Für eine einwandfreie Funktion des Adapters muss der ClamAV-Server auf derselben Maschine wie die X4 Suite installiert sein und laufen. Eine ausführliche Installationsbeschreibung und Anpassungen an den Konfigurationsdateien finden Sie in dem offiziellen ClamAV [Handbuch](#) (siehe Kapitel 4 und 5).

Download Intallationspakete

Eine Auflistung der verfügbaren Installationspakete finden Sie auf der Website des Anbieters: <https://www.clamav.net/downloads#otherversions>

- Download-Version mit Installationsassistent für Windows (32 bit): `clamav-<version_number>-x86.zip`
- Download-Version mit Installationsassistent für Windows (64 bit): `clamav-<version-number>-win-x64.zip`

6.17.1.2 Aktualisierung der Virendatenbank

Die Einstellungen zur automatischen Aktualisierung der Virendatenbank sind in dem offiziellen ClamAV [Handbuch](#) (siehe Kapitel 4) beschrieben.

6.17.2 BCrypt Calculator



BCrypt Calculator: Erstellt und verifiziert Salted BCrypt-Hashes.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <code>GenerateHash</code>: erstellt einen Salted Hash zum eingegebenen Klartext-Passwort • <code>Verify</code>: verifiziert das Klartext-Passwort mit dem eingegebenen Salted Hash • <code>VerifyAndReplace</code>: verifiziert das Klartext-Passwort mit dem eingegebenen Salted Hash und erstellt einen neuen Salted Hash
-----------	--

Parameter

Adapter	<p>Adapter-Hauptklasse (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.bcrypt.BCryptAdapter</code>: Hauptklasse (Standard)</p>
---------	--

clearText	Klartext-Passwort Mögliche Werte: Beliebiger String, z. B. <i>MeinPasswort</i>  Der Parameter darf nicht leer sein.
bcryptHash	Salted BCrypt-Hash, der mit dem Klartext-Passwort verglichen werden soll; Nötig bei den Operationen <i>Verify</i> und <i>VerifyAndReplace</i> Mögliche Werte: Beliebiger Salted BCrypt-Hash
newclearText	Neues Klartext-Passwort; Nur nötig bei der Operation <i>VerifyAndReplace</i> Mögliche Werte: Beliebiger String, z. B. <i>MeinPasswort</i>  Der Parameter darf nicht leer sein.

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt und im Output befindet sich ein XML-Dokument mit dem Resultat
0	Operationen <i>Verify</i> und <i>VerifyAndReplace</i> : der Parameter <i>bcryptHash</i> stimmt nicht mit dem <i>clearText</i> -String überein
-1	Es kam zu einem Fehler während der Ausführung <ul style="list-style-type: none"> • Die erforderlichen Parameter sind nicht befüllt • Operationen <i>Verify</i> und <i>VerifyAndReplace</i>: der Parameter <i>bcryptHash</i> ist nicht mit einem Salted BCrypt-Hash befüllt

Input

Der Adapter erwartet bei keiner Operation einen Input.

Ausgabe

Je nach Operation gibt der Adapter verschiedene Ergebnisse aus:

- Operationen *GenerateHash* und *VerifyAndReplace*: XML-Dokument mit dem Salted Hash
- Operation *Verify*: XML-Dokument mit dem Ergebnis, ob der Salted Hash dem Klartext-Passwort entspricht
- Operation *VerifyAndReplace* (wenn Salted Hash nicht mit Klartext-Passwort übereinstimmt): XML-Dokument mit dem Ergebnis, dass der Salted Hash nicht dem Klartext-Passwort entspricht
- XML-Dokument mit Fehlermeldungen

Beispiel-Ausgabe

```
<?xml version="1.0" encoding="UTF-8"
<Hash>$2a$10$EcngKl9hSlppvEcDspLzeq.9Blqim1SFNVzY6tbn5ixTZFMh4xs</Hash>
```

6.17.3 Checksum Creator



Checksum Creator: Bildet Prüfsummen für Dateien in einem Ordner und gibt die Prüfsummen gemeinsam mit den Dateinamen und Datei-Eigenschaften als XML-Dokument aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>Count CRC</i>: CRC-Algorithmus zum Bilden der Prüfsummen verwenden • <i>Count Adler</i>: Adler-Algorithmus zum Bilden der Prüfsummen verwenden
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.checksum.ChecksumAdapter</i> : Hauptklasse (Standard)
<i>path</i>	Ordner, für dessen Inhalt Prüfsummen erstellt werden soll Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>%USER_COLLECTION%/Project/Folder</i>)
<i>useRecursion</i>	Ordner rekursiv behandeln Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: Elemente in Unterordnern einbeziehen • <i>nein</i>: Auf die Elemente im angegebenen Ordner beschränken

Input

Dieser Adapter benötigt keinen Input aus dem vorhergehenden Prozess-Schritt.

Ausgabe

Beispiel für eine ausgegebenes XML-Dokument mit Dateinamen, Prüfsummen und einem einbezogenen Unterordner:

```
<list>
  <directory lastModifiedAsString="30.09.2009 15:22:47 MESZ"
    lastModifiedAsLong="1254316967185" size="0" hidden="false">Folder
  </directory>
  <file lastModifiedAsString="30.09.2009 15:22:47 MESZ"
    lastModifiedAsLong="1254316967185" size="123" hidden="false"
    checksum="1376899901">Folder\File.txt</file>
  <file lastModifiedAsString="30.09.2009 15:22:27 MESZ"
    lastModifiedAsLong="1254316947494" size="6" hidden="false"
    checksum="63888547">Textfile.txt</file>
  <file lastModifiedAsString="30.09.2009 15:22:39 MESZ"
    lastModifiedAsLong="1254316959989" size="6" hidden="false"
    checksum="4270080073">XMLDocument.xml</file>
</list>
```

6.17.4 Digest Calculator



Digest Calculator: Generiert bzw. berechnet den Hash-Code (RSA, MD5 etc.) der Input-Daten.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Calculate</i> : Hash-Code berechnen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.digest.DigestAdapter</i> : Hauptklasse (Standard)
<i>algorithm</i>	Verwendeter Algorithmus zur Berechnung des Hashcodes Mögliche Werte: <ul style="list-style-type: none"> • <i>MD5</i>: Message-Digest 5-Algorithmus verwenden (Standard) • <i>SHA</i>: Secure Hash (0)-Algorithmus verwenden • <i>SHA-1</i>: Secure Hash 1-Algorithmus verwenden • <i>SHA-256</i>: Secure Hash 256-Algorithmus verwenden • <i>CRC</i>: Cyclic Redundancy Check-Algorithmus verwenden • <i>ADLER</i>: Adler-Algorithmus verwenden

Ausgabe

Dieser Adapter gibt ein *UTF-8*-kodierte XML-Dokument aus, das den verwendeten Algorithmus und den berechneten Hashcode enthält:

```
<digest type="Algorithmus"><!--Hashcode der Input-Daten--></digest>
```

Der Adapter gibt beispielsweise folgendes XML-Dokument aus:

```
<digest type="MD5">2e5941901e8b5e214dcc5032dfce71bb</digest>
```

6.17.5 JSON WebToken Calculator

JSON WebToken Calculator: Generiert, verifiziert und verarbeitet JSON-WebTokens.

Name des Adapters bis Version 5.5.4: *JSON WebToken*

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • CreateJWT: Generiert ein JSON-WebToken (JWT) aus einem XML-Dokument • Verify: Verifiziert den vom X4 Server generierten JWT • CreateXML: Erstellt aus einem vom X4 Server generierten JWT das dazugehörige XML-Dokument
-----------	--

Parameter

Adapter	<p>Adapter-Hauptklasse (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.json.webtokenJSONWebTokenAdapter</i>: Hauptklasse (Standard)</p>
---------	--

Statuswerte

1	<p>Die Adapter-Operation wurde erfolgreich ausgeführt:</p> <ul style="list-style-type: none"> • Operation CreateJWT: Im Output befindet sich ein JSON-WebToken • Operation Verify: Der JSON-WebToken ist gültig • Operation CreateXML: Im Output befindet sich das XML-Dokument zum JSON-WebToken
-1	<p>Bei der Ausführung des Adapters kam es zu einem Fehler:</p> <ul style="list-style-type: none"> • Operation CreateJWT und CreateXML: Im Output befindet sich eine Fehlermeldung • Operation Verify: Der JSON-WebToken ist ungültig bzw. abgelaufen

Input

Je nach Operation erwartet der Adapter verschiedene Inputs:

- Operation CreateJWT: es wird ein XML-Dokument erwartet, welches folgendes Format besitzt.

Beispiel-Input

```
<?xml version="1.0" encoding="UTF-8">
<TokenContent>
  <Subject name="sub" type="String">1234567890</Subject>
  <ExpiresAt name="exp" type="Integer">1577750400</ExpiresAt>
  <NotBefore name="nbf" type="Integer">946684800</NotBefore>
  <Id name="jti" type="String">007</Id>
  <Claims>
    <Claim name="name" type="String">John Doe</Claim>
    <Claim name="admin" type="boolean">true</Claim>
  </Claims>
</TokenContent>
```

❗ Zusätzliche Informationen zu den verwendenden Claims und deren XML-Objekten finden Sie unter folgendem [Link](#). Bei nicht reservierten Claims können die Namen beliebig vergeben werden, jedoch dürfen sie nicht gleich sein mit bereits reservierten Claims wie beispielsweise *bsp.* oder *exp.*

❗ Als type stehen die Werte *String*, *Boolean*, *Integer*, *Long* und *Double* zur Verfügung.

❗ Für die Elemente *ExpiresAt*, *NotBefore* und *IssuedAt* werden Angaben im Unix-Zeitformat erwartet.

- Operationen *Verify* und *CreateXML*: es wird ein JSON-WebToken erwartet, das aus einem Zwischenschritt stammen oder als Text-Datei (*.txt*) erfolgen kann.

Ausgabe

Je nach Operation gibt der Adapter verschiedene Ergebnisse aus:

- Operation *CreateJWT*: Im Output befindet sich ein JSON-WebToken
- Operation *Verify*: Wenn der JWT gültig ist, wird er wieder ausgegeben
- Operation *CreateXML*: Im Output befindet sich das XML-Dokument zum JSON-WebToken

6.17.6 LDAP Connector

LDAP Connector: Stellt eine Verbindung zu einem LDAP-Server (*Lightweight Directory Access Protocol*) her, um Kontaktdaten abzufragen – bei Bedarf über LDAP Secure. Über adapterspezifische XML-Dokumente als Input-Daten können verschiedene Operationen ausgeführt werden.

❗ Die Konfiguration für LDAP Secure mit selbstsignierten Zertifikaten lässt sich in der Konfigurationsdatei *X4config.xml* vornehmen, siehe LDAPS-Konfiguration.

Eigenschaften (Properties)

Operation	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Execute</i> : Adapter ausführen
-----------	--

Parameter

Adapter	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.ldap.Ldap</i> : Hauptklasse (Standard)
java.naming.factory.initial	Name der Klasse, die den LDAP-Dienst bereitstellt Mögliche Werte: Beliebiger Klassenname, z.B. <i>com.sun.jndi.ldap.LdapCtxFactory</i>
java.naming.provider.url	URL des LDAP(S)-Servers Mögliche Werte: Beliebige URL
java.naming.security.authentication	Authentifizierungsmethode Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>simple</i>)
java.naming.security.principal	Name des Benutzers bzw. der Authentifizierungs-Anwendung (abhängig von der Authentifizierungsmethode) Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>cn=S. User, ou=NewHires, o=JNDITutorial</i>)
java.naming.security.credentials	Zugehöriges Passwort des Benutzers bzw. der Authentifizierungs-Anwendung (abhängig von der Authentifizierungsart) Mögliche Werte: Beliebige Zeichenkette (z.B. <i>mysecret</i>)
java.naming.referral	Definiert, wie mit Weiterleitungen umgegangen werden soll. Mögliche Werte: <ul style="list-style-type: none"> • <i>ignore</i>: Weiterleitungen ignorieren • <i>follow</i>: Allen Weiterleitungen automatisch folgen • <i>throw</i>: Für jede Weiterleitung eine <i>ReferralException</i> auslösen (siehe <i>ReferralException</i> auslösen)

attributesAsElements	Attribute eines Objektes ausgeben Mögliche Werte: <ul style="list-style-type: none"> • <code>true</code>: Attribute eines Objektes ausgeben • <code>false</code>: keine Objekt-Attribute ausgeben (standard)
----------------------	--

ReferralException auslösen

Wenn Weiterleitungen manuell verarbeitet werden sollen, dann müssen folgende Schritte befolgt werden:

1. Exception auffangen.
2. Weiterleitungsinformationen auslesen, indem `ReferralException.getReferralInfo()` verwendet wird und z. B. den Benutzer fragen, ob der Weiterleitung gefolgt werden soll.
 - a. Wenn der Weiterleitung gefolgt werden soll, den Weiterleitungskontext mit `ReferralException.getReferralContext()` auslesen und die ursprüngliche Kontextmethode mit den gleichen Argumenten erneut aufrufen, die auch für den ursprünglichen Aufruf verwendet werden sollte.
 - b. Wenn der Weiterleitung nicht gefolgt werden soll, `ReferralException.skipReferral()` aufrufen.
 - i. Wenn diese Methode `true` zurückgibt, dann bedeutet das, dass mehreren Weiterleitungen gefolgt werden soll.
Wenn das der Fall ist, `ReferralException.getReferralContext()` aufrufen, um fortzufahren. Wenn eine Kontextmethode das Ergebnis aufruft, dann wird für die nächste Weiterleitung wieder eine Exception ausgelöst. Diese Exception ab Schritt 1 behandeln.
 - ii. Wenn die Methode `false` zurückgibt, dann gibt es keine Verweise mehr und dieses Verfahren kann beendet werden.

Weitere Informationen unter <https://docs.oracle.com/javase/jndi/tutorial/ldap/referral/throw.html>.

Input

Dieser Funktions-Adapter erwartet spezifische Input-XML-Strukturen, die LDAP-Befehle enthalten. Diese Input-XML-Strukturen können über XSL-Mappings in Abhängigkeit von den verarbeiteten Daten dynamisch generiert werden.

6.17.6.1 search-Befehl

Mit dem LDAP-Befehl `search` können Sie Einträge im LDAP-Verzeichnis suchen.

Input

Der *LDAP Adapter* erwartet zum Ausführen des `search`-Befehls folgende Input-XML-Struktur:

```
<?xml version="1.0" encoding="UTF-8"?>
<ldap>
  <search>
    <!-- Distinguish Name - base node in LDAP tree where to start search -->
    <name>OU=Accounts,DC=softproject,DC=local</name>
    <!-- Search filter - required -->
    <!-- filter that is applied on search result -->
    <filter>objectclass=user</filter>
    <!-- Search control settings -->
    <controls>
      <!-- Search scope. -->
      <!-- Keywords: object, subtree, onelevel -->
      <!-- object - Results from search will contain one or zero element. -->
      <!-- One element if the named object satisfies the search -->
      <!-- filter specified in search. -->
      <!-- subtree - Search the entire subtree rooted at the named object. -->
      <!-- onelevel - Search one level of the named context. -->
      <scope>subtree</scope>
      <!-- Time limit (ms) - for request duration -->
      <timeLimit>5000</timeLimit>
      <!-- List of attributes that should be returned, separated by colon. -->
      <attributes>uSNCreated,givenname,displayName,cn</attributes>
      <!-- Count limitation for result -->
      <countLimit>100</countLimit>
    </controls>
  </search>
</ldap>
```

Ausgabe

Der erfolgreich ausgeführte Adapter gibt folgendes XML-Dokument aus:

```
<ldap name="dc=sp-linux" filter="(objectclass=*)" scope="subtree"
  attributes="cn,sn" countLimit="4">
  <entry dn="dc=sp-linux" />
  <entry dn="cn=Manager,dc=sp-linux">
    <cn>Manager</cn>
  </entry>
  <entry dn="o=softproject,dc=sp-linux" />
  <entry dn="sn=Tester,o=SoftProject,dc=sp-linux">
    <sn>Tester</sn>
    <cn>John</cn>
  </entry>
</ldap>
```

6.17.6.2 add-Befehl

Mit dem LDAP-Befehl add können Sie Einträge dem LDAP-Verzeichnis hinzufügen.

Input

Der *LDAP Adapter* erwartet zum Ausführen des add-Befehls folgende Input-XML-Struktur:


```
<ldap>
  <add>
    <name>sn=Tester, o=SoftProject, dc=sp-linux</name>
    <!--dn-->
    <attributes>
      <objectClass>person</objectClass>
      <sn>Tester</sn>
      <cn>John</cn>
    </attributes>
    <!--list of attributes that will be added-->
  </add>
</ldap>
```

Ausgabe

Der erfolgreich ausgeführte Funktions-Adapter gibt nach der Ausführung des add-Befehls kein XML-Dokument aus, sondern lediglich den Status 1 (successful) oder im Fehlerfall den Status -1 (Error).

6.17.6.3 modify-Befehl

Mit dem LDAP-Befehl `modify` können Sie Einträge im LDAP-Verzeichnis verändern. Dabei werden wie in einer Transaktion entweder alle Einträge geändert oder im Fehlerfall keine Änderungen an den Einträgen vorgenommen.

Input

Der *LDAP Adapter* erwartet zum Ausführen des `modify`-Befehls folgende Input-XML-Struktur:

```
<ldap>
  <modify>
    <name>dc=sp-linux</name>
    <!--dn-->
    <add>
      <o>Tester</o>
    </add>
    <replace>
      <o>SoftProject</o>
    </replace>
    <remove>
      <o>SoftProject</o>
    </remove>
  </modify>
</ldap>
```

Beispiel für ein Passwort `replace`:

```
<ldap>
  <modify>
    <name>dc=sp-linux</name>
    <add>
      <o>Tester</o>
    </add>
    <replace>
      <unicodePW base64="true">dGVzdCE=</unicodePW>
    </replace>
  </modify>
</ldap>
```

Ausgabe

Der erfolgreich ausgeführte Funktions-Adapter gibt nach der Ausführung des `modify`-Befehls kein XML-Dokument aus, sondern lediglich den Status `1` (`successful`) oder im Fehlerfall den Status `-1` (`Error`).

6.17.6.4 modifyDN(RDN)-Befehl

Mit dem LDAP-Befehl `modifyDN (RDN)` können Sie Einträge im LDAP-Verzeichnis bearbeiten.

Input

Der *LDAP Adapter* erwartet zum Ausführen des `modifyDN-RDN`-Befehls folgende Input-XML-Struktur:

```
<ldap>
  <modifyDN-RDN>
    <oldName>sn=Tester, o=SoftProject, dc=sp-linux</oldName>
    <!--dn-->
    <newName>sn=TestUser, o=SoftProject, dc=sp-linux</newName>
    <!--dn-->
  </modifyDN-RDN>
</ldap>
```

Ausgabe

Der erfolgreich ausgeführte Funktions-Adapter gibt nach der Ausführung des `modifyDN-RDN`-Befehls kein XML-Dokument aus, sondern lediglich den Status `1` (`successful`) oder im Fehlerfall den Status `-1` (`Error`).

6.17.6.5 delete-Befehl

Mit dem LDAP-Befehl `delete` können Sie Einträge im LDAP-Verzeichnis löschen.

Input

Der *LDAP Adapter* erwartet zum Ausführen des `delete`-Befehls folgende Input-XML-Struktur:

```
<ldap>
  <delete>
    <name>sn=Tester, o=SoftProject, dc=sp-linux</name>
    <!--dn-->
  </delete>
</ldap>
```

Ausgabe

Der erfolgreich ausgeführte Funktions-Adapter gibt nach der Ausführung des delete-Befehls kein XML-Dokument aus, sondern lediglich den Status 1 (successful) oder im Fehlerfall den Status -1 (Error).

6.17.6.6 compare-Befehl

Mit dem LDAP-Befehl compare können Sie Einträge im LDAP-Verzeichnis vergleichen.

Input

Der *LDAP Adapter* erwartet zum Ausführen des compare-Befehls folgende Input-XML-Struktur:

```
<ldap>
  <compare>
    <name>sn=Tester, o=SoftProject, dc=sp-linux</name>
    <!--dn-->
    <attributes>
      <objectClass>person</objectClass>
      <sn>Tester</sn>
      <cn>John</cn>
    </attributes>
    <!--list of attributes that will be compared by values-->
    <controls>
      <scope>subtree</scope>
      <!--keywords: object,subtree,onelevel-->
      <timeLimit />
      <!--timeLimit (ms)-->
    </controls>
  </compare>
</ldap>
```

Ausgabe

Der erfolgreich ausgeführte Funktions-Adapter gibt nach der Ausführung des compare-Befehls kein XML-Dokument aus, sondern lediglich den Status CompareTRUE, wenn der LDAP-Eintrag mit den Angaben im Input-XML-Dokument übereinstimmt, oder im anderenfalls den Status CompareFALSE.

6.17.7 Password Generator



Password Generator: Erzeugt ein zufälliges Passwort anhand von Passwortrichtlinien, die in den Adapter-Parametern definiert werden. Dieser Adapter erwartet keinen Input und gibt ein XML-Dokument mit dem erzeugten Passwort aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: Generate: Ein Passwort erzeugen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.security.PasswordGenerator</i> : Hauptklasse (Standard)
<i>length</i>	Anzahl der Zeichen für das zu erzeugende Passwort Mögliche Werte: <ul style="list-style-type: none">• Beliebige positive ganze Zahl• 8: Passwort mit 8 Zeichen erzeugen (Standard)
<i>numberRequired</i>	Definiert, ob eine numerische Ziffer im Passwort enthalten sein soll Mögliche Werte: <ul style="list-style-type: none">• <i>True</i>: Passwort wird eine numerische Ziffer enthalten• <i>False</i>: Passwort wird keine numerische Ziffer enthalten (Standard)
<i>lowerCaseCharRequired</i>	Definiert, ob ein Kleinbuchstabe im Passwort enthalten sein soll Mögliche Werte: <ul style="list-style-type: none">• <i>True</i>: Passwort wird einen Kleinbuchstaben enthalten• <i>False</i>: Passwort wird keinen Kleinbuchstaben enthalten (Standard)
<i>upperCaseCharRequired</i>	Definiert, ob ein Großbuchstabe im Passwort enthalten sein soll Mögliche Werte: <ul style="list-style-type: none">• <i>True</i>: Passwort wird einen Großbuchstaben enthalten• <i>False</i>: Passwort wird keinen Großbuchstaben enthalten (Standard)

<i>specialCharRequired</i>	Definiert, ob ein Sonderzeichen (<i>@#\$%</i>) im Passwort enthalten sein soll Mögliche Werte: <ul style="list-style-type: none"> • <i>True</i>: Passwort wird ein Sonderzeichen enthalten • <i>False</i>: Passwort wird kein Sonderzeichen enthalten (Standard)
----------------------------	--

Statuswerte

1 (successful)	Die Operation wurde erfolgreich ausgeführt.
-1 (failed)	Die Operation konnte aufgrund eines technischen Fehlers nicht ausgeführt werden.

Input

Dieser Adapter erwartet keinen Input.

Ausgabe

Dieser Adapter gibt ein XML-Dokument mit folgender Struktur aus:


```
<Password><!-- Erzeugtes Passwort --></Password>
```

6.17.8 PEP Adapter





PEP Adapter: Prüft innerhalb von technischen Prozessen, ob ein bestimmter Benutzer von einem PDP (*Policy Decision Point*) berechtigt ist, die nachfolgenden Prozess-Schritte auszuführen. Besitzt der Benutzer die erforderlichen Berechtigungen, wird das Input-Dokument wieder an den technischen Prozess übergeben und der technische Prozess fortgesetzt. Andernfalls gibt der Adapter einen negativen Status und ein leeres Dokument zurück.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>abac</i>: Prüft, ob der Benutzer für die Aktion und Anwendung berechtigt ist • <i>FilterXML</i>: Prüft ein XML-Dokument auf die Attribute <i>pepApp</i> und <i>pepAction</i> und filtert alle Elemente aus dem XML-Dokument für die der Anwender keine Berechtigungen hat.  Die Attribute müssen im Namespace <i>namespace_x4pep</i> liegen.
------------------	--


Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.pep.PEPAdapter</i> : Hauptklasse (Standard)
<i>realm</i>	Realm, der zur Anmeldung verwendet werden soll
<i>username</i>	Anmeldename des Benutzers <div> Die Prüfung wird mit dem angegebenen Benutzer durchgeführt. Wenn kein Benutzer angegeben ist, wird der Kontext automatisch ermittelt (z.B. aus HTTP-Request).</div>
<i>password</i>	Passwort zur Anmeldung
<i>x4Token</i>	X4-Token für die Anmeldung <div> Wenn Benutzername und Passwort nicht gesetzt sind, kann der X4-Token für die Anmeldung verwendet werden. Sind Benutzername und Passwort gesetzt, wird der Token ignoriert, auch wenn die Anmeldung fehlerhaft sein sollte.</div>
<i>pepApp</i>	Gibt an, in welchem Anwendungskontext die Aktion ausgeführt und geprüft werden soll <i>Beispiel: Custom Placeholders</i>
<i>pepAction</i>	Gibt an, welche Aktion ausgeführt und geprüft werden soll <i>Beispiel: group:read group:write</i>
<i>positiveErrorCode</i>	Definiert, ob die fachlichen Fehlercodes (–500, –503, –403 und –401) als positive Statuscodes ausgegeben werden. Dadurch wird der Prozess bei einem fachlichen Fehler nicht zwingend mit einer Fehlermeldung im Server-Log abgebrochen. Mögliche Werte: <ul style="list-style-type: none">• <i>true</i>: fachlichen Fehlercodes als positive Statuscodes ausgegeben• <i>false</i>: fachlichen Fehlercodes als negative Statuscodes ausgegeben

Statuswerte

<i>–503 oder 503 (unavailable)</i>	Die Prüfung konnte nicht durchgeführt werden; Der Dienst ist entweder ausgelastet oder zeitweise nicht verfügbar.
<i>–500 oder 500 (error)</i>	Während der Ausführung trat ein Fehler im PDP-Dienst auf.
<i>–403 oder 403 (forbidden)</i>	Die Prüfung ist fehlgeschlagen; Der Anwender ist angemeldet, hat jedoch keine Erlaubnis.

-401 oder 401(auth required)	Die Prüfung ist fehlgeschlagen; Der Anwender ist nicht angemeldet.
-1(error)	Während der Ausführung des Adapters trat ein Fehler auf. Nähere Details finden Sie im Server-Log.
1(successful)	<ul style="list-style-type: none"> • Operation abac: Die notwendigen Berechtigungen sind vorhanden; Der Adapter gibt das geprüfte Input-Dokument unverändert aus. • Operation FilterXML: Der Adapter hat das Input-XML geprüft und gibt das gefilterte Dokument wieder aus.

 Alle Prüfungsfehler-Statuscodes sind negativ, damit sichergestellt ist, dass die Ausführung des technischen Prozesses an dieser Stelle abbricht.

Input

Operation FilterXML

Für die Operation FilterXML erwartet der Adapter folgende Input-Struktur:

```
<ActionButtonConfig xmlns:ns="namespace_x4pep">
  <Button actionType="GET" selectionNeeded="false" displayType="MODAL_GWT_DIALOG"
buttonId="newId" ns:pepAction="user:create" width="735" processURL=".." ns:pepApp="UR
M"></Button>
  <Button actionType="MULTI_POST" selectionNeeded="true" displayType="MODAL_GWT_DI
ALOG" buttonId="deleteId" ns:pepAction="user:delete" width="300" processURL=".."
ns:pepApp="URM"></Button>
  <Button actionType="POST" selectionNeeded="false" displayType="DOWNLOAD"
buttonId="exportId" ns:pepAction="user:list | user:read" processURL=".." ns:pepApp="U
RM"></Button>
</ActionButtonConfig>
```

Ausgabe


Operation abac

Der Adapter gibt bei erfolgreicher Prüfung das Input-Dokument unverändert aus. Andernfalls wird ein leeres Dokument ausgegeben.

Operation FilterXML

Unter der Annahme, dass der Benutzer im Prozess-Kontext nur eine Lese-Berechtigung (*user:read*) besitzt, gibt der Adapter folgende XML aus:

```
<ActionButtonConfig>
  <Button actionType="POST" selectionNeeded="false" displayType="DOWNLOAD"
buttonId="exportId" processURL=".."></Button>
</ActionButtonConfig>
```

 Die Attribute pepApp und pepAction werden aus dem Ergebnis-Dokument entfernt.

6.17.9 Random Number Generator

Random Number Generator: Erstellt Zufallszahlen basierend auf der angegebenen Konfiguration.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Generate: generiert ein XML-Dokument mit einer oder mehreren Zufallszahlen • SetRandomNumbers: generiert Zufallszahlen in ein bereits vorhandenes XML-Dokument mittels XPath
-----------	--

Parameter

Adapter	<p>Adapter-Hauptklasse (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.randomNumberGenerator.RandomNumberGenerator</i>: Hauptklasse (Standard)</p>
min	<p>Minimum, ab welchem Zahlenwert eine Zufallszahlen generiert werden soll</p> <p>Mögliche Werte: beliebige positive Zahl, z. B. <i>1</i></p> <p> Darf weder negativ noch 0 sein.</p>
max	<p>Maximum, ab welchem Zahlenwert eine Zufallszahlen generiert werden soll</p> <p>Mögliche Werte: beliebige positive Zahl, z. B. <i>100</i></p> <p> Darf weder negativ noch 0 sein.</p>
count	<p>Anzahl der zu generierenden Zufallszahlen</p> <p>Mögliche Werte: beliebige positive Zahl, z. B. <i>4</i></p> <p> Darf weder negativ noch 0 sein.</p>
XPath	<p>XPath zum gewünschten XML-Element</p> <p> Nur wichtig für die Operation setRandomNumbers.</p> <p>Mögliche Werte: Beliebiger XPath-Ausdruck, z. B. <i>Data/Beispiel</i></p>
elementName	<p>Namen des XML-Elements in welchem die Zufallszahl(en) generiert werden soll(en).</p> <p> Nur wichtig für die Operation setRandomNumbers.</p> <p>Mögliche Werte: Beliebiger Name für ein XML-Element, z. B. <i>Id</i></p>

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt und im Output befindet sich ein XML-Dokument mit der/den Zufallszahl(en)
-1	Es kam zu einem Fehler während der Ausführung <ul style="list-style-type: none"> • Operationen Generate und setRandomNumbers: der Parameter count ist kleiner 1 • Operation setRandomNumbers: es liegt kein XML-Dokument als Input vor

Input

Der Adapter erwartet bei der Operation SetRandomNumbers ein XML-Dokument als Input.

Ausgabe

Je nach Operation, gibt der Adapter folgende Ergebnisse aus:

- Operation Generate: XML-Dokument mit der/den generierten Zufallszahl(en)
- Operation SetRandomNumbers: XML-Dokument mit der/den hinzugefügten Zufallszahl(en)

Beispiel

```
<?xml version="1.0" encoding="UTF-8"
<Data>
  <random>28</random>
  <random>6</random>
  <random>10</random>
</Data>
```

6.17.10 RSA Encryption



RSA Encryption: Verschlüsselt oder entschlüsselt Input-Daten (Byte-Datenstrom) mit dem RSA-Algorithmus und gibt einen Byte-Datenstrom aus (2048-Bit-RSA-Schlüssel).

Ehemalige Bezeichnung bis Version 4.5.1: *RSA Encryption Adapter*

- i** Die maximale Byte-Länge von verschlüsselten Input-Daten beträgt die Länge des RSA-Schlüssels in Bit dividiert durch 8, minus 11 Füllbytes. So berechnet sich beispielsweise die maximale Bytelänge bei einem 1024-Bit-RSA-Schlüssel wie folgt: $1024/8 - 11 = 117$ Bytes; Ein 2048-Bit-RSA-Schlüssel entschlüsselt Input-Daten mit maximal 245 Bytes Länge.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt
	Mögliche Werte: <ul style="list-style-type: none"> • <i>Encrypt</i>: Input-Daten RSA-verschlüsseln • <i>Decrypt</i>: RSA-verschlüsselte Input-Daten entschlüsseln

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.rsa.RSAAAdapter:</i> Hauptklasse (Standard)
<i>path</i>	Pfad zur Keystore-Datei Mögliche Werte: Beliebiger absoluter oder relativer Pfad
<i>alias</i>	Kurzbezeichnung des Eintrags, der einen Schlüssel im Keystore besitzt Mögliche Werte: Beliebige Zeichenkette
<i>keyType</i>	Verwendeter Schlüsseltyp Mögliche Werte: <ul style="list-style-type: none"> • <i>private</i>: Privaten Schlüssel verwenden (Standard) • <i>public</i>: Öffentlichen Schlüssel verwenden
<i>keystorePassword</i>	Passwort für die Keystore-Datei Mögliche Werte: Beliebige Zeichenkette
<i>privatePassword</i>	Passwort, um den privaten Schlüssel aus der Keystore-Datei zu erhalten Mögliche Werte: Beliebige Zeichenkette

6.17.11 SAML Calculator



SAML Calculator: Signiert, validiert, verschlüsselt und entschlüsselt SAML Assertions und Requests.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Encrypt Assertion: SAML Assertion verschlüsseln Erforderliche Parameter: cryptAlias, cryptPassword • Decrypt Assertion: SAML Assertion entschlüsseln Erforderliche Parameter: cryptAlias, cryptPassword • Sign Assertion: SAML Assertion mit einem Zertifikat signieren Erforderliche Parameter: signatureAlias, signaturePassword, algorithm • Validate Assertion: SAML Assertion validieren Erforderliche Parameter: cryptAlias, cryptPassword, signatureRequired, skipValidation • Sign Request: SAML Request mit einem Zertifikat signieren Erforderliche Parameter: signatureAlias, signaturePassword, algorithm • Validate Request: SAML Request validieren Erforderliche Parameter: signatureRequired, skipValidations
------------------	---

Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.x4.adapter.saml.SamlAdapter</i>: Hauptklasse (Standard)</p>
keystoreUrl	<p>URL des Keystores, der verwendet wird, um die SAML Assertions zu verschlüsseln und zu entschlüsseln.</p> <p>Mögliche Werte: String (URL)</p>
keystorePassword	<p>Passwort des Keystores, der verwendet wird, um die SAML Assertions zu verschlüsseln und zu entschlüsseln.</p> <p>Mögliche Werte: String</p>
keystoreType	<p>Art des Keystores, der verwendet wird, um die SAML Assertions zu verschlüsseln und zu entschlüsseln.</p> <p>Mögliche Werte: Art des Keystores, z. B. JKS, PKCS12, ...</p>
signatureRequired	<p>Legt fest, ob eine Signatur benötigt wird, um die Datei zu entschlüsseln.</p> <p>Mögliche Werte: true / false</p>

skipValidations	<p>Legt fest, ob die Prüfung von Signatur und Gültigkeit während des Entschlüsselns übersprungen werden sollen.</p> <p>Mögliche Werte: true / false</p>
cryptAlias	Alias des Schlüsselpaars oder Zertifikats, das verwendet wird, um SAML Assertions zu verschlüsseln und zu entschlüsseln.
cryptPassword	Passwort des Schlüsselpaars, das verwendet wird, um SAML Assertions zu entschlüsseln.
signatureAlias	Alias des Schlüsselpaars, das verwendet wird, um SAML Assertions zu signieren.
signaturePassword	Passwort des Schlüsselpaars, das verwendet wird, um SAML Assertions zu signieren.
algorithm	<p>Algorithmus, der verwendet werden soll, um SAML Assertions zu signieren.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • SHA1: • SHA256: • SHA512:

Statuswerte

1	Der Adapter gibt ein Ergebnis aus.
-1	Fehler bei der Ausführung des Adapters (Details siehe Server-Log)

Input

Der Adapter erwartet je nach gewählter Operation eine SAML Assertion oder ein SAML Request:

- SAML Assertion:

Beispiel Input SAML Assertion

```

<saml:Assertion
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  ID="_d71a3a8e9fcc45c9e9d248ef7049393fc8f04e5f75" Version="2.0"
  IssueInstant="2014-07-17T01:01:48Z">
  <saml:Issuer>http://idp.example.com/metadata.php</saml:Issuer>
  <saml:Subject>
    <saml:NameID
      SPNameQualifier="http://sp.example.com/demo1/metadata.php"
      Format="urn:oasis:names:tc:SAML:2.0:nameid-format:transient">
      _ce3d2948b4cf20146dee0a0b3dd6f69b6cf86f62d7
    </saml:NameID>
    <saml:SubjectConfirmation Method="urn:oasis:names:tc:SAML:
2.0:cm:bearer">
      <saml:SubjectConfirmationData NotOnOrAfter="2024-01-18T06:21:48Z"
        Recipient="http://sp.example.com/demo1/index.php?acs"
        InResponseTo="ONELOGIN_4fee3b046395c4e751011e97f8900b5273d56685
"/>
      </saml:SubjectConfirmation>
    </saml:Subject>
    <saml:Conditions NotBefore="2014-07-17T01:01:18Z" NotOnOrAfter="2024-01-18T
06:21:48Z">
      <saml:AudienceRestriction>
        <saml:Audience>http://sp.example.com/demo1/metadata.php</saml:Audie
nce>
      </saml:AudienceRestriction>
    </saml:Conditions>
    <saml:AuthnStatement AuthnInstant="2014-07-17T01:01:48Z"
      SessionNotOnOrAfter="2024-07-17T09:01:48Z"
      SessionIndex="_be9967abd904ddcae3c0eb4189adbe3f71e327cf93">
      <saml:AuthnContext>
        <saml:AuthnContextClassRef>urn:oasis:names:tc:SAML:
2.0:ac:classes:Password</saml:AuthnContextClassRef>
      </saml:AuthnContext>
    </saml:AuthnStatement>
    <saml:AttributeStatement>
      <saml:Attribute Name="uid"
        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
        <saml:AttributeValue xsi:type="xs:string">test</saml:AttributeValue
>
      </saml:Attribute>
      <saml:Attribute Name="mail"
        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
        <saml:AttributeValue xsi:type="xs:string">test@example.com</saml:At
tributeValue>
      </saml:Attribute>
      <saml:Attribute Name="eduPersonAffiliation"
        NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
        <saml:AttributeValue xsi:type="xs:string">users</saml:AttributeValu
e>
        <saml:AttributeValue xsi:type="xs:string">examplerole1</saml:Attrib
uteValue>
      </saml:Attribute>

```

```
</saml:AttributeStatement>
</saml:Assertion>
```

- SAML Request:

Beispiel Input SAML Request

```
<samlp:AuthnRequest ID="123456789" Version="2.0"
  IssueInstant="2019-01-01T12:00:00"
  Destination="https://www.example.org/saml/login"
  ForceAuthn="false" IsPassive="false"
  xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol">
  <saml:Issuer
    xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">BPMX4</saml:Issuer>
  <samlp:NameIDPolicy AllowCreate="true"
    Format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified" />
</samlp:AuthnRequest>
```

Ausgabe

Dieser Funktions-Adapter gibt je nach ausgeführter Operation eine andere Datei als Output aus:

- Encrypt Assertion: Der Adapter gibt eine entschlüsselte SAML-Assertion aus.
- Sign Assertion: Der Adapter gibt eine signierte SAML-Assertion aus.
- Sign Request: Der Adapter gibt einen signierte SAML-Request aus.

6.17.12 SSH Connector

SSH Connector: Verbindet sich mit einem SSH-Server über das Secure Shell Protocol.

Eigenschaften (Properties)

Operation	Vom Adapter ausgeführte Operation Mögliche Werte: <ul style="list-style-type: none"> • Execute: Führt den im Input eingegebenen Befehl auf dem Server aus
-----------	---

Parameter

Adapter	Adapter-Hauptklasse (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.ssh.SSHAdapter</i> : Hauptklasse (Standard)
host	Server-Host Mögliche Werte: Beliebiger Host oder IP-Adresse, z. B. <i>localhost</i> oder beliebige IP-Adresse
user	Benutzer, der sich auf dem Server verbinden soll Mögliche Werte: Beliebige Zeichenkette mit dem Benutzernamen, z. B. <i>userName</i>

password	Passwort zum Einloggen auf dem Server Mögliche Werte: Beliebige Zeichenkette mit dem Passwort, z. B. <i>myPassword</i>
port	Server-Port Mögliche Werte: <ul style="list-style-type: none"> • 22 (Standard) • beliebige ganze Zahl mit dem Port
timeout	Wartezeit, bis die Verbindung zum Server hergestellt ist in Millisekunden Mögliche Werte: <ul style="list-style-type: none"> • 30000 ms (Standard) • beliebige ganze Zahl für die Zeit in Millisekunden

Statuswerte


0	Die Adapter-Operation wurde erfolgreich ausgeführt
Beliebige Zahl außer 0	Bei der Ausführung des Adapters kam es zu einem Fehler

Input

Der Adapter ein spezifisches XML-Dokument als Input:

Beispiel-Input

```
<?xml version="1.0" encoding="UTF-8"
<Root>
  <Command>netstat -plnt | grep 8080</Command>
</Root>
```

 Das Element <Command> enthält den auszuführenden Befehl.

Ausgabe

Der Adapter gibt ein XML-Dokument mit dem Output und dem Statuswert aus.

Ausgabe für das Beispiel-Input

```
<?xml version="1.0" encoding="UTF-8"
<Output>
  <CommandOutput>tcp 0 0 0.0.0.0:8080 0.0.0.0:* LISTEN - </CommandOutput>
  <ExitStatus>0</ExitStatus>
</Output>
```





6.17.13 UUID Generator

UUID Generator: Generiert universelle einzigartige Identifikatoren (UUIDs).

Eigenschaften (Properties)

Operation	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • Generate: generiert ein XML-Dokument mit einer oder mehreren UUIDs • SetUUID: generiert UUIDs in ein bereits vorhandenes XML-Dokument mittels XPath-UUIDs
-----------	--

Parameter

Adapter	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.uuid.UuidGeneratorAdapter</i> : Hauptklasse (Standard)
count	Anzahl der zu generierenden UUIDs  Nur wichtig für die Operation Generate Mögliche Werte: Beliebige positive Zahl, z. B. 1  Darf nicht negativ oder 0 sein.
generateBraces	Generierte UUIDs in eine geschweifte Klammer setzen Mögliche Werte: <ul style="list-style-type: none"> • true: generiert die UUID in einer geschweiften Klammer • false: generiert keine geschweifte Klammer
xPath	XPath zum gewünschten XML-Element  Nur wichtig für die Operation SetUUID Mögliche Werte: Beliebiger XPath-Ausdruck, z. B. Data/Beispiel
elementName	Name des XML-Elements, in welchem die UUID generiert werden soll. Das Element kann bereits vorhanden sein und wird dann ggf. überschrieben.  Nur wichtig für die Operation SetUUID Mögliche Werte: Beliebiger Name für ein XML-Element, z. B. Id

Statuswerte

1	Der Adapter wurde erfolgreich ausgeführt und in der Ausgabe befindet sich ein XML-Dokument mit der/den UUID(s)
---	--

-1	<p>Der Adapter konnte nicht ausgeführt werden</p> <p>Mögliche Gründe:</p> <ul style="list-style-type: none"> • Operation Generate: der Parameter count ist kleiner oder gleich 0 • Operation SetUUID: es liegt kein XML-Dokument als Input vor
----	--

Input

Der Adapter erwartet ein XML-Dokument für die Operation Generate.

Ausgabe

Das ausgegebene Dokument ist abhängig von der gewählten Operation:

- Generate: Der Adapter gibt ein XML-Dokument mit der/den generierten UUID(s) aus
- SetUUID: Der Adapter gibt ein XML-Dokument mit der/den hinzugefügten UUID(s) aus

Beispiel-Ausgabe

```
<?xml version="1.0" encoding="UTF-8"
<Data>
  <UUID>de41435e-f0a2-4fea-bdf7-9c8a261f8948</UUID>
  <UUID>fdeb6f6f-1abb-4d34-a405-a8f9a893e0ea</UUID>
  <UUID>d124a7fe-17d1-4850-a49c-6bdbe92a4c3c</UUID>
</Data>
```

6.17.14 XML Encryption

6.17.14.1 XML Encryption (Keystore)



XML Encryption (Keystore): Verschlüsselt oder entschlüsselt ein beliebiges XML-Dokument gemäß dem W3C-Standard XML-Encryption (XML-Enc) auf Basis eines Keystores vom Typ *PKCS12* oder *JCEKS*. Der Adapter unterstützt die Verschlüsselungsalgorithmen AES mit 128, 192 oder 256 Bit Schlüssellänge sowie Triple-DES. Für Ver-/Entschlüsselung auf Basis eines Passworts, siehe [XML Encryption \(Password\)](#).

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Encrypt</i>: XML-Dokument verschlüsseln • <i>Decrypt</i>: XML-Dokument entschlüsseln
------------------	---

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.xml.encryption.XmlEncryption</code>: Hauptklasse (Standard)</p>
<i>method</i>	<p>Algorithmus zur Verschlüsselung bzw. Entschlüsselung</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • AES_128: AES-Algorithmus mit 128 Bit Schlüssellänge verwenden • AES_192: AES-Algorithmus mit 192 Bit Schlüssellänge verwenden • AES_256: AES-Algorithmus mit 256 Bit Schlüssellänge verwenden • TRIPLEDES: Triple-DES-Algorithmus verwenden
<i>xpath</i>	<p>Pfad zum Element bzw. zu den Elementen, die verschlüsselt/entschlüsselt werden sollen als XPath-Ausdruck</p> <p>Mögliche Werte: Beliebiger gültiger XPath-Ausdruck, z.B. <code>//Data</code></p>
<i>scope</i>	<p>Bestimmt den Wirkungsbereich für die Verschlüsselung innerhalb der per XPath adressierten XML-Dokumentstruktur</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Element: Das gesamte Element wird verschlüsselt • Content: Lediglich der Inhalt des Elements wird verschlüsselt
<i>keyStoreURL</i>	<p>URL zu einem gültigen Keystore-Dokument</p> <p>Mögliche Werte: Beliebige gültige URL, z.B. <code>x4db:/1/Project/MyKeystore.pkcs12</code></p>
<i>keyStoreType</i>	<p>Verwendeter Keystore-Typ</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • jceks: Keystore im JCEKS-Format verwenden • pkcs12: Keystore im PKCS12-Format verwenden
<i>password</i>	<p>Passwort zum Zugriff auf den Keystore</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>

<i>alias</i>	Alias für den Schlüssel im Keystore Mögliche Werte: Beliebige Zeichenkette
<i>aliasPassword</i>	Passwort um den Schlüssel über dessen o.g. Alias abzurufen (optional) Mögliche Werte: <ul style="list-style-type: none"> • Beliebige Zeichenkette • (leer): Keystore-Passwort verwenden (Standard)

Statuswerte

<i>-1(error)</i>	Fehler bei der Ausführung des Adapters (Details siehe Server-Log) Im Fehlerfall gibt der Adapter zudem eine Textdatei aus, welche die Exception enthält.
<i>0(empty)</i>	Dokument wurde nicht verändert. Es wurden keine passenden Elemente gefunden.
<i>1(successful)</i>	Dokument wurde verarbeitet und die entsprechenden Elemente ver-/entschlüsselt.

Beispiel

Im folgenden XML-Dokument sollen per XML-Encryption mit dem AES128-Algorithmus die Inhalte aller Data-Elemente verschlüsselt werden:

Input
<pre> <RootElement> <Data>Hello</Data> <Data>World!</Data> <Public>© SoftProject GmbH</Public> </RootElement> </pre>

Der Adapter liefert dann beispielsweise folgendes verschlüsselte Ergebnis-XML-Dokument zurück:

Ergebnis

```

<RootElement>
  <Data>
    <xenc:EncryptedData xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
      Type="http://www.w3.org/2001/04/xmlenc#Content">
      <xenc:EncryptionMethod
        Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc" />
      <xenc:CipherData>
        <xenc:CipherValue>JS1lBp4RcMk6RepmpptlJnBGqbpINGHSV9THG7+UJcY
        </xenc:CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedData>
  </Data>
  <Data>
    <xenc:EncryptedData xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
      Type="http://www.w3.org/2001/04/xmlenc#Content">
      <xenc:EncryptionMethod
        Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc" />
      <xenc:CipherData>
        <xenc:CipherValue>r0+m1iPsw4/PaP2AokM30DqsqeD10YRvD7KNG+bnxc
        </xenc:CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedData>
  </Data>
  <Public>© SoftProject GmbH</Public>
</RootElement>

```

6.17.14.2 XML Encryption (Password)

XML Encryption (Password): Verschlüsselt oder entschlüsselt ein beliebiges XML-Dokument gemäß dem W3C-Standard XML-Encryption (XML-Enc) auf Basis eines Passworts. Der Adapter unterstützt die Verschlüsselungsalgorithmen AES mit 128, 192 oder 256 Bit Schlüssellänge sowie Triple-DES. Für Ver-/Entschlüsselung auf Basis eines Keystores, siehe [XML Encryption \(Keystore\)](#).

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>Encrypt</i>: XML-Dokument verschlüsseln • <i>Decrypt</i>: XML-Dokument entschlüsseln
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.xml.encryption.XmlEncryption</code>: Hauptklasse (Standard)</p>
<i>method</i>	<p>Algorithmus zur Verschlüsselung bzw. Entschlüsselung</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • AES_128: AES-Algorithmus mit 128 Bit Schlüssellänge verwenden • AES_192: AES-Algorithmus mit 192 Bit Schlüssellänge verwenden • AES_256: AES-Algorithmus mit 256 Bit Schlüssellänge verwenden • TRIPLEDES: Triple-DES-Algorithmus verwenden
<i>xpath</i>	<p>Pfad zum Element bzw. zu den Elementen, die verschlüsselt/entschlüsselt werden sollen als XPath-Ausdruck</p> <p>Mögliche Werte: Beliebiger gültiger XPath-Ausdruck, z.B. <code>//Data</code></p>
<i>scope</i>	<p>Bestimmt den Wirkungsbereich für die Verschlüsselung innerhalb der per XPath adressierten XML-Dokumentstruktur</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Element: Das gesamte Element wird verschlüsselt • Content: Lediglich der Inhalt des Elements wird verschlüsselt
<i>password</i>	<p>Passwort, das zur Verschlüsselung und Entschlüsselung verwendet werden soll</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>

Statuswerte

<i>-1(error)</i>	<p>Fehler bei der Ausführung des Adapters (Details siehe Server-Log)</p> <p>Im Fehlerfall gibt der Adapter zudem eine Textdatei aus, welche die Exception enthält.</p>
<i>0(empty)</i>	Dokument wurde nicht verändert. Es wurden keine passenden Elemente gefunden.
<i>1(successful)</i>	Dokument wurde verarbeitet und die entsprechenden Elemente ver-/entschlüsselt.

Beispiel

Im folgenden XML-Dokument sollen per XML-Encryption mit dem AES128-Algorithmus die Inhalte aller Data-Elemente verschlüsselt werden:

Input

```
<RootElement>
  <Data>Hello</Data>
  <Data>World!</Data>
  <Public>© SoftProject GmbH</Public>
</RootElement>
```

Der Adapter liefert dann beispielsweise folgendes verschlüsselte Ergebnis-XML-Dokument zurück:

Ergebnis

```
<RootElement>
  <Data>
    <xenc:EncryptedData xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
      Type="http://www.w3.org/2001/04/xmlenc#Content">
      <xenc:EncryptionMethod
        Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc" />
      <xenc:CipherData>
        <xenc:CipherValue>JS1lBp4RcMk6RepmpLJnBGqbpINGHSV9THG7+UJcY
        </xenc:CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedData>
  </Data>
  <Data>
    <xenc:EncryptedData xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
      Type="http://www.w3.org/2001/04/xmlenc#Content">
      <xenc:EncryptionMethod
        Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc" />
      <xenc:CipherData>
        <xenc:CipherValue>r0+m1iPsw4/PaP2AokM30DqsqeD10YRvD7KNG+bnxc
        </xenc:CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedData>
  </Data>
  <Public>© SoftProject GmbH</Public>
</RootElement>
```

6.17.15 XML Signature



XML Signature: Fügt einem XML-Dokument eine Signatur hinzu oder überprüft eine Signatur innerhalb eines XML-Dokuments auf Basis einer Keystore-Datei im Format PKCS#12.

Ehemalige Bezeichnung bis Version 4.6: *Digital Signature Adapter*

Eigenschaften (Properties)

<i>Custom Parameters</i>	Benutzerdefinierte Parameter setzen; siehe Benutzerdefinierte Parameter verwenden
--------------------------	---

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • Sign: XML-Dokument signieren • Check signature: Gültigkeit der Signatur überprüfen
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <code>de.softproject.integration.adapter.digitalsignature.DigitalSignatureAdapter</code>
<i>keyStore</i>	Pfad zu einer Keystore-Datei im Format PKCS#12 Mögliche Werte: <ul style="list-style-type: none"> • Beliebiger relativer oder absoluter Pfad zu einem PKCS#12-Keystore • <i>Beispiel:</i> <code>%USER_COLLECTION%/Project/Folder/Keystore.p12</code>
<i>ksPass</i>	Passwort des Keystores Mögliche Werte: Beliebige Zeichenkette
<i>keyPass</i>	Passwort für einen Schlüssel (optional) Mögliche Werte: Beliebige Zeichenkette

Input

Der Adapter erwartet für die Operation Sign ein beliebiges XML-Dokument als Input.

Output

Für die Operation Sign gibt der Adapter ein signiertes XML-Dokument aus.

6.17.16 AES Encryption

AES Encryption : Verschlüsselt und entschlüsselt Eingabedaten mit dem AES-Verschlüsselungsalgorithmus.

AES-Verschlüsselung

Der öffentliche Verschlüsselungsalgorithmus **Advanced Encryption Standard (AES)** ist eines der meistgenutzten und sichersten Verschlüsselungsverfahren.

Seine Funktionsweise beruht auf einer Reihe von Byteersetzungen (Substitutionen), Verwürfelungen (Permutationen) und linearen Transformationen, die auf Datenblöcken von 16 Byte ausgeführt werden – daher die Bezeichnung Blockverschlüsselung. Diese Operationen werden mehrmals wiederholt, wobei in jeder dieser Runden ein individueller, aus dem Schlüssel berechneter Rundenschlüssel in die Berechnungen einfließt. Wird nur ein einziges Bit im Schlüssel oder im Datenblock verändert, entsteht ein komplett anderer Chiffreblock – ein Vorteil gegenüber klassischen Stromverschlüsselungen.

Die Bezeichnungen AES-128, AES-192 und AES-256 spezifizieren dabei die Länge des Schlüssels: 128, 192 bzw. 256 Bit. Bis heute ist für keine der AES-Varianten ein praktisch durchführbarer Angriff bekannt. AES ist daher der bevorzugte Verschlüsselungsstandard.




Zusätzliche Informationen zur AES-Verschlüsselung finden Sie auch unter <https://www.comparitech.com/blog/information-security/what-is-aes-encryption/>.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Encrypt: Verschlüsselt Eingabedaten mit dem AES-Verschlüsselungsalgorithmus. • Decrypt: Entschlüsselt Eingabedaten mit dem AES-Verschlüsselungsalgorithmus. • GenerateSharedKey: Erzeugt einen Shared Key zum Verschlüsseln und Entschlüsseln der Eingabedaten. • GenerateInitializationVector: Erzeugt einen zufälligen Initialisierungsvektor zum Verschlüsseln und Entschlüsseln der Eingabedaten.
-----------	---

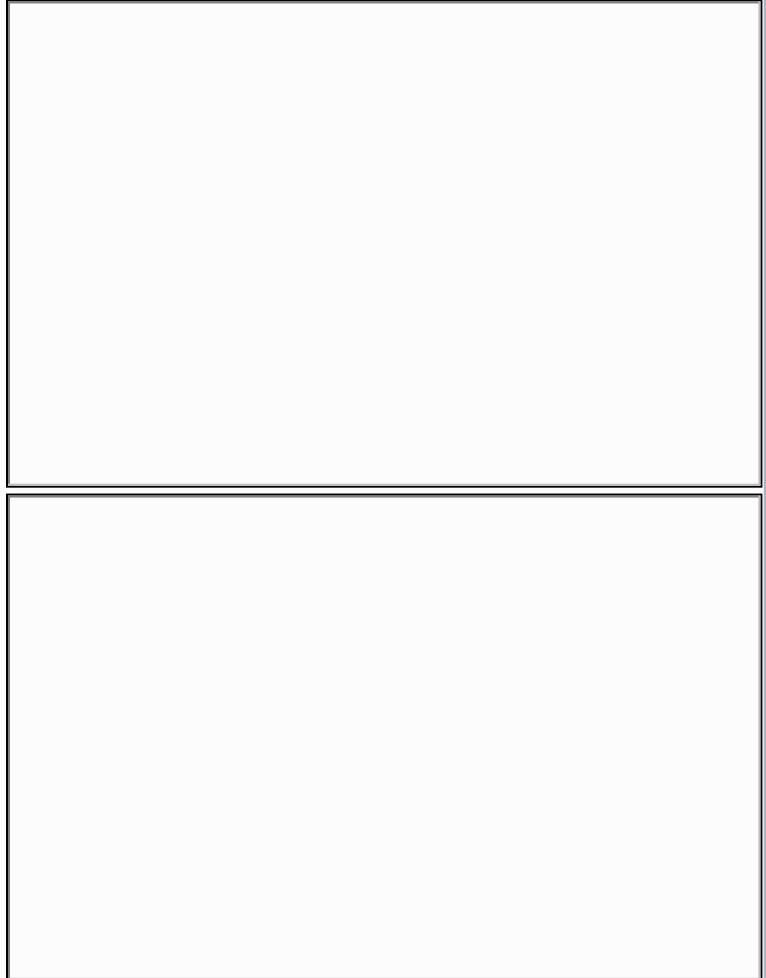
Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern)!</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.aes.AesEncryption</code> : Hauptklasse (Standard)</p>
password	<p><i>Erforderlich für die Operation <code>GenerateSharedKey</code>.</i> Passwort zum Erzeugen des Shared Keys.</p>

salt	<p><i>Erforderlich für die Operation <code>GenerateSharedKey</code>. Salt zum Erzeugen des Shared Keys (Base64-kodiert).</i></p> <div>  Salt bezeichnet in der Kryptografie eine zufällig gewählte Zeichenfolge, die an einen gegebenen Klartext vor dessen weiterer Verarbeitung (z. B. Eingabe in eine Hashfunktion) angehängt wird, um die Entropie der Eingabe zu erhöhen. Es wird häufig für die Speicherung und Übermittlung von Passwörtern benutzt, um die Informationssicherheit zu erhöhen. </div>
keyAlgorithm	<p><i>Erforderlich für die Operation <code>GenerateSharedKey</code>. Algorithmus zum Erzeugen des Shared Keys.</i></p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • DESede: Erstellt Secret Keys zur Verwendung mit dem DESede- (Triple DES)-Algorithmus. • PBESwith<digest>And<encryption> bzw. PBESwith<prf>And<encryption> : Secret Key Factory zur Verwendung für PKCS5-Passwortverschlüsselung, wobei <digest> ein Nachrichtendigest, <prf> eine Pseudozufallsfunktion und <encryption> ein Verschlüsselungsalgorithmus ist, z. B. <i>PBESwithMD5AndDES</i> (PKCS #5, Version 1.5), <i>PBESwithHmacSHA256AndAES_128</i> (siehe PKCS #5, Version 2.0). • PBKDF2With<prf>: Passwort-basierter Algorithmus zur Schlüsselableitung (siehe PKCS #5, Version 2.0), der die Pseudozufallsfunktion (<prf>) nutzt, z. B. <i>PBKDF2WithHmacSHA256</i>
keyLength	<p><i>Erforderlich für die Operation <code>GenerateSharedKey</code>. Schlüssellänge zum Erzeugen des Shared Keys.</i></p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • 128 • 196 • 256
keyIteration	<p><i>Erforderlich für die Operation <code>GenerateSharedKey</code>. Wiederholungen, die zum Erzeugen des Shared Keys verwendet werden soll.</i></p>

cipherAlgorithm	<p>Algorithmus zum Verschlüsseln und Entschlüsseln der Eingabedaten.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• AES/CBC/PKCS5PADDING• AES/CBC/NOPADDING• AES/ECB/PKCS5PADDING• AES/ECB/NOPADDING <div data-bbox="571 591 1433 1951"><p>📘 Zusatzinformationen zum CBC- und zum ECB-Modus</p><ul style="list-style-type: none">• Cipher Block Chaining Mode (CBC Mode) ist eine Betriebsart, in der Blockchiffren betrieben werden können. Vor dem Verschlüsseln eines Klartextblocks wird dieser zunächst mit dem im vorhergehenden Schritt erzeugten Geheimtextblock per XOR (exklusives Oder) verknüpft. Die Struktur der Verschlüsselung und der Entschlüsselung im CBC-Modus ist in nachfolgenden Abbildungen dargestellt:<div data-bbox="671 943 1433 1429"></div><div data-bbox="671 1467 1433 1951"></div></div>
-----------------	--

- **Electronic Code Book Mode (ECB Mode)** ist der einfachste Betriebsmodus für Blockverschlüsselungen, denn die Klartextblöcke werden unabhängig voneinander verschlüsselt. Dadurch ergeben bei gleichem Schlüssel, gleiche Klartextblöcke immer den gleichen Geheimtextblock. Dies ist auch der große Nachteil dieses Verfahrens, denn dadurch bleiben Klartextmuster erhalten. In nachfolgenden Abbildungen ist die Struktur des ECB bei der Ver- und Entschlüsselung dargestellt:



sharedKey

Shared Key zum Verschlüsseln und Entschlüsseln der Eingabedaten (Base64-kodiert).

initializationVector	<p>Initialisierungsvektor zum Verschlüsseln und Entschlüsseln der Eingabedaten; Funktioniert nur im CBC-Modus</p> <div> <p>i Beim Verschlüsseln von Nachrichten muss vermieden werden, dass gleiche Klartextblöcke immer wieder gleiche Geheimtextblöcke ergeben.</p> <p><i>Beispiel:</i> Ein förmlicher Brief fängt im Deutschen in der Regel mit „Sehr geehrte(r) Frau/Herr“ gefolgt vom Namen an. Aus diesem Wissen könnte ein Angreifer versuchen, Rückschlüsse auf den verwendeten Schlüssel zu ziehen. Um das zu vermeiden, wird der erste Klartextblock mit einem Initialisierungsvektor-XOR verknüpft. Da der Initialisierungsvektor zufällig erzeugt wurde, unterscheiden sich die entstehenden Geheimtexte auch dann, wenn die Klartexte mit identischen Daten beginnen.</p> </div>
prependInitializationVector	<p>Legt fest, ob der Initialisierungsvektor als Präfix für die verschlüsselten Daten vorangestellt werden soll.</p> <div> <p>i Der Initialisierungsvektor kann problemlos im Klartext an die verschlüsselten Daten übertragen werden, da Angreifer ohne den Schlüssel keinen Informationsgewinn aus dem Initialisierungsvektor erhalten. Er ist lediglich zur Verwässerung des Nachrichtenbeginns gedacht bevor die Nachricht verschlüsselt wird.</p> <p>Wenn der Initialisierungsvektor nicht übertragen wird, muss er auf anderem Wege erhalten werden, da er für die Entschlüsselung notwendig ist.</p> </div> <p>Mögliche Werte: true- / false-Angabe</p>
encoding	<i>Erforderlich für die Operation Decrypt.</i> Gibt das Encoding der entschlüsselten Daten an.
mimeType	<i>Erforderlich für die Operation Decrypt.</i> Gibt den Mime-Type der entschlüsselten Daten an.

Statuswerte

-1	Während der Ausführung der Operation kam es zu einem Fehler. Nähere Details finden Sie im Server-Log.
1	Die Operation wurde erfolgreich ausgeführt.

Input

Der Adapter erwartet je nach Operation verschiedene Input-Dokumente.

- Für die OperationEncrypt erwartet der Adapter ein beliebiges Input-Dokument, z. B. XML-, Text- oder Binär-Dateien.

Beispielhafter Input für Operation Encrypt

This is a top secret message and should be AES encrypted!

- Für die Operation Decrypt erwartet der Adapter AES-verschlüsselte Daten (Base64).

Beispielhafter Input für Operation Decrypt

dpI4dqEK7yiRcyz1HdLs9yxqNLIg6u8fqGR0JxADiCdHb/
6PGDqSxzID6VcuecxUP44x7bkHTggDpn8D0/bgKg==

Ausgabe

Der Adapter gibt je nach Operation verschiedene Ergebnis-Dokumente aus.

- Bei der Operation Encrypt gibt der Adapter die Eingabedaten verschlüsselt zurück (Base64-kodiert).

Beispielhafte Ausgabe für Operation Encrypt

dpI4dqEK7yiRcyz1HdLs9yxqNLIg6u8fqGR0JxADiCdHb/
6PGDqSxzID6VcuecxUP44x7bkHTggDpn8D0/bgKg==

- Bei der Operation Decrypt gibt der Adapter die verschlüsselten Eingabedaten entschlüsselt zurück.

Beispielhafte Ausgabe für Operation Decrypt

This is a top secret message and should be AES encrypted!

- Bei der Operation GenerateSharedKey gibt der Adapter den Shared Key mit den im Adapter definierten Key Parametern zurück.

Beispielhafte Ausgabe für Operation GenerateSharedKey

SfQYr8ULxmBIeHbZE28XkquKZHFx0qfsSTGBL+4WQDM=

- Bei der Operation GenerateInitializationVector gibt der Adapter einen zufälligen Initialisierungsvektor abhängig vom Cipher-Algorithmus aus.

Beispielhafte Ausgabe für Operation GenerateInitializationVector

JXAxGJ343MMjwLFi78z8xw==

Beispielszenario:

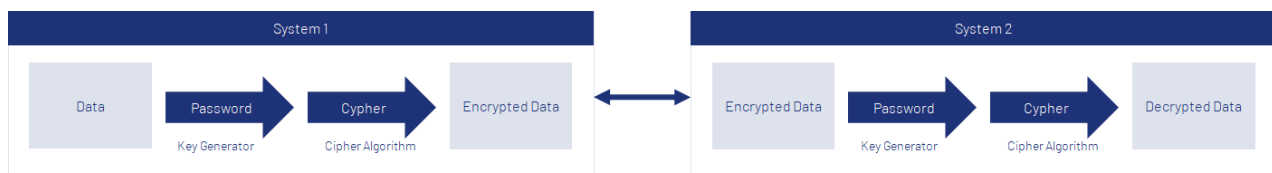
Ein typisches Szenario ist der Austausch von verschlüsselten Daten zwischen zwei Systemen. Eine Möglichkeit dies zu erreichen, ist ein geheimes Passwort zum Verschlüsseln und Entschlüsseln der Daten, das beide Systeme kennen. Dabei wird aus dem Passwort über den GenerateSharedKey-Algorithmus ein Schlüssel generiert. Der Schlüssel bereitet das Passwort kryptografisch auf und sorgt je nach gewähltem Algorithmus für eine stärkere Verschlüsselung. Je stärker der Schlüssel, desto schwerer können die verschlüsselten Daten durch Dritte entschlüsselt werden.

Alternativ zum Passwort kann auch der Schlüssel beiden Systemen bereitgestellt werden. Dies ist insofern weniger sicher, als das die zusätzliche Information, d.h. mit welchem Passwort der Schlüssel generiert wurde, nicht bekannt ist.

Mit dem Schlüssel und dem Cipher-Algorithmus werden die Daten für Dritte unkenntlich gemacht und können nur mit dem gleichen Schlüssel wieder kenntlich gemacht werden.

Die Verschlüsselten Daten können nun zum zweiten System transportiert werden, entweder über ein Netzwerk oder Hardware (USB-Stick). Auf diesem Weg kann kein Dritter die Daten verwenden.

Das zweite System erzeugt sich ebenfalls den Schlüssel mit den exakt gleichen Parametern wie das erste System. Zusammen mit dem wiederum gleichen Cipher-Algorithmus werden die Daten entschlüsselt und sind wieder verwendbar.



6.18 System Administration

6.18.1 JMeter Test Execution



JMeter Test Execution: Führt einen JMeter-Testplan aus und gibt ein Testergebnis als XML-Dokument zurück.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Execute</i> : JMeter-Testplan ausführen
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.jmeter.JMeterAdapter</i>: Hauptklasse (Standard)</p>
<i>jmeterHome</i>	<p>Absoluter Pfad zum JMeter-Installationsordner</p> <p>Mögliche Werte: Beliebige Pfadangabe (z.B. <i>c:\jakarta-jmeter-2.3.1\</i>)</p>
<i>proxyHost</i>	<p>IP-Adresse oder Host-Name eines Proxy-Servers für JMeter</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>192.168.1.1</i>)</p>
<i>proxyPort</i>	<p>Portnummer des Proxy-Servers für JMeter</p> <p>Mögliche Werte: Beliebige ganze Zahl</p>
<i>proxyUser</i>	<p>Benutzername des Proxy-Servers für JMeter</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>proxyPass</i>	<p>Passwort des Proxy-Servers für JMeter</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>outputFormat</i>	<p>Ausgabeformat für das Testergebnis-Dokument</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>XML</i>: XML-Dokument ausgeben (Standard) • <i>CSV</i>: "Comma separated Value"-Textdatei ausgeben
<i>timestamp</i>	<p>Zeitstempel-Angabe als Attribut ts im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ms</i>: Zeitstempel-Angabe in Millisekunden ausgeben (Standard) • <i>none</i>: Keinen Zeitstempel ausgeben

<i>dataType</i>	<p>Datentyp als Attribut dt im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>ja</i>: Datentyp ausgeben• <i>nein</i>: Keinen Datentyp ausgeben
<i>label</i>	<p>Beschriftung (Label) als Attribut lb im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>ja</i>: Beschriftung ausgeben• <i>nein</i>: Keine Beschriftung ausgeben (Standard)
<i>responseCode</i>	<p>Antwortcode als Attribut rc im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>ja</i>: Antwortcode ausgeben (Standard)• <i>nein</i>: Keinen Antwortcode ausgeben
<i>responseData</i>	<p>Antwortdaten als Attribut responseData im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>ja</i>: Antwortdaten ausgeben• <i>nein</i>: Keine Antwortdaten ausgeben (Standard)
<i>responseDataOnError</i>	<p>Antwortdaten bei Fehlern als Attribut responseDataOnError im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>ja</i>: Antwortdaten bei Fehlern ausgeben• <i>nein</i>: Keine Antwortdaten bei Fehlern ausgeben (Standard)
<i>responseMessage</i>	<p>Antwortnachricht als Attribut rm im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>ja</i>: Antwortnachricht ausgeben• <i>nein</i>: Keine Antwortnachricht ausgeben (Standard)

<i>successful</i>	<p>Erfolgsmeldung als Attribut <i>successful</i> im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Erfolgsmeldung ausgeben (Standard) • <i>nein</i>: Keine Erfolgsmeldung ausgeben
<i>time</i>	<p>Zeitangabe (in Millisekunden) als Attribut <i>t</i> im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Zeitangabe ausgeben (Standard) • <i>nein</i>: Keine Zeitangabe ausgeben
<i>subresults</i>	<p>Untergebnis als Attribut <i>subresults</i> im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Untergebnis ausgeben • <i>nein</i>: Kein Untergebnis ausgeben (Standard)
<i>assertions</i>	<p>Aussagen als Attribut <i>assertions</i> im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Aussagen ausgeben • <i>nein</i>: Keine Aussagen ausgeben (Standard)
<i>latency</i>	<p>Latenz als Attribut <i>l</i> im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Latenz ausgeben • <i>nein</i>: Keine Latenz ausgeben (Standard)
<i>samplerData</i>	<p>Sampler-Daten als Attribut <i>samplerData</i> im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Sampler-Daten ausgeben • <i>nein</i>: Keine Sampler-Daten ausgeben (Standard)

<i>responseHeaders</i>	<p>Antwort-Header als Attribut responseHeaders im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>ja</i>: Antwort-Header ausgeben• <i>nein</i>: Keine Antwort-Header ausgeben (Standard)
<i>requestHeaders</i>	<p>Request-Header als Attribut requestHeaders im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>ja</i>: Request-Header ausgeben• <i>nein</i>: Keine Request-Header ausgeben (Standard)
<i>encoding</i>	<p>Zeichenkodierung als Attribut c im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>ja</i>: Zeichenkodierung ausgeben• <i>nein</i>: Keine Zeichenkodierung ausgeben (Standard)
<i>bytes</i>	<p>Bytes als Attribut bytes im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>ja</i>: Bytes ausgeben• <i>nein</i>: Keine Bytes ausgeben (Standard)
<i>url</i>	<p>URL als Attribut url im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>ja</i>: URL ausgeben• <i>nein</i>: Keine URL ausgeben (Standard)
<i>threadCounts</i>	<p>Anzahl der Threads als Attribut threadCounts im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>ja</i>: Datentyp ausgeben• <i>nein</i>: Keinen Datentyp ausgeben (Standard)

<i>filename</i>	<p>Dateiname als Attribut filename im Testergebnis-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Dateiname ausgeben • <i>nein</i>: Keinen Dateiname ausgeben
<i>assertionResults</i>	<p>Ergebnis als Feld assertionResults im Testergebnis-Dokument ausgeben (nur wenn Parameter outputFormat auf CSV gesetzt ist)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>none</i>: Kein Ergebnis ausgeben (Standard) • <i>first</i>: Erstes Ergebnis ausgeben • <i>all</i>: Alle Ergebnisse ausgeben
<i>printFieldNames</i>	<p>In der ersten Zeile des Testergebnis-Dokuments die Feldnamen definieren (nur wenn Parameter outputFormat auf CSV gesetzt ist)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Feldnamen in der ersten Zeile ausgeben • <i>nein</i>: Keine Feldnamen in der ersten Zeile ausgeben (Standard)
<i>defaultDelimiter</i>	<p>Trennzeichen zwischen den einzelnen Datenfelder eines Datensatzes im Testergebnis-Dokument (nur wenn Parameter outputFormat auf CSV gesetzt ist)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebiges Zeichen außer Escape-Sequenzen wie <code>\t</code> • <code>,</code>: Komma (Standard)

6.18.2 JMX MBean



JMX MBean: Ermöglicht den Zugriff auf MBeans des lokalen JMX-Servers. Dabei können MBeans gesucht, MBean-Attribute gelesen und gesetzt, MBean-Operationen aufgerufen sowie MBean-Metadaten und Schlüsseleigenschaften separat ausgegeben werden. Dieser Adapter erwartet je nach gewählter Operation eine spezifische Input-XML-Struktur. Der Adapter gibt immer ein XML-Dokument aus, das in Abhängigkeit von der gewählten Operation Informationen über MBeans, die Ergebnisse der aufgerufenen MBean-Methoden oder ggf. Fehlermeldungen enthält.

Ehemalige Bezeichnung: *JMX MBean Manipulation Adapter*

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>List MBeans</i>: Alle verfügbaren <i>MBeans</i> mit Name, Domäne und Schlüsseleigenschaften auflisten (kein Input-XML-Dokument erforderlich) • <i>Find MBeans</i>: MBean anhand des Objektnamens suchen; erwartet ein spezifisches Input-XML-Dokument; siehe Operation <i>Find MBeans</i> • <i>Execute Operations</i>: JMX-Operationen ausführen; erwartet je nach gewünschter Operation ein spezifisches Input-XML-Dokument
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.jmx.JmxAdapter</i>: Hauptklasse (Standard)</p>
----------------	---

Input

Dieser Adapter erwartet für die Operationen *Find MBeans* und *Execute Operations* eine adapterspezifische Input-XML-Struktur, siehe unten. Diese Input-XML-Struktur kann über XSL-Mappings in Abhängigkeit von den verarbeiteten Daten dynamisch generiert werden.

6.18.2.1 Operation: Find MBeans

Über die Operation *Find MBeans* können Sie nach MBeans suchen. Alle MBeans, die dem Suchmuster entsprechen, werden ausgegeben. Der Adapter erwartet bei dieser Operation folgendes Input-XML-Dokument.

```
<Elementname>
  <Query objectname="Suchausdruck" attribut="..." />
</Elementname>
```

Das Input-XML-Dokument besitzt ein beliebiges Wurzelement und beliebig viele Elemente *<Query>*. Jedes Element *<Query>* enthält im Attribut *objectname* einen Suchausdruck für den MBean-Objektnamen, der nach dem Muster *Domänenname:Schlüssel* bzw.

Domänenname:Schlüsselname=Wert aufgebaut ist; siehe <http://java.sun.com/javase/6/docs/api/javax/management/ObjectName.html>.

Der Suchausdruck kann die Platzhalter (*?* und ***) enthalten. Alle zusätzlich angegebenen Attribute werden unverändert zum Ausgabe-Dokument durchgereicht.

Beachten Sie:

- Je nach verwendetem Applikations-Server können Namensauflösungen fehlerhaft implementiert sein. So löst der MBean-Server von *JBoss* Platzhalter ggf. nicht auf, wenn Sie in Schlüsselnamen und -werten stehen.
- Jeder Objektname muss durch einen Schlüssel qualifiziert werden. Wenn alle Instanzen des JMX-Objekts angesprochen werden sollen, ergänzen Sie den Objektnamen um `:*`.

Beispiel

Die folgende Suchanfrage sucht innerhalb der Domänen `de.softproject.X0` bis `de.softproject.Xz` nach allen MBeans, deren Schlüssel (*Key Property*) `name` mit `X4` beginnt, und reicht ein Attribut `sampleAttribute="1"` ans Ausgabedokument weiter.

```
<JMX>
  <Query objectname="de.softproject.X?:name=X4*" sampleAttribute="1" />
</JMX>
```

Der Adapter gibt beispielsweise folgendes XML-Dokument mit den Suchergebnissen aus:

```
<MBeans>
  <Query objectname="de.softproject.X?:name=X4*" sampleAttribute="value">
    <MBean objectname="de.softproject.X4:name=GlobalContextManager"
      domain="de.softproject.X4" keyProperties="name=GlobalContextManager">
      <KeyProperty name="name">GlobalContextManager</KeyProperty>
    </MBean>
    <MBean objectname="de.softproject.X4:name=X4Management"
      domain="de.softproject.X4" keyProperties="name=X4Management">
      <KeyProperty name="name">X4Management</KeyProperty>
    </MBean>
  </Query>
</MBeans>
```

6.18.2.2 Operation: Execute Operations

Über die Operation `Execute Operations` können Sie MBean-Operationen aufrufen, Attribute lesen und setzen sowie MBean-Metadaten und Schlüsseleigenschaften separat ausgeben. Der grundsätzliche Aufbau des Input-XML-Dokument ist identisch wie bei der Operation `Find MBeans`, siehe *Operation: Find MBeans*, jedoch können innerhalb des Elements `<Query>` unterschiedliche Elemente eingefügt werden. Diese Elemente repräsentieren Anweisungen, die im Folgenden erläutert werden.

Beachten Sie:

- Wie bei Operation Find MBeans kann auch bei Operation Execute Operations der Suchausdruck in Attribut objectname die Platzhalter ? und * enthalten.
- Jeder Objektname muss durch einen Schlüssel qualifiziert werden. Wenn alle Instanzen des JMX-Objekts angesprochen werden sollen, ergänzen Sie den Objektnamen um :*.

Metadaten auslesen

Das Kindelement `<GetMetadata/>` des Elements `<Query>` weist den Adapter an, sämtliche Metadaten der entsprechenden MBeans auszugeben:

- Im Attribut `class`: Nennt die Implementierungsklasse
- `<Description>`: Enthält einen Beschreibungstext der MBean
- `<Constructors>`: Listet Konstruktoren mit zugehörigen Operationen und Beschreibung auf
- `<Attributes>`: Führt alle Attribute der MBean auf
- `<Operations>`: Führt alle Operationen der MBean mit Name, Beschreibung, Datentypen, Parametern usw.
- `<Notifications>`: Führt ggf. Benachrichtigungen der MBean auf

```
<Query objectname="Suchausdruck" attribut="wert">
  <GetMetadata/>
</Query>
```

Schlüsseleigenschaften separat ausgeben

Das Kindelement `<ExpandKeyProperties/>` des Elements `<Query>` weist den Adapter an, alle Schlüsseleigenschaften in jeweils einem separaten Element `<Property>` auszugeben:

```
<Query objectname="Suchausdruck" attribut="wert">
  <ExpandKeyProperties/>
</Query>
```

Attributwerte auslesen

Das Kindelement `<GetAttributeValue/>` des Elements `<Query>` weist den Adapter an, Werte zu MBean-Attributen auszugeben, die in jeweils einem Kindelement `<Attribute>` im Attribut `name` angegeben werden:

```
<Query objectname="Suchausdruck" attribut="wert">
  <GetAttributeValue>
    <Attribute name="Attribut-Name" />
  </GetAttributeValue>
</Query>
```

Attributwerte auslesen

Das Kindelement `<SetAttributeValue/>` des Elements `<Query>` weist den Adapter an, MBean-Attributwerte zu setzen. Dabei wird für jedes MBean-Attribut in einem Element `<Attribute>` festgelegt:

- Im Attribut `name`: Name des zu setzenden MBean-Attributs
- Im Attribut `class`: Datentyp des zu setzenden MBean-Attributs
- Der Wert des MBean-Attributs wird im Elementinhalt angegeben.


```
<Query objectname="Suchausdruck" attribut="wert">
  <SetAttributeValue>
    <Attribute name="Attribut-Name" class="Datentyp"/></Attribute>
  </SetAttributeValue>
</Query>
```

MBean-Operationen aufrufen

Das Kindelement `<InvokeOperation/>` des Elements `<Query>` weist den Adapter an, eine MBean-Operation aufzurufen. Das Element `<InvokeOperation/>` muss im Attribut `name` den Namen der MBean-Operation enthalten. In beliebig vielen Kindelementen `<Parameter>` wird der jeweilige Parameterwert angegeben, im Attribut `class` wird der Datentyp des Parameterwerts definiert.

Folgende Datentyp-Angaben im Attribut `class` können verwendet werden:

- | | | |
|------------------|------------------------------|-------------------------------------|
| • <i>boolean</i> | • <i>double</i> | • <i>java.lang.Float</i> |
| • <i>char</i> | • <i>java.lang.Boolean</i> | • <i>java.lang.Double</i> |
| • <i>byte</i> | • <i>java.lang.Character</i> | • <i>java.lang.String</i> |
| • <i>short</i> | • <i>java.lang.Byte</i> | • <i>java.math.BigInteger</i> |
| • <i>int</i> | • <i>java.lang.Short</i> | • <i>java.math.BigDecimal</i> |
| • <i>long</i> | • <i>java.lang.Integer</i> | • <i>java.management.ObjectName</i> |
| • <i>float</i> | • <i>java.lang.Long</i> | |

 Da MBean-Parameter keine Namen besitzen, ist die Reihenfolge maßgeblich, in der diese notiert werden.

Ein Aufruf einer MBean-Operation erfolgt also durch eine nach folgendem Muster aufgebaute Input-XML-Struktur:

```
<Query objectname="Suchausdruck" attribut="wert">
  <InvokeOperation name="MBean-Operation">
    <Parameter class="Datentyp">Parameterwert</Parameter>
  </InvokeOperation>
</Query>
```

Beispiel

Im folgenden Input-XML-Dokument werden Metadaten abgefragt, Schlüsseleigenschaften gelistet, MBean-Attribute gelesen und gesetzt sowie eine MBean-Operation mit zwei Parametern ausgeführt:

```
<RootElement>
  <Query objectname="de.softproject.X?:name=X4Management">
    <GetMetadata/>
  </Query>
  <Query objectname="java.lang*:*>
    <ExpandKeyProperties/>
  </Query>
  <Query objectname="de.softproject.X4.WS*:*" text="enabling WebServices">
    <GetAttributeValue>
      <Attribute name="InService"/>
    </GetAttributeValue>
    <SetAttributeValue>
      <Attribute name="InService" class="boolean">true</Attribute>
    </SetAttributeValue>
  </Query>
  <Query objectname="de.softproject.X4:name=X4Management"
    description="Start a process">
    <InvokeOperation name="startProcessByName">
      <Parameter class="java.lang.String">1</Parameter>
      <Parameter class="java.lang.String">/Project/Folder/Process.wrf
    </Parameter>
    </InvokeOperation>
  </Query>
</RootElement>
```

6.18.3 JMX Number



JMX Number: Ermöglicht die Überwachung des aktuell ausgeführten technischen Prozesses in Form eines Prüfpunkts, der jeweils einer JMX-MBean entspricht. Wenn der Prüfpunkt erfolgreich durchlaufen wird, dann wird der Zeitstempel der MBean aktualisiert. Der zurückgegebene Adapter-Status hängt vom Wert der MBean-Eigenschaft `value` ab, der einen ganzzahligen Wert zwischen `0` und `899` annehmen kann. Über den jeweiligen numerischen Adapter-Status können mit diesem Adapter auch komplexere Fallunterscheidungen im technischen Prozess abgebildet werden.

Falls die entsprechende MBean noch nicht existiert, wird diese angelegt, mit einem aktuellen Zeitstempel versehen und die MBean-Eigenschaft `value` auf den Parameterwert von `startvalue` gesetzt. Zudem kann über den Adapter eine bestehende Prüfpunkt-MBean aus dem JMX-Register wieder entfernt werden.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>Check</i>: Zeitstempel aktualisieren und MBean-Wert <i>value</i> einlesen • <i>Unregister</i>: MBean aus dem JMX-Register entfernen
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.jmxtools.JmxNumberAdapter</i> : Hauptklasse (Standard)
<i>Objectname</i>	Name des bearbeiteten Objekts (Prüfpunkt) Mögliche Werte: <ul style="list-style-type: none"> • <i>de.softproject.x4.jmxnumber: pid=%PID%, process=%CURRENT_PROCESS%, userid=%USER_ID%</i>: aktuelle Prozess-Instanz (Standard) • Beliebiger gültiger JMX-Objektname, siehe http://java.sun.com/javase/6/docs/api/javax/management/ObjectName.html)
<i>Startvalue</i>	Standardwert für die MBean-Eigenschaft <i>value</i> Mögliche Werte: <ul style="list-style-type: none"> • Beliebige ganze Zahl zwischen 0 und 899 • 1: Wert 1 zur Ausgabe von Status 1 (Standard)
<i>Persistent</i>	Wert der JMX-Bean persistieren Mögliche Werte: <ul style="list-style-type: none"> • <i>Ja</i>: Wert persistieren • <i>Nein</i>: Nicht persistieren (Standard)

Input

Der Funktions-Adapter nimmt beliebige Input-Dokumente entgegen und gibt diese unverändert aus.

Ausgabe

Das ausgegebene Dokument entspricht dem Input-Dokument.

Statuswerte

<i>0...899</i>	MBean-Eigenschaft <i>Value</i> besitzt einen Wert zwischen 0 und 899
----------------	--

-1	Fehler beim JMX-Aufruf (z. B. inkompatible MBean)
----	---

6.18.4 JMX Switch



JMX Switch: Ermöglicht die Überwachung des aktuell ausgeführten X4-Prozesses in Form eines Prüfpunkts, der jeweils einer JMX-MBean entspricht.

Wenn der Prüfpunkt erfolgreich durchlaufen wird, dann wird der Zeitstempel der MBean aktualisiert. Der zurückgegebene Adapter-Status hängt vom Wert der MBean-Eigenschaft `value` ab. Falls die entsprechende MBean noch nicht existiert, wird diese angelegt, mit einem aktuellen Zeitstempel versehen und die MBean-Eigenschaft `value` auf den Parameterwert von `startvalue` gesetzt. Zudem kann über den Adapter eine bestehende Prüfpunkt-MBean aus dem JMX-Register wieder entfernt werden.

Ehemalige Bezeichnung bis Version 4.5.1: *JMX Switch Adapter*

i Mit dem Adapter *JMX Number* lässt sich statt eines Booleschen ein ganzzahliger Wert von 0 bis einschließlich 899 setzen. Über den jeweiligen numerischen Adapter-Status können mit diesem Adapter, auch komplexere Fallunterscheidungen im X4-Prozess abgebildet werden, siehe [JMX Number](#).

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt
	Mögliche Werte: <ul style="list-style-type: none"> • <i>check</i>: Zeitstempel aktualisieren und MBean-Wert <code>value</code> einlesen • <i>unregister</i>: MBean aus dem JMX-Register entfernen

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.jmxtools.JmxSwitchAdapter</i> : Hauptklasse (Standard)
<i>Objectname</i>	Name des bearbeiteten Objekts (Prüfpunkt) Mögliche Werte: <ul style="list-style-type: none"> • <i>de.softproject.x4.jmxswitch:pid=%PID%,process=%CURRENT_PROCESS%,userid=%USER_ID%</i>: aktuelle Prozess-Instanz (Standard) • Beliebiger gültiger JMX-Objektnamen; siehe http://java.sun.com/javase/6/docs/api/javax/management/ObjectName.html

<i>Startvalue</i>	Standardwert für die MBean-Eigenschaft Value Mögliche Werte: <ul style="list-style-type: none"> • <i>Ja</i>: True (Standard) • <i>Nein</i>: False
<i>Persistent</i>	Wert der JMX-Bean persistieren Mögliche Werte: <ul style="list-style-type: none"> • <i>Ja</i>: Wert persistieren • <i>Nein</i>: Nicht persistieren (Standard)

Input

Der Funktions-Adapter nimmt beliebige Input-Dokumente entgegen und gibt diese unverändert aus.

Ausgabe

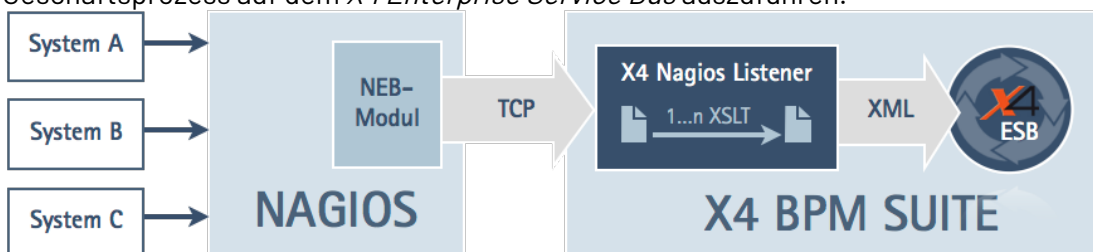
Das ausgegebene Dokument entspricht dem Input-Dokument.

Statuswerte

0	MBean-Eigenschaft Value besitzt den Wert <i>false</i>
1	MBean-Eigenschaft Value besitzt den Wert <i>true</i>
-1	Fehler beim JMX-Aufruf (z. B. inkompatible MBean)

6.18.5 Nagios Listener

Mit dem *Nagios Listener* lässt sich das Infrastruktur-Monitoring-Werkzeug *Nagios* an den *X4 Enterprise Service Bus* anbinden, um bei Überwachungs-Ereignissen einen entsprechenden Geschäftsprozess auf dem *X4 Enterprise Service Bus* auszuführen.



Features

- Verarbeitung beliebiger Überwachungsereignisse aus *Nagios*
- Ausgabe von strukturierten Daten als XML-Dokument
- Filterung und Aufbereitung der Ereignisdaten über XSL-Transformationsketten ohne wichtige Informationen über den Zustand des überwachten Systems zu verlieren

6.18.5.1 Aufbau und Funktionsweise des X4 Nagios Listeners

Im folgenden Abschnitt werden der Aufbau und die Funktionsweise des Prozess-Starters *X4 Nagios Listener* erläutert.

6.18.5.1.1 Aufbau

Die Komponente *X4 Nagios Listener* ist Bestandteil des X4 Servers. Die entsprechende Konfigurationsdatei befindet sich unter `<X4DB>/0/NagiosListener.properties`.

6.18.5.1.2 Funktionsweise


Nagios sendet über dessen NEB-Modul (NDOMOD Event Broker Module) Ereignisse an ein *TCP Socket*. Der *X4 Nagios Listener* "horcht" auf einem frei wählbaren Port und nimmt die von *Nagios* bereitgestellten Rohdaten zur weiteren Verarbeitung entgegen.

Die von *Nagios* bereitgestellten Rohdaten lassen sich direkt im *X4 Nagios Listener* via XSLT filtern bzw. transformieren. Abschließend startet der *X4 Nagios Listener* einen X4-Prozess mit dem Ergebnis der XSL-Transformationskette als Input (sinnvollerweise XML).

Den Prozesspfad im X4 Repository lässt sich ebenso wie der Port und die auszuführenden XSL-Transformationen in der Datei `<X4DB>/0/NagiosListener.properties` konfigurieren, siehe [Konfiguration des X4 Nagios Listeners](#).

6.18.5.1.3 Datentransformation

Da *Nagios* sehr umfangreiche, teils redundante Daten liefert, lassen sich direkt im *X4 Nagios Listener* relevante Informationen über XSL-Stylesheets herausfiltern und aufbereiten.

 Enthält das erzeugte XML-Dokument nach der XSL-Transformation ein Wurzel-Element ohne Kindelemente, wird der hinterlegte X4-Prozess nicht gestartet.

Folgende Stylesheets sind bereits im Lieferumfang des *X4 Nagios Listeners* verfügbar:

- `ReLabel.xsl` benennt im erzeugten XML-Dokument alle für die Verarbeitung relevanten Elemente in verständliche, natürlichsprachliche Bezeichnungen um.
- `RemoveUnderscored.xsl` filtert aus dem gesendeten Ereignis die Daten heraus, die nicht durch `ReLabel.xsl` umbenannt wurden.

Sie können beliebig viele XSL-Stylesheets zur Datentransformation und -aufbereitung definieren, die bei jedem Überwachungs-Ereignis nacheinander ausgeführt werden, siehe [Konfiguration des X4 Nagios Listeners](#).

6.18.5.1.4 Beispiel

Ohne XSL-Transformationen gibt der *X4 Nagios Listener* z. B. folgendes XML-Dokument aus:

```

<NagiosEvent typ="213">
  <_1>1202</_1>
  <_2>0</_2>
  <_3>0</_3>
  <_4>1330418120.146854</_4>
  <_53>localhost</_53>
  <_114>Current Users</_114>
  <_95>USERS OK - 1 users currently logged in</_95>
  <_125></_125>
  <_99>users=1;20;50;0</_99>
  <_27>0</_27>
  <_51>1</_51>
  <_115>1</_115>
  <_25>1</_25>
  <_76>5</_76>
  <_61>1330418116</_61>
  <_83>1330418176</_83>
  <_12>0</_12>
  <_63>1316420128</_63>
  <_57>1316420128</_57>
  <_56>0</_56>
  <_66>1330418116</_66>
  <_70>0</_70>
  <_67>0</_67>
  <_64>0</_64>
  <_121>1</_121>
  <_62>0</_62>
  <_84>0</_84>
  <_85>0</_85>
  <_88>1</_88>
  <_101>0</_101>
  <_7>0</_7>
  <_26>0</_26>
  <_97>1</_97>
  <_38>1</_38>
  <_9>1</_9>
  <_47>1</_47>
  <_54>0</_54>
  <_98>0.000000</_98>
  <_71>0.10200</_71>
  <_42>0.03403</_42>
  <_113>0</_113>
  <_45>1</_45>
  <_103>1</_103>
  <_93>1</_93>
  <_80>0</_80>
  <_37></_37>
  <_11>check_local_users!20!50</_11>
  <_86>1.000000</_86>
  <_109>1.000000</_109>
  <_209>24x7</_209>
  <_999/>
</NagiosEvent>

```

Nach der Transformation durch die mitgelieferten XSL-Stylesheets wird folgendes XML-Dokument ausgegeben und an den zu startenden X4-Prozess als Input übergeben:

```
<NagiosEvent typ="213">
  <Instance>localhost</Instance>
  <Check>Current Users</Check>
  <Result>USERS OK - 1 users currently logged in</Result>
  <Timestamp>1330418536</Timestamp>
  <Period>24x7</Period>
</NagiosEvent>
```

6.18.5.2 Konfiguration des X4 Nagios Listeners

Konfigurationsdateien des *X4 Nagios Listeners* und wie diese angepasst werden können

6.18.5.2.1 Konfigurationsdateien

Der *X4 Nagios Listener* lässt sich über die Datei `<x4DB>/NagiosListener.properties` konfigurieren.


6.18.5.2.2 X4 Nagios Listener konfigurieren

1. Listener aktiviere bzw. deaktivieren

In Zeile 7 von `NagiosListener.properties` kann der *X4 Nagios Listener* aktiviert oder deaktiviert werden. Bei einer Änderung muss die Konfiguration neu geladen werden.

2. Port konfigurieren

In Zeile 13 die Port-Nummer hinter `port` = angeben.

 Die Portnummer muss der Portnummer aus der *NDOMOD Event Broker Module* Konfigurationsdatei `ndomod.cfg` entsprechen.

3. Namen und Pfad des aufzurufenden X4-Prozesses in `NagiosListener.properties` konfigurieren

- In Zeile 21 das entsprechende Benutzer-Repository hinter `x4user` = angeben, in dem der X4-Prozess liegt.
Beispiel: 1 für das Standard-Benutzer-Repository `<X4 Server>/X4DB/1/`
- In Zeile 27 den Pfad des X4-Prozesses innerhalb des Benutzer-Repositorys angeben.
Beispiel: Project/Folder/Process.wrf

4. XSL-Stylesheets für die Datentransformation bzw. Filterung definieren

- Für jedes zusätzliche XSL-Stylesheet eine Zeile hinzufügen, die nach folgendem Muster aufgebaut ist: `xsl_ Nummer = X4DB-URL zur XSL-Stylesheet-Datei`.
Über die Nummer definieren Sie die Reihenfolge, in der die verschiedenen XSL-Stylesheets hintereinander ausgeführt werden können.
Beispiel: xsl_3 = x4db:/1/Example/MyTransformation.xsl
- Nicht benötigte XSL-Stylesheets können Sie durch Auskommentieren (über ein vorangestelltes `!` oder `#`) ausschließen.
Wenn sämtliche XSL-Stylesheet-Referenzen auskommentiert oder gelöscht wurden,

gibt der *X4 Nagios Listener* ein ungefiltertes XML-Dokument mit allen Rohdaten aus.
Beispiel: Wenn das zweite Standard-XSL-Stylesheet nicht angewendet werden soll: `#
 xsl_2 = x4db:/0/Nagios/RemoveUnderscored.xsl` notieren.

6.18.5.3 Nagios Listener starten und beenden

Der *X4 Nagios Listener* lässt sich über die Konfigurationsdatei aktivieren und deaktivieren. Beim Starten des X4 Servers wird die Konfiguration eingelesen und der *X4 Nagios Listener* ggf. direkt gestartet.

Änderungen können im laufenden Betrieb durch die Methode *reload* in der

MBean `de.softproject.X4.NagiosStarter` geladen werden. So kann der *X4 Nagios Listener* auch gestartet oder beendet werden.

6.18.6 SNMP Management



SNMP Management: Verwaltet Geräte in einem IP-Netzwerk über das *Simple Network Management Protocol* (SNMP) in Version 1, 2c oder 3 (mit automatisch gewählter Sicherheitsebene). Der Adapter fordert einzelne oder mehrere Management-Datensätze (MIB-Objekte) von Agenten an, setzt Werte von Management-Datensätzen und versendet Trap-Nachrichten an Agenten. Dieser Adapter erwartet ein Input-XML-Dokument mit einer spezifischen Struktur, die MIB-Objekt-IDs und SNMP-Parameter enthält, und gibt ein XML-Dokument aus.

i Wenn der Agent bei der Überwachung des Systems einen Fehler erkennt, kann er diesen mit Hilfe eines Trap-Paketes unaufgefordert an den *SNMP Management Adapter* melden. Diese Pakete werden nicht vom Adapter bestätigt. Der Agent kann damit nicht feststellen, ob der Trap beim Adapter angekommen ist.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>GET</i>: Einzelnen Management-Datensatz (MIB-Objekt) anfordern. Der Agent antwortet mit einem Response-Paket, das entweder die angeforderten Daten oder eine Fehlermeldung enthält. • <i>GETNEXT</i>: Nachfolgenden Management-Datensatz (MIB-Objekt) anfordern (um Tabellen zu durchlaufen) • <i>GETBULK</i>: Mehrere Management-Datensätze (MIB-Objekte) gleichzeitig anfordern • <i>SET</i>: Einen oder mehrere Management-Datensätze (MIB-Objekte) eines Geräts verändern oder Aktionen auszulösen. Der Agent bestätigt die Übernahme der Werte mit einem Response-Paket. • <i>SENDTRAP</i>: Trap-Nachricht zu einem Ereignis unaufgefordert an den Manager per UDP von Port 161 senden
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.snmp.SnmpAdapter</i>: Hauptklasse (Standard)</p>
<i>host</i>	<p>IP-Adresse oder Host-Name der Netzwerk-Management-Station (NMS)</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>127.0.0.1</i>)</p>
<i>port</i>	<p>UDP-Portnummer (nur für SNMP in Version 2c und 3 änderbar)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>161</i>: Standard-Port, auf dem SNMP-Agenten horchen für die Operationen <i>GET</i>, <i>GETNEXT</i>, <i>GETBULK</i> und <i>SET</i> (Standard) • <i>162</i>: Port, auf dem die Netzwerk-Management-Station standardmäßig horcht (für die Operation <i>SENDTRAP</i>)
<i>version</i>	<p>Verwendete SNMP-Version</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>version1</i>: Version 1 verwenden • <i>version2c</i>: Version 2c verwenden • <i>version3</i>: Version 3 verwenden
<i>timeout</i>	<p>Zeitbeschränkung für den Verbindungsaufbau in Millisekunden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl • <i>3</i>: Zeitbeschränkung von 3 ms (Standard)
<i>retries</i>	<p>Anzahl der Wiederholung von Verbindungsversuchen, falls die Verbindung nicht direkt aufgebaut werden kann</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl • <i>0</i>: Kein zusätzlicher Verbindungsversuch (Standard)

<i>walk</i>	<p>Sämtliche Anfragen im Input-XML-Dokument mit der <i>Walk</i>-Operation ausführen, bis alle Instanzen im OID-Teilbaum abgefragt wurden (wenn die Eigenschaft <i>Operation</i> auf <i>GETNEXT</i> oder <i>GETBULK</i> gesetzt ist)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: <i>GETNEXT</i>- und <i>GETBULK</i>-Anfragen mit der <i>SNMP Walk</i>-Operation ausführen • <i>false</i>: Keine <i>Walk</i>-Operation ausführen, wenn nicht anders im Input-XML-Dokument definiert (Standard)
<i>contextEngineId</i>	<p>SNMP-Entity, die die Anfrage ausführen soll (wenn Parameter <i>version</i> auf <i>version3</i> gesetzt ist)</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>community</i>	<p>Name der Community der SNMP-Agenten (wenn der Parameter <i>version</i> auf <i>version1</i> oder <i>version2c</i> gesetzt ist)</p> <p>Mögliche Werte: Beliebige Zeichenkette (z. B. <i>public</i>)</p>
<i>securityname</i>	<p>Sicherheitsnamen zur Authentifizierung von Nachrichten über SNMP Version 3 (wenn Parameter <i>version</i> auf <i>version3</i> gesetzt ist)</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>authenticationprotocol</i>	<p>Verwendetes Verfahren zur Authentifizierung einer Nachricht über SNMP Version 3 (wenn Parameter <i>version</i> auf <i>version3</i> gesetzt ist)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>MD5</i>: <i>MD5</i>-Prüfsumme erstellen • <i>SHA</i>: <i>Secure Hash</i>-Code erstellen
<i>authenticationpass</i>	<p>Passwort zur Authentifizierung über SNMP Version 3 auf Sicherheitsebene <i>authNoPriv</i> (wenn Parameter <i>version</i> auf <i>version3</i> gesetzt ist)</p> <p>Mögliche Werte: Beliebige Zeichenkette mit mindestens 8 Zeichen Länge</p>

<i>privacyprotocol</i>	<p>Verschlüsselungs-Algorithmus für Nachrichten mit SNMP Version 3 (wenn Parameter <i>version</i> auf <i>version3</i> gesetzt ist)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>DES</i>: <i>Data Encryption Standard</i>-Algorithmus verwenden • <i>AES</i>: <i>Advanced Encryption Standard</i>-Algorithmus verwenden • <i>AES192</i>: <i>Advanced Encryption Standard</i>-Algorithmus mit 192 Bit Schlüssellänge verwenden • <i>AES256</i>: <i>Advanced Encryption Standard</i>-Algorithmus 256 Bit Schlüssellänge verwenden
<i>privacypass</i>	<p>Passwort für die Verschlüsselung von Nachrichten mit SNMP Version 3 auf Sicherheitsebene <i>authPriv</i> (wenn Parameter <i>version</i> auf <i>version3</i> gesetzt ist)</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>

Input

Dieser Adapter erwartet eine adapterspezifische Input-XML-Struktur, wobei jedes Element `<pdu/>` für ein Netzwerkgerät steht.

```
<root>
  <pdu oid="MIB-Objekt-ID" timeout="Zeitbeschränkung in ms"
    retries="Anzahl zusätzlicher Verbindungsversuche" />
</root>
```

Ausgabe


Dieser Adapter gibt ein UTF-8-kodiertes XML-Dokument mit folgender Struktur aus.

```
<root>
  <pdu oid="MIB-Objekt-ID">
    <![CDATA[ Ausgegeben Daten des MIB-Objekts ]]>
  </pdu>
</root>
```

6.18.7 Windows Event Reader



Windows Event Reader: Liest das Event Log (Ereignisprotokoll) von Microsoft Windows und gibt sämtliche oder nach Filterkriterien gewählte Ereignis-Einträge in einem XML-Dokument aus. Ehemalige Bezeichnung: *Windows Event Logging Adapter*

-  Folgende Bedingungen müssen erfüllt sein, um diesen Adapter verwenden zu können:
- Der Windows-Computer, von dem das Ereignisprotokoll gelesen werden soll, muss in derselben Domäne liegen wie der *X4 Server*.
 - Der *X4 Server* muss auf einem Windows-Computer laufen, der Domänen-Administratoren-Rechte besitzt.
 - Eine JNI-Lizenz wird benötigt.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt
	Mögliche Werte: <i>Execute</i> : Adapter ausführen

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.windowseventlog.WindowsEventLogAdapter</i> : Hauptklasse (Standard)
<i>log</i>	Windows-Protokoll/Log, dessen Ereignisse angezeigt werden sollen Mögliche Werte: <ul style="list-style-type: none"> • Beliebiges Windows-Protokoll/Log (z.B. <i>Application</i>) • <i>System</i>: Systemereignisse ausgeben (Standard)
<i>from</i>	Startdatum zur Beschränkung auf einen Zeitraum Mögliche Werte: Beliebiges Datum und Uhrzeit im Format YYYY-MM-DDThh:mm:ss (z.B. <i>2012-03-31T13:45:03</i>)
<i>to</i>	Enddatum zur Beschränkung auf einen Zeitraum Mögliche Werte: Beliebiges Datum und Uhrzeit im Format YYYY-MM-DDThh:mm:ss (z.B. <i>2012-03-31T13:45:03</i>)
<i>type</i>	Nach einem Ereignis-Typ (Ebene) filtern Mögliche Werte: Beliebiger Ereignis-Typ (z.B. <i>Error</i>)
<i>source</i>	Nach einer Ereignis-Quelle (dies kann beispielsweise eine Anwendung sein) filtern Mögliche Werte: Beliebige Zeichenkette (z.B. <i>PrintSpooler</i>)

<i>category</i>	Nach einer Ereignis-Kategorie filtern Mögliche Werte: Beliebige Zeichenkette
<i>eventId</i>	Nach einer Ereignis-ID filtern Mögliche Werte: Beliebige ganze Zahl
<i>user</i>	Nach einem Benutzer filtern Mögliche Werte: Beliebige Zeichenkette (z.B. <i>SYSTEM</i>)
<i>computer</i>	Name des Computers, von dem das Ereignisprotokoll gelesen werden soll Mögliche Werte: <ul style="list-style-type: none"> • Beliebiger Computer-Name oder IP-Adresse • <i>127.0.0.1</i>: Momentan genutztes System, d.h. localhost (Standard)

Ausgabe

Dieser Funktions-Adapter gibt ein UTF-8-kodiertes XML-Dokument mit folgender Struktur aus. Für jedes Ereignis wird ein XML-Element `<log/>` erzeugt, das eine Reihe von Informationen sowie die Ereignis-Beschreibung zum Ereignis enthält:

```
<logs>
  <log type="Ereignistyp" date="Datum und Uhrzeit"
    source="Quelle" category="Kategorie" event="Ereignis-ID" user="Benutzer"
    computer="Computername">Ereignis-Beschreibung</log>
</logs>
```

6.18.8 Windows Services Explorer



Windows Services Explorer: Erzeugt eine Liste von Windows-Diensten und deren Status und gibt diese in einem XML-Dokument aus.

- i** Folgende Voraussetzungen müssen erfüllt sein, um diesen Adapter verwenden zu können:
- Der Windows-Computer, von dem Dienst-Statusinformationen gelesen werden sollen, muss in derselben Domäne liegen wie der *X4 Server*.
 - Der *X4 Server* muss auf einem Windows-Computer laufen, der Domänen-Administratoren-Rechte besitzt.
 - Eine JNI-Lizenz wird benötigt.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Execute</i> : Adapter ausführen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.windowsservice.WindowsServiceAdapter</i> : Hauptklasse (Standard)
<i>host</i>	Name des Computers, der den Windows-Dienst ausführt Mögliche Werte: <ul style="list-style-type: none"> • Beliebiger Computer-Name oder IP-Adresse • <i>127.0.0.1</i>: Momentan genutztes System, d.h. localhost (Standard)
<i>service</i>	Nach einem bestimmten Windows-Dienst filtern Mögliche Werte: Beliebige Zeichenkette

Ausgabe

Dieser Adapter gibt ein UTF-8-kodiertes XML-Dokument mit folgender Struktur aus. Für jeden Dienst wird ein XML-Element `<service/>` erzeugt, das Informationen zum Windows-Dienst sowie dessen Status enthält:

```
<services>
  <service name="Dienst-Name" displayName="Anzeigename des Diensts"
    startupType="Starttyp" logOnAs="Benutzer/Anmeldetyp">
    <!--Status des Dienstes--></service>
</services>
```

6.19 Web Services

6.19.1 HTTP(S) Output Parameter



HTTP(S) Output Parameter: Setzt in der HTTP-Response eines X4-Prozesses, der per ReST Service aufgerufen wurde, zusätzliche HTTP-Header-Informationen oder überschreibt bestehende HTTP-Header-Informationen gleichen Namens.

ⓘ Beachten Sie:

- Beim Bereitstellen von ReSTful Webservices mit dem Output-Handler *HTTP-MetaXML* wird vom X4-Prozess als Response eine spezifische XML-Struktur u.a. mit HTTP-Header-Informationen erwartet, siehe [ReSTful Webservices: Erwartete Output-XML-Struktur](#). In dieser XML-Struktur definierte HTTP-Header-Informationen überschreiben die mit dem HTTP(S) Output Parameter Adapter gesetzten Header-Informationen!
- Wenn Sie mit diesem Adapter HTTP-Headerwerte setzen, werden mit den Werten von gleichnamigen HTTP-Headern überschrieben, die im Prozess-Output stehen, wenn dieser der Output-Struktur für *HTTP-MetaXML* entspricht.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: SetOutputParameter: Output-Parameter setzen
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.outputParameter.OutputParameter</i> : Hauptklasse (Standard)
----------------	---

Input

Als Input erwartet der Adapter ein XML-Dokument mit einem Wurzel-Element `<Output>` und beliebig vielen Kindelementen, die jeweils eine HTTP-Header-Information definieren gemäß folgender Struktur:

```
<Output>
  <HTTP-Header-Name><!-- zugehöriger Wert des HTTP-Headers --></HTTP-Header-Name>
  <!-- beliebig viele weitere Elemente für weitere HTTP-Header-Informationen -->
</Output>
```

Beispiel

Im folgenden Beispiel-Input-Dokument wird die HTTP-Header-Information X-Token gesetzt oder – falls bereits vorhanden – mit dem Wert 123456789 überschrieben:

```
<Output>
  <X-Token>123456789</X-Token>
</Output>
```

6.19.2 JSON

6.19.2.1 JSON Converter 2.0



JSON Converter 2.0: Konvertiert ein Textdokument, das strukturierte Daten in der *JavaScript Object Notation (JSON)* enthält, in ein XML-Dokument, und konvertiert umgekehrt ein XML-Dokument in ein UTF-8-kodiertes JSON-Dokument. Dabei verwendet der Adapter ein neues XML-Format, das eine bessere Kontrolle des generierten JSON-Dokuments erlaubt.

Dieser Adapter ersetzt den veralteten [JSON Converter](#).

Eigenschaften (Properties)

Operation	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <ul style="list-style-type: none">• JSON2XML: JSON-Dokument in XML konvertieren• XML2JSON: XML-Dokument in JSON konvertieren
-----------	---

Parameter

Adapter	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.json.Json2Converter</i> : Hauptklasse (Standard)
allowBareObjectKeys	Angaben zum Umgang mit Object Keys, die nicht in Anführungszeichen stehen Mögliche Werte: <ul style="list-style-type: none">• <code>true</code>: erlaubt Object Keys, die nicht in Anführungszeichen stehen• <code>false</code>: erlaubt keine Object Keys, die nicht in Anführungszeichen stehen
escapeStringInXml	Angaben zum Umgang mit Nicht-XML-Zeichen Mögliche Werte: <ul style="list-style-type: none">• <code>true</code>: Nicht-XML-Zeichen escapen. Zeichen, die in XML verboten sind, werden durch <code>\u????</code> ersetzt, wobei die Fragezeichen den Hexadezimalen Ziffern des Unicode-Zeichens entsprechen• <code>false</code>: Nicht-XML-Zeichen filtern. Zeichen, die in XML verboten sind werden durch das Unicode-Zeichen <code>uFFFD</code> (Replacement Character) ersetzt

emptyStringAsNull	<p>Leere Zeichenketten bei Zahlen und Boolean-Werten als null ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • true: Leere Zeichenketten als null ausgeben, z. B. <code><Value type="Boolean"></Value></code> wird als null ausgegeben • false: Leere Zeichenketten nicht als null ausgeben, z. B. <code><Value type="Boolean"></Value></code> erzeugt einen Fehler <div> <p>i <code><Value type="String"></Value></code> erzeugt immer "".</p> </div>
-------------------	--

i Bei der Operation XML2JSON werden für numerische und boolesche Werte Whitespaces am Anfang und Ende des Wertes ignoriert.

Beispiel

Folgende Beispiele zeigen JSON-Ausdrücke und ihre jeweiligen Entsprechungen in XML.

JSON	XML
<code>null</code>	<code><Value type="Null"/></code>
<code>true</code>	<code><Value type="Boolean">true</Value></code>
<code>false</code>	<code><Value type="Boolean">>false</Value></code>
<code>1</code>	<code><Value type="Number">1</Value></code>
<code>1.0</code>	<code><Value type="Number">1.0</Value></code>
<code>1.2e3</code>	<code><Value type="Number">1200</Value></code>
<code>"Hallo Welt!"</code>	<code><Value type="String">Hallo Welt!</Value></code>

JSON	XML
<div>"\u0000"</div>	<div>escapeStringInXml = true</div> <div><Value type="String">\u0000</Value></div> <div>escapeStringInXml = false</div> <div><Value type="String">&#xFFFD;</Value></div>
<div>{}</div>	<div><Object/></div>
<div>[]</div>	<div><Array/></div>
<div>{ "key" : "value", "a number" : 42, "true" : false, "emptyList" : [], "emptyObject" : {} }</div>	<div><Object> <Value name="key" type="String">value</Value> <Value name="a number" type="Number">42</Value> <Value name="true" type="Boolean">>false</Value> <Array name="emptyList" /> <Object name="emptyObject" /> </Object></div>
<div>[1,2,3]</div>	<div><Array> <Value type="Number">1</Value> <Value type="Number">2</Value> <Value type="Number">3</Value> </Array></div>

JSON	XML
<pre>{ "title" : "A List", "list" : [{ "id" : 1, "content" : "first" }, { "id" : 2, "content" : "second" }, { "id" : 3, "content" : "third" }] }</pre>	<pre><Object> <Value name="title" type="String">A List</Value> <Array name="list"> <Object> <Value name="id" type="Number">1 </Value> <Value name="content" type="Stri ng">first</Value> </Object> <Object> <Value name="id" type="Number">2 </Value> <Value name="content" type="Stri ng">second</Value> </Object> <Object> <Value name="id" type="Number">3 </Value> <Value name="content" type="Stri ng">third</Value> </Object> </Array> </Object></pre>

6.19.2.2 JSON Converter (Veraltet)

JSON Converter : Konvertiert ein Textdokument, das strukturierte Daten in der *JavaScript Object Notation (JSON)* enthält, in ein XML-Dokument mit dem Wurzel-Element `<JSON>`, und konvertiert umgekehrt ein XML-Dokument in ein UTF-8-kodiertes JSON-Dokument.



- Veraltet: Bitte den [JSON Converter 2.0](#) verwenden!
- Ehemalige Bezeichnung bis Version 4.5.1: *JSON Converter Adapter*



Bitte beachten Sie bei der Konvertierung von XML ins JSON-Format:

- Processing Instructions, Kommentare, Entities und CDATA-Abschnitte im Input-XML-Dokument können nicht konvertiert werden.
- Wenn Sie aus XML-Elementstrukturen explizit Arrays in JSON erzeugen möchten, fügen Sie dem entsprechenden Element das Attribut `class="array"` hinzu.
- Um leere Arrays in JSON zu unterdrücken, fügen Sie dem entsprechenden XML-Element ein Attribut `class="object"` hinzu, so wird beispielsweise `<element class="object"/>` in JSON `element : null`.
- Mehr Informationen zur verwendeten Bibliothek finden Sie unter <http://json-lib.sourceforge.net/usage.html>.

Eigenschaften

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>JSON2XML</i>: JSON-Dokument in XML konvertieren • <i>XML2JSON</i>: XML-Dokument in JSON konvertieren
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.json.JSONConverter</i> : Hauptklasse (Standard)
----------------	---

6.19.2.3 JSON Validation



JSON Validation: Validiert ein Input-JSON-Dokument gegen ein JSON-Schema. Ist das geprüfte JSON-Dokument gültig, gibt der Adapter dieses unverändert aus. Wenn das geprüfte JSON-Dokument ungültig ist, wird ein XML-Dokument mit einer entsprechenden Fehlermeldung ausgegeben.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>validate</i> : JSON-Dokument validieren
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.json.JSONValidator</i> : Hauptklasse (Standard)
<i>schemaURL</i>	Pfad zur Schema-Definition (JSON-Schema) Mögliche Werte: Beliebige <i>xstore://</i> -URL, z. B. <i>xstore://Project/Folder/Schema.json</i>

Statuswerte

<i>-1 (error)</i>	Während der Ausführung des Adapters trat ein Fehler auf. Nähere Details finden Sie im Server-Log.
-------------------	---

1 (<i>successful</i>)	Das geprüfte JSON-Dokument ist gültig gemäß dem angegebenen Schema; Der Adapter gibt das geprüfte Input-JSON-Dokument unverändert aus.
0	Das geprüfte JSON-Dokument ist nicht gültig gemäß dem angegebenen Schema. Der Adapter gibt eine entsprechende Fehlermeldung aus (siehe <i>Ausgabe</i>).

Input

Als Input erwartet der Adapter eine JSON-Datei, die er anschließend gegen ein JSON-Schema validiert.

Ausgabe


Der Adapter gibt bei erfolgreicher Validierung das Input-Dokument unverändert aus. Ist das geprüfte JSON-Dokument ungültig, wird ein XML-Dokument mit einer entsprechenden Fehlermeldung ausgegeben, wobei für jeden Fehler ein XML-Element `<Error/>` erzeugt wird. Dieses Element enthält eine detaillierte Fehlerbeschreibung, z.B.:

Beispiel einer Fehlermeldung

```
<ValidationErrors>
  <Error>#/rectangle/a: -5.0 is not higher or equal to 0</Error>
</ValidationErrors>
```

6.19.3 OData Connector

OData Connector: Ermöglicht die Kommunikation zu Systemen, die ReST-Services nach OData-Standard bereitstellen z.B. SAP S4/Hana, Microsoft Sharepoint, Microsoft Dynamics Navision und weitere.

 Bitte beachten Sie, dass sich der OData Connector in einem experimentellen Stadium befindet und mit zukünftigen Versionen der X4 Suite erweitert wird.

Eigenschaften

Operation	<p>Vom Adapter ausgeführte Operationen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ReadMetadata: Ruft das Datenmodell und die Operationen ab, die von einem bestimmten Dienst ausgeführt werden. • ReadCollections: Ruft alle verfügbaren Auflistungen aus dem Dienst ab. • ReadEntitySet: Ruft eine bestimmte Auflistung aus dem Service ab. • ReadSingleEntity: Ruft eine einzelne Entität aus dem Dienst auf, spezifiziert durch die Auflistung und die ID. • ReadNavigationEntity: Ruft die zugehörige Auflistung für eine Entität ab. • ReadAssociation: Ruft die zugehörige Zuordnung zur Entität ab. • ReadSingleton: Ruft das spezifische Singleton ab. • ReadMediaEntity: Ruft die Medienentität ab. • CreateEntity: Erstellt neue Entitäten im System, spezifiziert durch die Auflistung und die ID in einem XML-Inputdokument. • UpdateEntity: Aktualisiert im System vorhandene Entitäten, spezifiziert durch die Auflistung und die ID in einem XML-Inputdokument. • DeleteEntity: Löscht im System vorhandene Entitäten, spezifiziert durch die Auflistung und die ID in einem XML-Inputdokument. • CreateSingleton: Erstellt ein neues Singleton im System, spezifiziert durch die Auflistung und die ID in einem XML-Inputdokument. • UpdateSingleton: Aktualisiert ein Singleton im System, spezifiziert durch die Auflistung und die ID in einem XML-Inputdokument. • DeleteSingleton: Löscht ein Singleton im System, spezifiziert durch die Auflistung und die ID in einem XML-Inputdokument. • CreateMediaEntity: Erstellt neue Medienentitäten im System, spezifiziert durch die Auflistung und die ID innerhalb der Parameter entitySet und entityId. • UpdateMediaEntity: Aktualisiert Medienentitäten im System, spezifiziert durch die Auflistung und die ID innerhalb der Parameter entitySet und entityId. • DeleteMediaEntity: Löscht Medienentitäten im System, spezifiziert durch die Auflistung und die ID innerhalb der Parameter entitySet und entityId. • Functions: Fragt eine durch den Dienst ausgeführte Operation (Funktion, Aktion) an, wobei die entsprechenden Ausgabewerte abgerufen werden.
-----------	---

Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.x4.adapter.odataquery.ODataQueryAdapter</i>: Hauptklasse (Standard)</p>
serviceUrl	OData-Service-URL, z.B. <i>https://services.odata.org/V4/TripPinServiceRW</i>
entitySet	Entitäts-Auflistung
entityId	Entitäts-ID

navigationProperty	Zur Entität zugehörige Auflistung abrufen
association	Zur Entität zugehörige Zuordnung abrufen
singletonId	Singleton-ID
contentTypeMetadata	ContentType der Service-Metadaten Mögliche Werte: <ul style="list-style-type: none"> • Full: Vollständige Metadaten ausgeben • Minimal: Nur grundlegende Metadaten ausgeben • None: Keine Metadaten ausgeben

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt und ein Output wurde ausgegeben
-1	Es kam zu einem Fehler während der Ausführung und es wurde kein Output ausgegeben

Input

Abhängig von der gewählten Operation, erwartet der Adapter folgende Input-Struktur:

- Die Operationen ReadMetadata, ReadSingleEntity, ReadNavigationEntity, ReadSet und ReadSingleton benötigen kein Input-Dokument. Die Parameter entitySet und entityId werden jedoch benötigt.
- Die Operation ReadAssociation benötigen kein Input-Dokument. Die Parameter entitySet, entityId und association werden jedoch benötigt.
- Operation ReadEntitySet:
Folgende Systemabfrageoptionen können für die Operation ReadEntitySet verwendet werden:
 - \$count: Ignoriert \$top, \$skip, oder \$expand Abfrageoptionen und gibt die Gesamtzahl der Ergebnisse auf allen Seiten zurück, einschließlich der Ergebnisse, die mit einem bestimmten \$filter und einer bestimmten \$search übereinstimmen.
 - \$skip: Werden \$top und \$skip gemeinsam verwendet, muss \$skip vor \$top verwendet unabhängig davon, in welcher Reihenfolge sie in der Abfrage vorkommen.
 - \$top: Ermöglicht es dem Benutzer eines OData-Dienstes, die maximale Anzahl von Einträgen anzugeben, die von Anfang an zurückgegeben werden sollen.
 - \$orderby: Beim Anfordern einer Liste von Entitäten aus einem Dienst, liegt es an der Service-Implementierung zu entscheiden, in welcher Reihenfolge sie präsentiert werden.
 - \$filter: Beim Anfordern einer Liste von Entitäten aus einem Dienst, werden standardmäßig alle Entitäten aus der Liste zurückgegeben. Der Verbraucher eines OData-Dienstes möchte möglicherweise eine Teilmenge empfangen können, indem er bestimmte Kriterien angibt, die jede der zurückgegebenen Entitäten erfüllen muss.
 - \$select: Beim Anfordern einer Entitätsauflistung aus dem Backend, gibt der OData-Dienst eine Liste von Entitäten zurück und jede Entität enthält eine Liste von Eigenschaften.
 - Nur einen Eigenschaftsnamen angeben: \$select=Name
 - Eine kommagetrennte Liste von Eigenschaften angeben: \$select=Name,Description

- Mit einem Stern (*) alle Eigenschaften einschließen: \$select=*
- \$expand: Zeigt die zugehörigen Entitäten und Stream-Werte an, die inline angegeben werden **müssen**.
- \$search: Beschränkt das Ergebnis auf die Elemente, die dem angegebenen Suchbegriff entsprechen.
- \$format: Gibt den Medientyp der Antwort an.

ReadEntitySet

```
<Request>
  <QueryOptions entitySet="Products">
    <Parameter name="format" value="application/xml"></Parameter>
    <Parameter name="search" value="Ergo"></Parameter>
    <Parameter name="top" value="10"></Parameter>
    <Parameter name="skip" value="5"></Parameter>
    <Parameter name="count"></Parameter>
    <Parameter name="filter" value="ID le 10"></Parameter>
    <Parameter name="count" value="true"></Parameter>
    <Parameter name="expand">
      <Items>Category</Items>
    </Parameter>
    <Parameter name="select">
      <Items>ID</Items>
      <Items>Description</Items>
    </Parameter>
  </QueryOptions>
</Request>
```

- Operationen CreateEntity, UpdateEntity und DeleteEntity:

Create-, Update- und Delete-Anfrage

```

<Request>
  <Entity qualifiedName="Microsoft.OData.SampleService.Models.TripPin.Person" entitySet="People">
    <Property name="UserName" value="carlosreyestest" type="String"
isId="true"/>
    <Property name="FirstName" value="Carlos" type="String"/>
    <Property name="LastName" value="Carlos" type="String"/>
    <Collection name="AddressInfo">
      <Property type="Microsoft.OData.SampleService.Models.TripPin.
Location">
        <Property name="Address" value="Calle Falsa 123" type="Str
ing"/>
        <Property name="City" type="Microsoft.OData.SampleService.
Models.TripPin.City">
          <Property name="CountryRegion" value="Espana" type="
String"/>
          <Property name="Name" value="Madrid" type="String"/>
          <Property name="Region" value="Madrid" type="String"
/>
        </Property>
      </Property>
    </Collection>
  </Entity>
</Request>

```

- Operationen CreateSingleton, UpdateSingleton und DeleteSingleton:

CreateSingleton-, UpdateSingleton- und DeleteSingleton-Anfrage

```

<Request>
  <Singleton qualifiedName="Microsoft.OData.SampleService.Models.TripPin.P
erson" entitySet="People">
    <Property name="UserName" value="carlosreyestest" type="String"
isId="true"/>
    <Property name="FirstName" value="Carlos" type="String"/>
    <Property name="LastName" value="Carlos" type="String"/>
    <Collection name="AddressInfo">
      <Property type="Microsoft.OData.SampleService.Models.TripPin.
Location">
        <Property name="Address" value="Calle Falsa 123" type="Str
ing"/>
        <Property name="City" type="Microsoft.OData.SampleService.
Models.TripPin.City">
          <Property name="CountryRegion" value="Espana" type="
String"/>
          <Property name="Name" value="Madrid" type="String"/>
          <Property name="Region" value="Madrid" type="String"
/>
        </Property>
      </Property>
    </Collection>
  </Singleton>
</Request>

```

- Für die Operationen CreateMediaEntity, UpdateMediaEntity und DeleteMediaEntity sind sowohl die Parameter entitySet und entityId als auch der Medien-Input im Adapter notwendig.
- Operation ExecuteFunction:

Anfrage ExecuteFunction

```

<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Function name="CountCategories">
    <Parameter name="Amount" value="2"></Parameter>
  </Function>
</Request>

```

Unterstützte Typen

Adapter Input-Typ	Typ	Bedeutung
Binary	Edm.Binary	Binärdaten
Boolean	Edm.Boolean	Binäre Logik
Byte	Edm.Byte	Vorzeichenlose 8-Bit Ganzzahl
Date	Edm.Date	Datum ohne Zeitzoneversatz
DateTimeOffset	Edm.DateTimeOffset	Datum und Uhrzeit mit Zeitzoneversatz, keine Schaltsekunden

Adapter Input-Typ	Typ	Bedeutung
Decimal	Edm.Decimal	Zahlenwerte mit dezimaler Darstellung
Double	Edm.Double	IEEE 754 binary64-Gleitkommazahl (15-17 Dezimalstellen)
Duration	Edm.Duration	Signierte Dauer in Tagen, Stunden, Minuten und (Teil-)Sekunden
Guid	Edm.Guid	16 Byte (128 Bit) eindeutige Kennung
Int16	Edm.Int16	Vorzeichenbehaftete 16-Bit Ganzzahl
Int32	Edm.Int32	Vorzeichenbehaftete 32-Bit Ganzzahl
Int64	Edm.Int64	Vorzeichenbehaftete 64-Bit Ganzzahl
SByte	Edm.SByte	Vorzeichenbehaftete 8-Bit Ganzzahl
Single	Edm.Single	IEEE 754 binary32-Gleitkommazahl (6-9 Dezimalstellen)
String	Edm.String	Folge von UTF-8 Zeichen

Ausgabe

Abhängig von der gewählten Operation, gibt der Adapter folgende Strukturen aus:

- Operation ReadMetadata:

Antwort ReadMetadata

```

<?xml version="1.0" encoding="UTF-8"?>
<edmx:Edmx Version="4.0"
  xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx">
  <edmx:DataServices>
    <Schema xmlns="http://docs.oasis-open.org/odata/ns/edm"
      Namespace="OData.Demo">
      <EntityType Name="Product">
        <Key>
          <PropertyRef Name="ID" />
        </Key>
        <Property Name="ID" Type="Edm.Int32"></Property>
        <Property Name="Name" Type="Edm.String"></Property>
        <Property Name="Description" Type="Edm.String"></Property>
        <NavigationProperty Name="Category"
          Type="OData.Demo.Category" Partner="Products"></
NavigationProperty>
      </EntityType>
      <EntityType Name="Category">
        <Key>
          <PropertyRef Name="ID" />
        </Key>
        <Property Name="ID" Type="Edm.Int32"></Property>
        <Property Name="Name" Type="Edm.String"></Property>
        <NavigationProperty Name="Products"
          Type="Collection(OData.Demo.Product)" Partner="Category"></
NavigationProperty>
      </EntityType>
      <EntityType Name="Advertisement" HasStream="true">
        <Key>
          <PropertyRef Name="ID" />
        </Key>
        <Property Name="ID" Type="Edm.Guid"></Property>
        <Property Name="Name" Type="Edm.String"></Property>
        <Property Name="AirDate" Type="Edm.DateTimeOffset"></Property>
      </EntityType>
      <Action Name="Reset" IsBound="false">
        <Parameter Name="Amount" Type="Edm.Int32"></Parameter>
      </Action>
      <Function Name="CountCategories">
        <Parameter Name="Amount" Type="Edm.Int32" Nullable="false"></
Parameter>
      <ReturnType Type="Collection(OData.Demo.Category)" />
      </Function>
      <EntityContainer Name="Container">
        <EntitySet Name="Products" EntityType="OData.Demo.Product">
          <NavigationPropertyBinding
            Path="Category" Target="Categories" />
        </EntitySet>
        <EntitySet Name="Categories"
          EntityType="OData.Demo.Category">
          <NavigationPropertyBinding
            Path="Products" Target="Products" />
        </EntitySet>

```

```

        <EntitySet Name="Advertisements"
            EntityType="OData.Demo.Advertisement"></EntitySet>
        <ActionImport Name="Reset" Action="OData.Demo.Reset"></
ActionImport>
        <FunctionImport Name="CountCategories"
            Function="OData.Demo.CountCategories"
            EntitySet="OData.Demo.Categories" IncludeInServiceDocument="
true"></FunctionImport>
    </EntityContainer>
</Schema>
</edmx:DataService>
</edmx:Edmx>

```

- Operationen ReadSingleEntity und ReadSingleton:

Antwort ReadSingleEntity

```

<?xml version="1.0" encoding="UTF-8"?>
<Object>
  <Value type="String" name="@odata.context">$metadata#Products/$entity</
Value>
  <Value type="Number" name="ID">1</Value>
  <Value type="String" name="Name">Notebook Professional 17</Value>
  <Value type="String" name="Description">Notebook Professional, 2.8GHz - 15
XGA
    - 8GB DDR3 RAM - 500GB</Value>
</Object>

```

- Operation ReadEntitySet:

Antwort ReadCollectionEntity

```

<?xml version="1.0" encoding="UTF-8"?>
<Object>
  <Value type="String" name="@odata.context">$metadata#Products</Value>
  <Array name="value">
    <Object>
      <Value type="Number" name="ID">0</Value>
      <Value type="String" name="Name">Notebook Basic 15</Value>
      <Value type="String" name="Description">Notebook Basic, 1.7GHz - 15
XGA - 1024MB DDR2 SDRAM - 40GB</Value>
    </Object>
    <Object>
      <Value type="Number" name="ID">1</Value>
      <Value type="String" name="Name">Notebook Professional 17</Value>
      <Value type="String" name="Description">Notebook Professional, 2.8GHz
- 15 XGA - 8GB DDR3 RAM - 500GB</Value>
    </Object>
    <Object>
      <Value type="Number" name="ID">2</Value>
      <Value type="String" name="Name">1UMTS PDA</Value>
      <Value type="String" name="Description">Ultrafast 3G UMTS/HSDPA Pocket
PC, supports GSM network</Value>
    </Object>
    <Object>
      <Value type="Number" name="ID">3</Value>
      <Value type="String" name="Name">Comfort Easy</Value>
      <Value type="String" name="Description">32 GB Digital Assitant with
high-resolution color screen</Value>
    </Object>
    <Object>
      <Value type="Number" name="ID">4</Value>
      <Value type="String" name="Name">Ergo Screen</Value>
      <Value type="String" name="Description">19 Optimum Resolution 1024 x
768 @ 85Hz, resolution 1280 x 960</Value>
    </Object>
    <Object>
      <Value type="Number" name="ID">5</Value>
      <Value type="String" name="Name">Flat Basic</Value>
      <Value type="String" name="Description">Optimum Hi-Resolution max.
1600 x 1200 @ 85Hz, Dot Pitch: 0.24mm</Value>
    </Object>
  </Array>
</Object>

```

- Operation ReadCollections:

Antwort ReadCollections

```
<?xml version="1.0" encoding="UTF-8"?>
<Object>
  <Value type="String" name="@odata.context">$metadata</Value>
  <Array name="value">
    <Object>
      <Value type="String" name="name">Products</Value>
      <Value type="String" name="url">Products</Value>
    </Object>
    <Object>
      <Value type="String" name="name">Categories</Value>
      <Value type="String" name="url">Categories</Value>
    </Object>
    <Object>
      <Value type="String" name="name">Advertisements</Value>
      <Value type="String" name="url">Advertisements</Value>
    </Object>
    <Object>
      <Value type="String" name="name">CountCategories</Value>
      <Value type="String" name="url">CountCategories</Value>
      <Value type="String" name="kind">FunctionImport</Value>
    </Object>
  </Array>
</Object>
```

- Operationen CreateEntity, UpdateEntity und DeleteEntity:

CreateEntity-, UpdateEntity- und DeleteEntity-Antwort

```
<Response>
  <OKEntities entitySet="Products" qualifiedName="OData.Demo.Product">
    <Property isId="true" type="int32" value="1" name="ID"/>
    <Property type="String" value="Car" name="Name"/>
    <Property type="String" value="Good car" name="Description"/>
  </OKEntities>
  <FailEntities entitySet="Products" qualifiedName="OData.Demo.Product">
    <Property isId="true" type="int32" value="2" name="ID"/>
    <Property type="String" value="Cookie" name="Name"/>
    <Property type="String" value="Great cookie" name="Description"/>
  </FailEntities>
</Response>
```

- Operationen CreateSingleton, UpdateSingleton und DeleteSingleton:

CreateSingleton-, UpdateSingleton- und DeleteSingleton-Antwort

```

<Response>
  <OKEntities entitySet="Products" qualifiedName="OData.Demo.Product">
    <Property isId="true" type="int32" value="1" name="ID"/>
    <Property type="String" value="Car" name="Name"/>
    <Property type="String" value="Good car" name="Description"/>
  </OKEntities>
  <FailEntities entitySet="Products" qualifiedName="OData.Demo.Product">
    <Property isId="true" type="int32" value="2" name="ID"/>
    <Property type="String" value="Cookie" name="Name"/>
    <Property type="String" value="Great cookie" name="Description"/>
  </FailEntities>
</Response>

```

- Operation ExecuteFunction:

Antwort ExecuteFunction

```

<?xml version="1.0" encoding="UTF-8"?>
<Object>
  <Value type="String" name="@odata.context">$metadata#Collection(OData.Demo.Category)</Value>
  <Array name="value">
    <Object>
      <Value type="Number" name="ID">0</Value>
      <Value type="String" name="Name">Notebooks</Value>
    </Object>
    <Object>
      <Value type="Number" name="ID">1</Value>
      <Value type="String" name="Name">Organizers</Value>
    </Object>
    <Object>
      <Value type="Number" name="ID">2</Value>
      <Value type="String" name="Name">Monitors</Value>
    </Object>
  </Array>
</Object>

```

6.19.4 URL Encoding Converter

URL Encoding Converter: Wandelt eine URL oder eine Zeichenkette in eine URL und umgekehrt.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Encode: Verschlüsselt die angegebene URL • Decode: Entschlüsselt die angegebene URL • EncodePlain: Verschlüsselt den angegebenen Text • DecodePlain: Entschlüsselt den angegebenen Text
-----------	---

Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.url.UrlEncodingConverter</i>: Hauptklasse (Standard)</p>
---------	---

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt.
-1	Die Operation konnte nicht ausgeführt werden.

Input

Der Adapter erwartet eine XML-Datei mit der zu konvertierenden Zeichenkette bzw. URL als Input.

- Encode-Operation:

Erwartete Struktur für die Operation Encode

```
<?xml version="1.0" encoding="UTF-8" ?>
<URL>http://www.softproject.de/search?q=Hello Wörl@dX4$</URL>
```

- Decode-Operation:

Erwartete Struktur für die Operation Decode

```
<?xml version="1.0" encoding="UTF-8" ?>
<URL>http://www.softproject.de/search?q=Hello%C3%B6%C3%B6rl@dX4$</URL>
```

- EncodePlain-Operation:

Erwartete Struktur für die Operation EncodePlain

```
<?xml version="1.0" encoding="UTF-8"?>
<Text>Hello Wörl@dX4$</Text>
```

- DecodePlain-Operation:

Erwartete Struktur für die Operation DecodePlain

```
<?xml version="1.0" encoding="UTF-8"?>
<Text>Hell%C3%B6%20W%C3%B6rld@X4$</Text>
```

Output:

Der Adapter liefert ein XML-Dokument mit der konvertierten URL.

- Encode-Operation

Beispiel-Output für die Operation Encode

```
<?xml version="1.0" encoding="UTF-8" ?>
<URL>http://www.softproject.de/search?q=Hell%C3%B6%20W%C3%B6rld@X4$</URL>
```

- Decode-Operation

Beispiel-Output für die Operation Decode

```
<?xml version="1.0" encoding="UTF-8" ?>
<URL>http://www.softproject.de/search?q=Hellö Wörld@X4$</URL>
```

- EncodePlain-Operation:

Beispiel-Output für die Operation EncodePlain

```
<?xml version="1.0" encoding="UTF-8"?>
<Text>Hellö Wörld@X4$</Text>
```

- DecodePlain-Operation:

Beispiel-Output für die Operation DecodePlain

```
<?xml version="1.0" encoding="UTF-8"?>
<Text>Hell%C3%B6%20W%C3%B6rld@X4$</Text>
```

6.19.5 URL Reader



URL Reader: Liest ein Dokument von einer URL (*Uniform Resource Locator*) und gibt dieses Dokument aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Read</i> : Dokument von einer URL lesen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.urlReader.URLReader</i> : Hauptklasse (Standard)
<i>Url</i>	Uniform Resource Locator (URL) mit beliebigem gültigen Protokoll (<i>http://, ftp://, xstore://, file://</i> etc.) Mögliche Werte: Beliebige gültige URL (z.B. <i>https://www.google.com/favicon.ico</i>)

6.19.6 Webservice

6.19.6.1 HTTP(S) Communication (Veraltet)



HTTP(S) Communication (Veraltet): Dieser nur noch aus Abwärtskompatibilitätsgründen mitgelieferte Adapter ermöglichte die Kommunikation mit Servern über HTTP (*HyperText Transfer Protocol*) oder HTTPS (*HTTP Secure*) mit den Methoden GET, POST, PUT, DELETE (und *Multipart POST*). Der Adapter unterstützt HTTP- und NTLM-Authentifizierung sowie Proxy-Server und präemptive Authentifizierung.



Der *HTTP(S) Communication* Adapter ist abgekündigt und wird in zukünftigen Versionen der X4 Suite nicht mehr unterstützt. Bitte verwenden Sie stattdessen den **HTTP(S) Connector**.



Der Adapter unterstützt auch *SOAP MTOM (Message Transmission Optimization Mechanism)*, mit dem sich große Dokumente bzw. Binärdaten in MTOM-basierenden WebServices über XOP (XML-binary Optimized Packaging) performant übertragen lassen, siehe [WebService Starter \(veraltet\)](#).

Die MTOM-Verarbeitung für Anfragen wird nur aktiviert, wenn das Input-Dokument des Adapters den MIME-Typ *text/xml* oder *application/xml* besitzt.

Gesteuert wird die MTOM-Funktion über Processing Instructions, die dem SOAP-Request-Dokument zum Aktivieren von MTOM auf Eingangsseite hinzugefügt werden:

- `<?useMTOM?>`: MTOM-Funktion für diesen SOAP-Request aktivieren
- `<?MTOM-xpath 0-n XPath-Ausdrücke ?>`: Beliebige viele XPath-Ausdrücke, um die Elemente zu definieren, die Binärdaten enthalten
- `<?MTOM-size Größe in KB ?>`: Größe der Daten in Kilobyte, ab der die Komprimierung aktiviert wird (Standard: 1 KB)

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Get</i>: Daten von einer URL mit der GET-Methode anfordern • <i>Post</i>: Daten im Body des HTTP-Requests per POST an eine URL senden • <i>Put</i>: Daten im Body des HTTP-Requests über die PUT-Methode auf einer URL publizieren • <i>Delete</i>: Die im HTTP-Request aufgerufene Ressource per DELETE-Methode löschen • <i>Multipart Post</i>: Wie <i>Post</i>, jedoch wird der Body in mehrere Teile zerlegt. <i>Multipart Post</i> wird zum Hochladen von Dateien auf einen Webserver verwendet.
------------------	---

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.httpadapter.HttpAdapter</code>: Hauptklasse (Standard)</p>
<i>host</i>	<p>IP-Adresse oder Host-Name des Webserver</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige gültige Zeichenkette (z. B. <code>www.google.com</code>) • URL mit HTTP-Protokollangabe (z. B. <code>http://www.google.com</code>) • URL mit HTTPS-Protokollangabe (z. B. <code>https://www.google.com</code>)
<i>port</i>	<p>Portnummer des HTTP(S)-Dienstes</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl (z. B. <code>8080</code>) • <i>(leer)</i>: Port 80 verwenden (bei HTTPS standardmäßig Port 443) • <code>80</code>: Standard-Port für HTTP (Standard)

<i>remoteFilePath</i>	<p>(Nur für Mehrfach-Operationen mit gemeinsamem Basis-Pfad relevant) Ordnerpfad auf dem Webserver; Führende oder angehängte Schrägstriche sind optional, da sie implizit angefügt werden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige Pfadangabe (z. B. <i>/intl/en_ALL/images/</i>) • <i>(leer)</i>: Kompletten Pfad aus <i>documentName</i> verwenden
<i>documentName</i>	<p>Pfad (relativ zu <i>remoteFilePath</i>) zu einem Dokument auf dem Webserver, das aufgerufen werden soll; falls <i>remoteFilePath</i> leer ist, bezieht sich der Pfad auf die Wurzel /</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebiger Dateiname (z. B. <i>logo.gif</i>) • <i>(leer)</i>: Parameterwert von <i>remoteFilePath</i> verwenden inklusive abschließendem Schrägstrich (falls noch nicht vorhanden)
<i>login</i>	<p>Benutzername für eine einfache HTTP-Authentifizierung</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>pwd</i>	<p>Passwort für eine einfache HTTP-Authentifizierung</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>preemptiveAuth</i>	<p>Präemptive Authentifizierung (mit den Zugangsdaten aus Parameter <i>login</i> und <i>pwd</i>) für alle HTTP(S)-Aufrufe aktivieren</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Ja</i>: Zugangsdaten direkt (präemptiv) bei jedem HTTP(S)-Aufruf mitschicken • <i>Nein</i>: Keine präemptive Authentifizierung (Standard)
<i>pxHost</i>	<p>IP-Adresse oder Host-Name eines Proxy-Servers</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette (z. B. <i>192.168.1.1</i>)</p>
<i>pxPort</i>	<p>Portnummer des Proxy-Servers</p> <p>Mögliche Werte: Beliebige ganze Zahl</p>
<i>pxLogin</i>	<p>Benutzername für den Proxy-Server</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>

<i>pxPwd</i>	Passwort für den Proxy-Server Mögliche Werte: Beliebige Zeichenkette
<i>pxNTLMDomain</i>	Windows-Domäne für eine NTLM-Authentifizierung via NTLM-Proxy Mögliche Werte: Beliebiger Domänenname
<i>pxNTLMhost</i>	Host (Computer-Name), von dem die Authentifizierungsanfrage ausgeht (für eine NTLM-Authentifizierung via NTLM-Proxy) Mögliche Werte: Beliebiger Hostname
<i>pxUseNTLM</i>	Proprietäre NTLM-Authentifizierung von Microsoft für den Proxy verwenden Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: Per NTLM authentifizieren • <i>nein</i>: Nicht per NTLM authentifizieren (Standard)
<i>nTLMDomain</i>	Windows-Domäne für eine NTLM-Server-Authentifizierung Mögliche Werte: Beliebiger Domänenname
<i>nTLMhost</i>	Host (Computer-Name), von dem die NTLM-Server-Authentifizierungsanfrage ausgeht Mögliche Werte: Beliebiger Hostname
<i>useNTLM</i>	Proprietäre NTLM-Server-Authentifizierung von Microsoft verwenden Mögliche Werte: <ul style="list-style-type: none"> • <i>ja</i>: Per NTLM authentifizieren • <i>nein</i>: Nicht per NTLM authentifizieren (Standard)
<i>headers</i>	Benutzerdefinierte HTTP-Header-Parameter Mögliche Werte: Beliebige Parameter nach folgendem Muster: <Parameter1>=<Wert1>; <Parameter2>=<Wert2> (z. B. <i>custom1="value 1"; custom2="value 2"</i>)

<i>parts</i>	<p>Definition der Teile einer Multipart-Post-Methode (nur wenn Eigenschaft <i>Operation</i> auf <i>Multipart Post</i> gesetzt ist), durch Semikolon getrennt</p> <p>Mögliche Werte: Beliebige Teile nach folgendem Muster: "<Variablenname>": "<Teil>": "<Dateiname>": "<MIME-Typ>": "<Zeichenkodierung>"; wobei MIME-Typ und Zeichenkodierung optional sind (z. B. <i>"part1": "part1": "data1.xml": "text/xml": "UTF-8"; "part2": "part2": "data2.xml"</i>)</p>
<i>timeout</i>	<p>Zeitbeschränkung für den Verbindungsaufbau in Millisekunden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl • 0: Keine Zeitbeschränkung
<i>keystorePath</i>	<p>Kompletter Pfad zur Keystore-Datei; wenn ein Keystore angegeben wird, werden nur Zertifikate dieses Keystores akzeptiert</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebiger Pfad zur Keystore-Datei (nur Zertifikate dieses Keystores verwenden) • (kein Pfad): Alle Zertifikate akzeptieren
<i>keystorePassword</i>	<p>Passwort für die Keystore-Datei</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>toXML</i>	<p>Alle Header als Elemente und den Inhalt des Bodys als Base64-kodierten Text in einem XML-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: XML-Dokument ausgeben • <i>nein</i>: Antwort des Webserver unverändert ausgeben

6.19.6.2 HTTP(S) Connector



HTTP(S) Connector: Kommuniziert mit Servern über das *HyperText Transfer Protocol* (HTTP) oder *HTTP Secure* (HTTPS) auf Basis des *Apache HttpClient* in Version 4. Der Adapter bietet die Standard-HTTP-Methoden GET, POST, PUT, DELETE, HEAD, OPTIONS, TRACE und PATCH. Zusätzlich können benutzerdefinierte HTTP-Methoden verwendet werden. Dabei unterstützt der Adapter HTTP-Basic-, Digest- und NTLMv2-Authentifizierung sowie Proxy-Server und präemptive Authentifizierung.

Zudem unterstützt der Adapter auch Server Name Indication für HTTPS-Verbindungen sowie Client-Zertifikate zur Validierung von Requests.

Ehemalige Bezeichnung bis Version 4.5.1: *HTTP(S) Client 4 Adapter*

ⓘ Beachten Sie:


- Im Standardmodus (Parameter `fromXML` deaktiviert) verarbeitet der Adapter je nach Operation ggf. ein beliebiges Input-Dokument mit unverändertem Content Type und Zeichenkodierung.
- Im Expertenmodus (Parameter `fromXML` aktiviert) verarbeitet der Adapter ein spezifisches XML-Dokument, das Header, Cookie-Informationen etc. enthalten kann und mit dem sich die HTTP(S)-Operation detailliert steuern lässt, siehe Parameter `fromXML` und die Beschreibung der Input-Struktur.
- Dieser Adapter löst den bestehenden Adapter *HTTP(S) Communication* ab, siehe [HTTP\(S\) Communication \(Veraltet\)](#).

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Funktions-Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Get</i>: Daten von einer URL mit der GET-Methode anfordern • <i>Post</i>: Daten im Body des HTTP-Requests per POST an eine URL senden • <i>Put</i>: Daten im Body des HTTP-Requests über die PUT-Methode auf einer URL publizieren • <i>Delete</i>: die in der HTTP-Anfrage aufgerufene Ressource per DELETE-Methode löschen • <i>Head</i>: Nur die Header-Informationen per HEAD-Methode anfordern • <i>Options</i>: Vom Server per OPTIONS-Methode eine Liste der unterstützten HTTP-Methoden abrufen • <i>Trace</i>: Die vom Server empfangene Anfrage zurückgeben, um ggf. Änderungen von dazwischen liegenden Servern sichtbar zu machen • <i>Patch</i>: Die im Body des HTTP-Requests abgelegten Daten partiell aktualisieren • <i>Execute</i>: Daten im Body des HTTP-Request mit einer beliebigen HTTP-Methode senden, siehe auch Input-Beispiel
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.http4.HttpClient4Adapter</i>: Hauptklasse (Standard)</p>
<i>protocol</i>	<p>Verwendetes Protokoll (HTTP oder HTTPS)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • http: HTTP (unverschlüsselt) verwenden (Standard) • https: HTTPS (SSL-verschlüsselt) verwenden
<i>host</i>	<p>IP-Adresse oder Host-Name des physikalischen Servers</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette ohne Protokollangabe (z. B. <i>www.google.com</i>)</p>
<i>vHost</i>	<p>Host-Name eines logischen Servers (<i>virtual host</i>), falls dieser durch die physikalische Angabe in Parameter <i>host</i> nicht eindeutig identifiziert werden kann</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette ohne Protokollangabe (z. B. <i>www.google.com</i>)</p>
<i>port</i>	<p>Portnummer des HTTP(S)-Dienstes</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl (z. B. <i>8080</i>) • <i>-1</i>: Für HTTP-Verbindungen Port 80 verwenden; bei HTTPS-Verbindungen Port 443 verwenden (Standard)
<i>path</i>	<p>Ordnerpfad auf dem Server, beginnend mit einem Schrägstrich</p> <p>Mögliche Werte: Beliebige Pfadangabe (z. B. <i>/intl/en_ALL/images/logo.gif</i>)</p>
<i>method</i>	<p>Name der HTTP-Methode, die bei der Operation <i>Execute</i> verwendet werden soll. Optional kann die Methode auch über den XML-Request gesetzt werden.</p> <p>Mögliche Werte: beliebige Zeichenkette</p>

<i>authenticationEncoding</i>	<p>Zeichenkodierung für die Anmeldedaten</p> <p>Mögliche Werte: Angabe einer Zeichenkodierung, z. B. UTF-8 oder ISO-8859-1</p> <p> Mithilfe dieses Parameters können auch Umlaute in den Anmeldedaten verwendet werden.</p>
<i>query</i>	<p>Argument(e) für die HTTP(S)-Anfrage</p> <p>Mögliche Werte: Beliebige URL-kodierte Argumente ohne führendes Fragezeichen (z. B. <i>Arg1=V1&Arg2=V2</i>)</p>
<i>username</i>	<p>Benutzername für eine Basic-, Digest- oder NTLM-Authentifizierung</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Bei HTTP-Basic-Authentifizierung: der Benutzername • Bei HTTP-Basic-Authentifizierung an einer Windows-Domäne: Zeichenkette nach dem Muster <i><Domäne>\<Benutzername></i> (z. B. <i>MyDomain\MyUserName</i>) • Bei NTLM-Authentifizierung an einer Windows-Domäne: nur der Benutzername (und in Parameter <i>ntlmDomain</i> die Domäne angeben) (z. B. <i>MyUsername</i>)
<i>password</i>	<p>Passwort für eine Basic-, Digest- oder NTLM-Authentifizierung</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>ntlmWorkstation</i>	<p>Workstation (Computer-Name) des Clients (nur für NTLM-Authentifizierung, siehe Parameter <i>allowNtlm</i>)</p> <p>Mögliche Werte: Beliebiger gültiger Workstation-Name</p>
<i>ntlmDomain</i>	<p>Name der Domäne (nur für NTLM-Authentifizierung, siehe Parameter <i>allowNtlm</i>)</p> <p>Mögliche Werte: Beliebiger gültiger Domäne-Name (NetBIOS Name)</p>

<i>usePreemptiveBasicAuth</i>	<p>Präemptive Authentifizierung (mit den Zugangsdaten aus Parameter username und password) für HTTP(S)-Aufrufe aktivieren (nur für Basic-Authentifizierung, siehe Parameter allowBasic)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• ja: Zugangsdaten direkt (präemptiv) bei jedem HTTP(S)-Aufruf mitschicken• nein: Keine präemptive Authentifizierung (Standard)
<i>allowBasic</i>	<p>HTTP(S)-Basic-Authentifizierung verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• ja: Basic-Authentifizierung verwenden• nein: Keine Basic-Authentifizierung verwenden (Standard)
<i>allowDigest</i>	<p>Digest-Authentifizierung verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• ja: Per Digest authentifizieren• nein: Nicht per Digest authentifizieren (Standard)
<i>allowNtlm</i>	<p>Proprietäre NTLM-Server-Authentifizierung von Microsoft verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• ja: Per NTLM authentifizieren• nein: Nicht per NTLM authentifizieren (Standard)
<i>fromXml</i>	<p>Expertenmodus aktivieren; Der Adapter erwartet ein spezifisches XML-Dokument inklusive Headern, Cookie-Informationen etc., mit dem sich die HTTP(S)-Operation detailliert steuern lässt.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• ja: Expertenmodus aktivieren (Adapter erwartet ein spezifisches Input-XML-Dokument)• nein: Adapter-Input direkt verarbeiten

<i>toXml</i>	<p>Alle Header als Elemente und den Inhalt des Bodys als Base64-kodierten Text in einem XML-Dokument ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• ja: XML-Dokument ausgeben• nein: Server-Antwort unverändert ausgeben (Standard)
<i>hideCookieHeader</i>	<p>Im Ergebnis-XML-Dokument auch die Header-Elemente mit Cookie-Informationen ausgeben (wenn Parameter <i>toXml</i> aktiviert ist); Nur zu Entwicklungs- und Testzwecken deaktivieren oder bei Cookies, die vom Standard abweichen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• ja: Keine Cookie-Informationen als Header im Ergebnis-XML-Dokument (Standard)• nein: Cookie-Informationen als Header im Ergebnis-XML-Dokument ausgeben
<i>useProxy</i>	<p>Proxy-Server für den Verbindungsaufbau verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• ja: Verbindung über den in Parameter <i>proxyHost</i> angegeben Proxy-Server aufbauen• nein: Kein Proxy-Server (Standard)
<i>proxyHost</i>	<p>IP-Adresse oder Host-Name eines Proxy-Servers</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette (z. B. <i>192.168.1.1</i>)</p>
<i>proxyPort</i>	<p>Portnummer des Proxy-Servers</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• Beliebige ganze Zahl• -1: Standard-Port verwenden (Standard)

<i>proxyUsername</i>	<p>Benutzername für eine Basic-, Digest- oder NTLM-Authentifizierung am Proxy-Server</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Bei HTTP-Basic-Authentifizierung: der Benutzername • Bei HTTP-Basic-Authentifizierung an einer Windows-Domäne: Zeichenkette nach dem Muster <Domäne>\<Benutzername> (z. B. <i>MyDomain\MyUserName</i>) • Bei NTLM-Authentifizierung an einer Windows-Domäne: nur der Benutzername (und in Parameter <i>ntlmDomain</i> die Domäne angeben)(z. B. <i>MyUsername</i>)
<i>proxyPassword</i>	<p>Passwort für den Proxy-Server</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>proxyNtlmWorkstation</i>	<p>Workstation (Computer-Name), von der die Authentifizierungsanfrage ausgeht (für eine NTLM-Authentifizierung via NTLM-Proxy)</p> <p>Mögliche Werte: Beliebiger Workstation-Name</p>
<i>proxyNtlmDomain</i>	<p>Windows-Domäne für eine NTLM-Authentifizierung via NTLM-Proxy</p> <p>Mögliche Werte: Beliebiger Domänenname</p>
<i>proxyUsePreemptiveBasicAuth</i>	<p>Präemptive Authentifizierung (mit den Zugangsdaten aus Parameter <i>proxyUsername</i> und <i>proxyPassword</i>) für den Proxy-Server aktivieren (nur für Basic-Authentifizierung, siehe Parameter <i>proxyAllowBasic</i>)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Ja: Zugangsdaten direkt (präemptiv) bei jedem Proxy-Aufruf mitschicken • Nein: Keine präemptive Authentifizierung (Standard)
<i>proxyAllowBasic</i>	<p>Basic-Authentifizierung für den Proxy verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ja: Basic-Authentifizierung für den Proxy verwenden • nein: Keine Basic-Authentifizierung für den Proxy verwenden (Standard)

<i>proxyAllowDigest</i>	<p>Digest-Authentifizierung für den Proxy verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ja: Am Proxy per Digest authentifizieren • nein: Nicht per Digest am Proxy authentifizieren (Standard)
<i>proxyAllowNtlm</i>	<p>Proprietäre NTLM-Server-Authentifizierung von Microsoft für den Proxy verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ja: Am Proxy per NTLM authentifizieren • nein: Nicht per NTLM am Proxy authentifizieren (Standard)
<i>connectTimeout</i>	<p>Timeout-Wert für den Verbindungsaufbau in Millisekunden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • >0: Timeout-Wert in Millisekunden • <=0: Kein Timeout (Standard)
<i>socketTimeout</i>	<p>Timeout-Wert zwischen zwei eingehenden Datenpaketen in Millisekunden; Dieser sagt nichts über die Gesamt-Anfragezeit aus.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • >0: Timeout-Wert in Millisekunden • <=0: Kein Timeout (Standard)
<i>redirectsEnabled</i>	<p>URL-Weiterleitungen bei einer Server-Antwort mit HTTP-Status <i>301 Moved Permanently</i> zulassen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ja: URL-Redirects ausführen (Standard) • nein: Keine URL-Redirects ausführen
<i>circularRedirectsEnabled</i>	<p>Zirkuläre Weiterleitungen bei einer Server-Antwort mit HTTP-Status <i>301 Moved Permanently</i> zulassen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ja: Zirkuläre Weiterleitungen ausführen (Standard) • nein: Keine zirkulären Weiterleitungen ausführen

<i>maxRedirects</i>	<p>Maximale Anzahl der möglichen URL-Weiterleitungen bei einer Server-Antwort mit HTTP-Status <i>301 Moved Permanently</i></p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• Beliebige ganze Zahl• -1: Keine Beschränkung (Standard)
<i>acceptAllCertificates</i>	<p>Zertifikatsprüfung abschalten</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• beliebiger String: Zertifikate werden auf Vertrauen geprüft (Standard)• Yes, I want to disable security by accepting untrusted certificates.: Zertifikatsprüfung wird abgeschaltet <p>Bitte beachten: Hier muss die genaue Zeichenfolge eingegeben werden!</p>
<i>acceptAllSslHostnames</i>	<p>Hostname-Überprüfung für Zertifikate abschalten</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• beliebiger String: Hostname muss zum Zertifikat passen (Standard)• Yes, I want to disable security by accepting non-matching hostnames.: Hostname-Überprüfung wird abgeschaltet <p>Bitte beachten: Hier muss die genaue Zeichenfolge eingegeben werden!</p>
<i>useClientCertificate</i>	<p>Client-Zertifikate zur Validierung des Requests verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• ja: Client-Zertifikate zur Validierung des Requests verwenden• nein: Kein Client-Zertifikat zur Validierung des Requests verwenden (Standard)

<i>keystoreType</i>	<p>Keystore-Typ für die Speicherung von Client-Zertifikaten (gemäß http://docs.oracle.com/javase/6/docs/technotes/guides/security/StandardNames.html#KeyStore)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • jceks: Proprietäre Keystore-Implementierung von SunJCE verwenden • jks: Proprietäre Keystore-Implementierung von SUN verwenden • pkcs12: Keystore-Transferformat PKCS12 verwenden
<i>keystoreURL</i>	<p>URL zum Keystore für die Validierung des Requests über Client-Zertifikate</p> <p>Mögliche Werte: Beliebige gültige URL, z.B. file:/C:/Folder/ExampleKeystore</p>
<i>keystorePassword</i>	<p>Passwort für den Keystore zur Validierung des Requests über Client-Zertifikate</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette</p>
<i>keyAlias</i>	<p>Schlüssel-Alias zur Zugriff auf einen bestimmten Schlüssel innerhalb des Keystores</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette</p>
<i>aliasPassword</i>	<p>Zugehöriges Passwort für einen bestimmten Schlüssel innerhalb des Keystores</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette</p>

Input (Standardmodus)

Im Standardmodus (Parameter `fromXML` deaktiviert) verarbeitet der Adapter je nach Operation ggf. ein beliebiges Input-Dokument mit unverändertem Content Type und Zeichenkodierung; siehe [Document Properties Changer](#).

Input (Expertenmodus)

Im Expertenmodus (Parameter `fromXML` aktiviert) verarbeitet der Adapter ein spezifisches XML-Dokument, das Header, Cookie-Informationen etc. enthalten kann und mit dem sich die HTTP(S)-Operation detailliert steuern lässt.

Diese XML-Struktur entspricht im Wesentlichen sowohl der XML-Struktur, die der Adapter ausgibt, wenn Parameter `toXML` aktiviert ist, als auch der *X4 ReST*-Schnittstelle (wenn für die jeweilige URI-Operation in der Konfigurationsdatei `<X4>/X4DB/0/restconfig.xml` in Attribut `isDirect="false"` definiert wurde). Einziger Unterschied zum Ergebnis-XML-Dokument von *X4 ReST* ist, dass dieses den Base64-kodierten Body in einem Element `<Content>` (statt `<Body>` beim Adapter) bereitstellt.

In Element `<Cookies>` sind ggf. zudem Cookie-Informationen in jeweils einem Element `<Cookie>` enthalten:

```
<Request>
  <Headers>
    <Header>
      <Name><!--HTTP-Header-Name--></Name>
      <Value><!--Header-Wert--></Value>
    </Header>
    <!-- Oder bei Datumswerten: -->
    <Header>
      <Name><!--HTTP-Header-Name--></Name>
      <Date timezone="Zeitzone"><!--Header-Datumswert--></Date>
    </Header>
  </Headers>
  <Cookies>
    <Cookie expires="Verfallsdatum" path="Pfad" domain="Domäne">
      <Version><!--Versionsnummer--></Version>
      <Secure/>      <!-- Wenn Secure-Attribut gesetzt -->
      <Persistent />  <!-- Bei einem persistenten Cookie -->
      <Domain><!--Domäne--></Domain>
      <Path><!--Pfad--></Path>
      <Name><!--Cookie-Name--></Name>
      <Value><!--Cookie-Wert--></Value>
      <ExpiryDate><!--Verfallsdatum im Format YYYY-MM-DD hh:mm:ss--></ExpiryDate>
    </Cookie>
  </Cookies>
  <Body><!--Base64-kodierte Daten der HTTP-Anfrage--></Body>
</Request>
```

Operation Execute

Bei der Operation Execute in Verbindung mit dem aktiviertem Parameter *fromXML* kann die HTTP-Methode auch im Input-XML wie folgt gesetzt werden:

```
<Request method="CUSTOM_METHOD">
  <Headers>
    <!-- ... -->
  </Headers>
  <Cookies>
    <!-- ... -->
  </Cookies>
  <Body><!--Base64-kodierte Daten der HTTP-Anfrage--></Body>
</Request>
```

Ausgabe

Standardmäßig (wenn Parameter *toXml* deaktiviert ist) wird das Ergebnisdokument des HTTP(S)-Aufrufs ausgegeben.

Wenn Parameter *toXml* aktiviert ist, wird ein XML-Dokument ausgegeben, das alle HTTP-Header der Antwort jeweils in einem Element `<Header>` enthält. Falls ein Header-Wert mehrere Werte enthält,

die durch Kommas getrennt sind, so werden diese Teilwerte zusätzlich in jeweils einem Element `<Element>` in den Kindelementen `<Name>` und `<Value>` angegeben.

Der Inhalt des Bodys ist als Base64-kodierte Zeichenkette in Element `<Body>` enthalten. Wenn Parameter `hideCookieHeader` deaktiviert wurde, wird zusätzlich jeder Header ausgegeben, der zum Cookie geführt hat.

```
<Response>
  <Request line="HTTP-Anfragezeile">
    <Method><!--HTTP-Methode--></Method>
    <URI><!--Pfad auf dem Server--></URI>
    <Protocol>HTTP/1.1</Protocol>
  </Request>
  <Status line="HTTP-Statuszeile">
    <Code><!--Dreistelliger Statuscode--></Code>
    <Reason><!--Statusname--></Reason>
    <Protocol><!--HTTP/1.1</Protocol>
  </Status>
  <Headers>
    <Header>
      <Name><!--HTTP-Header-Name--></Name>
      <Value><!--Header-Wert--></Value>
    </Header>
    <!-- Oder bei Datumswerten: -->
    <Header>
      <Name><!--HTTP-Header-Name--></Name>
      <Date timezone="Zeitzone"><!--Header-Datumswert--></Date>
    </Header>
  </Headers>
  <Cookies>
    <Cookie expires="Verfallsdatum" path="Pfad" domain="Domäne">
      <Version><!--Versionsnummer--></Version>
      <Secure/>      <!-- Wenn Secure-Attribut gesetzt -->
      <Persistent />  <!-- Bei einem persistenten Cookie -->
      <Domain><!--Domäne--></Domain>
      <Path><!--Pfad</Path>
      <Name><!--Cookie-Name--></Name>
      <Value><!--Cookie-Wert--></Value>
      <ExpiryDate><!--Verfallsdatum im Format YYYY-MM-DD hh:mm:ss--></ExpiryDate>
    </Cookie>
  </Cookies>
  <Body><!--Base64-kodierte Daten im Body der Antwort--></Body>
</Response>
```

Statuswerte

-1	Technischer Fehler: Server antwortet nicht, oder das Adapter-Input-XML kann nicht verarbeitet werden (nur wenn der Expertenmodus über Parameter <code>fromXml</code> aktiviert ist)
>=100	Dreistelliger HTTP-Statuscode je nach Ergebnis des Aufrufs, siehe http://tools.ietf.org/html/rfc2616#section-6.1.1

6.19.6.3 Webservice (SOAP) Connector



Webservice (SOAP) Connector: Sendet eine SOAP-Anfrage (*SOAP Request*) via HTTP(S) zu der URL eines Webservice und gibt die zurückgelieferte SOAP-Antwort (*SOAP Response*) aus. Er kommuniziert mit Servern über HTTP (Hyper Text Transfer Protocol) oder HTTPS (HTTP Secure).

Falls eine Authentifizierung erforderlich ist, muss diese SOAP-basiert über den *SOAP Header* erfolgen; Dessen Aufbau (zertifikats-, token-basiert etc.) muss zunächst mit der Gegenstelle geklärt werden. Zudem werden HTTP- und NTLM-Authentifizierung sowie Proxy-Server unterstützt.

Ehemalige Bezeichnung: *HTTP WebService Request Adapter*

- i** Der Adapter unterstützt *SOAP MTOM (Message Transmission Optimization Mechanism)*, mit dem sich große Dokumente bzw. Binärdaten in MTOM-basierenden Webservice über XOP (XML-binary Optimized Packaging) performant übertragen lassen, siehe [WebService Starter \(veraltet\)](#).

Die MTOM-Verarbeitung für Anfragen wird nur aktiviert, wenn das Input-Dokument des Adapters den MIME-Typ `text/xml` oder `application/xml` besitzt.

Gesteuert wird die MTOM-Funktion über Processing Instructions, die dem SOAP-Request-Dokument zum Aktivieren von MTOM auf Eingangsseite hinzugefügt werden:

- `<?useMTOM?>`: MTOM-Funktion für diesen SOAP-Request aktivieren
- `<?MTOM-xpath 0-n XPath-Ausdrücke ?>`: Beliebige XPath-Ausdrücke, um die Elemente zu definieren, die Binärdaten enthalten
- `<?MTOM-size Größe in KB ?>`: Größe der Daten in Kilobyte, ab der die Komprimierung aktiviert wird (Standard: 1 KB)

Außerdem wandelt der Adapter MTOM-basierende SOAP-Responses, die ein Webservice liefert, automatisch in Standard-SOAP-Responses um.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt
	Mögliche Werte: <code>call</code> : Adapter ausführen

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <code>de.softproject.integration.adapter.httpws.HttpWSAdapter</code> : Hauptklasse (Standard)
----------------	--

<i>wsURL</i>	<p>URL, an die der SOAP-Request gesendet wird, mit dem Standard-Port <i>80</i></p> <p>Mögliche Werte: Beliebige Webservice-URL</p> <p> Abweichende Ports müssen in der URL angegeben werden.</p>
<i>soapAction</i>	<p>HTTP-Header des SOAP-Requests. Der Wert dieses Parameters ist eine URL, die die abgefragte Operation identifiziert.</p>
<i>login</i>	<p>Benutzername für eine einfache HTTP-Authentifizierung</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>pwd</i>	<p>Passwort für eine einfache HTTP-Authentifizierung</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>preemptiveAuth</i>	<p>Präemptive Authentifizierung (mit den Zugangsdaten aus Parameter <i>login</i> und <i>pwd</i>) für alle HTTP(S)-Aufrufe aktivieren</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Ja</i>: Zugangsdaten direkt (präemptiv) bei jedem HTTP(S)-Aufruf mitschicken • <i>Nein</i>: Keine präemptive Authentifizierung (Standard)
<i>pxHost</i>	<p>IP-Adresse oder Host-Name eines Proxy-Servers</p> <p>Mögliche Werte: Beliebige gültige Zeichenkette (z.B. <i>192.168.1.1</i>)</p>
<i>pxPort</i>	<p>Portnummer des Proxy-Servers</p> <p>Mögliche Werte: Beliebige ganze Zahl</p>
<i>pxLogin</i>	<p>Benutzername für den Proxy-Server</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>pxPwd</i>	<p>Passwort für den Proxy-Server</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>pxNTLMDomain</i>	<p>Windows-Domäne für eine NTLM-Authentifizierung via NTLM-Proxy</p> <p>Mögliche Werte: Beliebiger Domänenname</p>


<i>pxNTLMhost</i>	<p>Host (Computer-Name), von dem die Authentifizierungsanfrage ausgeht (für eine NTLM-Authentifizierung via NTLM-Proxy)</p> <p>Mögliche Werte: Beliebiger Hostname</p>
<i>pxUseNTLM</i>	<p>Proprietäre NTLM-Authentifizierung von Microsoft für den Proxy verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Per NTLM authentifizieren • <i>nein</i>: Nicht per NTLM authentifizieren (Standard)
<i>nTLMDomain</i>	<p>Windows-Domäne für eine NTLM-Server-Authentifizierung</p> <p>Mögliche Werte: Beliebiger Domänenname</p>
<i>nTLMhost</i>	<p>Host (Computer-Name), von dem die NTLM-Server-Authentifizierungsanfrage ausgeht</p> <p>Mögliche Werte: Beliebiger Hostname</p>
<i>useNTLM</i>	<p>Proprietäre NTLM-Server-Authentifizierung von Microsoft verwenden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Per NTLM authentifizieren • <i>nein</i>: Nicht per NTLM authentifizieren (Standard)
<i>timeout</i>	<p>Zeitbeschränkung für den Verbindungsaufbau in Millisekunden</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl • <i>0</i>: Keine Zeitbeschränkung
<i>keystorePath</i>	<p>Kompletter Pfad zur Keystore-Datei; wenn ein Keystore angegeben wird, werden nur Zertifikate dieses Keystores akzeptiert</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebiger Pfad zur Keystore-Datei (nur Zertifikate dieses Keystores verwenden) • (<i>kein Pfad</i>): Alle Zertifikate akzeptieren

<i>keystorePassword</i>	Passwort für die Keystore-Datei Mögliche Werte: Beliebige Zeichenkette
-------------------------	--

Input

Um einen Webservice mit MTOM aufzurufen, muss der SOAP-Request die o.g. Processing Instructions enthalten, beispielsweise:

```
<?useMTOM?>
<?MTOM-xpath soapenv:Envelope/soapenv:Body/ws:signVerifyRequest/ws:Service
  RequestData/ws:Document?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ws="http://authentidate.de/ws">
  <soapenv:Header />
  <soapenv:Body>
    <ws:signVerifyRequest>
      <ws:ServiceRequestData>
        <ws:Document>JVBiRi0xLjMKMyAwIG9iago8PC9UeXBlic9QYWdlCi9QYXJlbnQgM
          SAwIFIKL1Jlc291cmNlcYAyIDAgUgovQ29udGVudHMgNCAwIFI+PgplbmRvYmoKNC
          AwIG9iago8PC9GaWx0ZXIgL0ZsYXRlRGVjb2RlIC9MZW5ndGggMjEzNT4+CnN0cmV
        </ws:Document>
        <ws:DocumentName>R_1639_37480.pdf</ws:DocumentName>
        <ws:ServiceRequest>SIGN</ws:ServiceRequest>
      </ws:ServiceRequestData>
    </ws:signVerifyRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

 Mehrere Processing Instructions werden immer als einzelne Elemente nacheinander aufgeführt.

6.19.6.4 Webservice Security



Web Service Security: Verschlüsselt, signiert und entschlüsselt eine SOAP-Nachricht oder fügt dieser einen Benutzer-Token oder einen Zeitstempel hinzu. Der Adapter erwartet als Input eine gültige SOAP-Nachricht und gibt eine gemäß der gewählten Operation veränderte SOAP-Nachricht aus.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Encode</i>: Nachricht verschlüsseln • <i>Decode</i>: Nachricht entschlüsseln • <i>Sign Message</i>: Nachricht signieren • <i>Add Username Token</i>: Benutzernamens-Token hinzufügen • <i>Add Timestamp</i>: Zeitstempel hinzufügen
------------------	---

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.wssecurity.WSSecurityAdapter</i> : Hauptklasse (Standard)</p>
<i>cryptoProviderClass</i>	<p>Klassennamen für den WSS4J-Crypto-Provider (Provider-Klasse muss das Interface <code>org.apache.ws.security.components.crypto.Crypto</code> implementieren)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebiger Klassenname • <i>org.apache.ws.security.components.crypto.Merlin</i>: Merlin-Crypto-Provider (Standard)
<i>keystoreType</i>	<p>Keystore-Typ des Crypto-Providers</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebiger Keystore-Typ • <i>jks</i>: Java Keystore (Standard)
<i>keystorePassword</i>	<p>Passwort des Keystores</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>keystoreAlias</i>	<p>Alias-Name des Keystores</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>aliasPassword</i>	<p>Alias-Passwort des Keystores</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>keystoreFile</i>	<p>Pfad zur Keystore-Datei</p>

<i>encodingAlgorithm</i>	<p>Algorithmus für die symmetrische Verschlüsselung einer Nachricht</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Triple DES: Data Encryption Standard</i> -Dreifachverschlüsselungs-Algorithmus verwenden • <i>AES 128: Advanced Encryption Standard</i> -Algorithmus mit 128 Bit Schlüssellänge verwenden • <i>AES 192: Advanced Encryption Standard</i> -Algorithmus mit 192 Bit Schlüssellänge verwenden • <i>AES 256: Advanced Encryption Standard</i> -Algorithmus mit 256 Bit Schlüssellänge verwenden
<i>canonicalization</i>	<p>Kanonisierungs-Algorithmus für die Enkodierung/Signatur gemäß http://www.w3.org/TR/xml-exc-c14n/ oder http://www.w3.org/TR/xml-c14n</p>
<i>keyEncodingAlgorithm</i>	<p>Symmetrischer Verschlüsselungs-Algorithmus für den Schlüssel</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>RSA15</i>: RSA 15 • <i>RSA OEP</i>: RSA OEP
<i>keyType</i>	<p>Schlüssel-ID-Typ gemäß <i>WS Security X.509 certificate token profile</i></p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ISSUER SERIAL</i>: Aussteller und Seriennummer • <i>BST DIRECT REFERENCE</i>: Binärer Sicherheits-Token • <i>SKI KEY IDENTIFIER</i>: Subject Key Identifier • <i>X509 KEY IDENTIFIER</i>: X.509-Aussteller und Seriennummer
<i>signatureAlgorithm</i>	<p>Signatur-Algorithmus</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>RSA with SHA</i>: RSA mit SHA verwenden • <i>DSA with SHA1 (DSS)</i>: DSA mit SHA1(DSS) verwenden
<i>actor</i>	<p>Von der Sicherheitsengine verwendeter Akteur/Rolle gemäß SOAP-Spezifikation</p>

<i>keyAlias</i>	Schlüssel- oder Zertifikats-Alias im Keystore Mögliche Werte: Beliebige Zeichenkette
<i>keyPassword</i>	Passwort für den Keystore-Alias Mögliche Werte: Beliebige Zeichenkette
<i>parts</i>	Nachrichten-Teile, die verschlüsselt oder signiert werden sollen Mögliche Werte: Zeichenkette mit Element-Name, Namensraum und der Verschlüsselungsmethode in Anführungszeichen, durch Semikolon getrennt Beispiel: <i>"Element";"";"Content"</i>
<i>embeddedName</i>	Embedded Name Mögliche Werte: Beliebige Zeichenkette

6.19.6.5 WebService Starter (veraltet)

In der X4 Suite wurden bislang zwei weitere WebService Starter zur Verfügung gestellt:

- [WebService Starter 1](#)
- [WebService Starter 2](#)

Der *WebService Starter 1* erwartet einen SOAP-Request via HTTP und startet daraufhin einen Webservice. Für jede Webservice-Methode ist ein technischer Prozess definiert, der beim Aufruf der Methode ausgeführt wird.

Der *WebService Starter 2* erwartet einen SOAP-Request via HTTP und startet daraufhin einen Webservice, der auf Basis eines XML-Schema-Dokuments (XSD) auf dem X4 Server bereitgestellt wurde.

6.19.6.5.1 WebService Starter 1

Der *WebService Starter 1* erwartet einen SOAP-Request via HTTP und startet daraufhin einen Webservice. Für jede Webservice-Methode ist ein technischer Prozess definiert, der beim Aufruf der Methode ausgeführt wird.

Um einen Webservice zu starten, senden Sie einen SOAP-Request an eine URL nach folgendem Muster:

```
http://<server>/<Instanzname>/httpstarter/WS1/<Benutzer-ID>/<Repository-Pfad>
```

Beispiel: `http://localhost:8080/X4/httpstarter/WS1/1/Project/Directory`

SOAP Request

Der SOAP Request enthält die relevanten Informationen über die aufgerufene(n) Methode(n) und ggf. vorhandene Parameter, z.B. für einen Beispiel-Webservice DemoWebService mit einem Parameter Name:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://DemoWebService.generated.ws.integration.softproject.de/xsd">
  <soapenv:Header/>
  <soapenv:Body>
    <xsd:demoMethod>
      <xsd:Name>X4Experience User</xsd:Name>
    </xsd:demoMethod>
  </soapenv:Body>
</soapenv:Envelope>
```

Unterstützung von SOAP MTOM

Der *WebService Starter 1* unterstützt zudem *SOAP MTOM* (*Message Transmission Optimization Mechanism*), mit dem sich große Dokumente bzw. Binärdaten in MTOM-basierenden Webservices über XOP (XML-binary Optimized Packaging) performant übertragen lassen.

Gesteuert wird die MTOM-Funktion über Processing Instructions, die dem SOAP-Request- oder -Response-Dokument zum Aktivieren von MTOM auf Eingangsseite hinzugefügt werden:

- `<?useMTOM?>`: MTOM-Funktion für diesen SOAP-Request aktivieren
- `<?MTOM-xpath 0-n XPath-Ausdrücke ?>`: Beliebige XPath-Ausdrücke, um die Elemente zu definieren, die Binärdaten enthalten
- `<?MTOM-size Größe in KB ?>`: Größe der Daten in Kilobyte, ab der die Komprimierung aktiviert wird (Standard: 1 KB)

Mit dem [Webservice\(SOAP\)Connector](#) Adapter können Sie die MTOM-Funktion ebenfalls verwenden, um Webservices aufzurufen. Dabei muss der SOAP-Request oder die SOAP-Response die o.g. Processing Instructions enthalten, beispielsweise:

```

<?useMTOM?>
<?MTOM-xpath soapenv:Envelope/soapenv:Body/ws:signVerifyRequest/ws:Service
  RequestData/ws:Document?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ws="http://authentidate.de/ws">
  <soapenv:Header />
  <soapenv:Body>
    <ws:signVerifyRequest>
      <ws:ServiceRequestData>
        <ws:Document>JVBiRi0xLjMKMyAwIG9iago8PC9UeXBlic9QYWdlCi9QYXJlbnQgM
          SAwIFIKL1Jlc291cmNlcYAyIDAgUgovQ29udGVudHMgNCAwIFI+PgplbmRvYmoKNC
          AwIG9iago8PC9GaWx0ZXIgL0ZsYXRlRGVjb2RlIC9MZW5ndGggMjEzNT4+CnN0cmV
        </ws:Document>
        <ws:DocumentName>R_1639_37480.pdf</ws:DocumentName>
        <ws:ServiceRequest>SIGN</ws:ServiceRequest>
      </ws:ServiceRequestData>
    </ws:signVerifyRequest>
  </soapenv:Body>
</soapenv:Envelope>

```

Außerdem wandelt der *WebService* Adapter MTOM-basierende SOAP-Responses, die ein Webservice liefert, automatisch in Standard-SOAP-Responses um.

Erwartete Ordnerstruktur im Repository

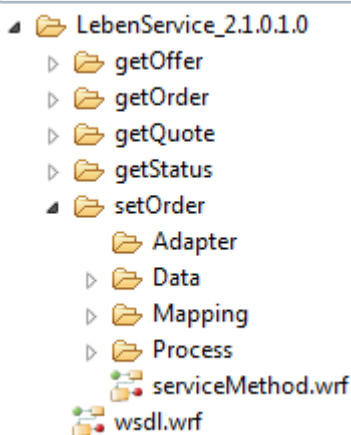
Für jeden aufgerufenen Webservice bzw. Repository-Ordner beachten Sie folgende Regeln:

- Jede Webservice-Methode besitzt einen Unterordner gleichen Namens.
- In jedem Ordner einer Methode befindet sich ein technischer Prozess, der `serviceMethod.wrf` benannt sein muss, und der automatisch durch den Methodenaufruf ausgeführt wird.
- Innerhalb jedes Methoden-Ordners dürfen beliebig viele weitere Repository-Elemente liegen (Prozesse, Adapter etc.), die von `serviceMethod.wrf` verwendet werden können.
- Im Ordner des Webservices liegt ein technischer Prozess `wsdl.wrf`, der die WSDL-Definition ausgibt (siehe unten).



Tipp

Diese Ordnerstruktur kann mit dem X4 Designer über den *WebService Import Wizard* erzeugt werden.



WSDL-Information ausgeben

Der WebService Starter ermöglicht zusätzlich die Ausgabe der Webservice-Definition im WSDL-Format. Rufen Sie hierzu den WebService Starter mit einer URL nach folgendem Schema auf:

```
http://<server>/<Instanzname>/httpstarter/WS/<Benutzer-ID>/<Repository-Pfad>?wsdl
```

Beispiel: `http://localhost:8080/X4/httpstarter/WS/1/Project/Directory?wsdl`


6.19.6.5.2 WebService Starter 2

Der *WebService Starter 2* erwartet einen SOAP-Request via HTTP und startet daraufhin einen Webservice, der auf Basis eines XML-Schema-Dokuments (XSD) auf dem *X4 Server* bereitgestellt wurde, siehe *X4 Entwickler-Referenz – Webservice von einer XSD exportieren*.

Um einen bereitgestellten Webservice zu starten, senden Sie einen kompletten, gültigen SOAP-Request an eine URL nach folgendem Muster:

```
http://<Server>/<X4-Instanz>/httpstarter/WS2/<URL-Part>
```

Beispiel: `http://localhost:8080/X4/httpstarter/WS2/testURLpart`

 Beachten Sie dabei Folgendes:

- Wenn Sie die Webservice-URL per *HTTP POST* aufrufen, müssen Sie den HTTP-Header *SOAPAction* mitgeben und den definierten Operations-Namen exakt angeben. Routing findet ausschließlich über diesen Header-Wert statt.
- Das **I/O Model** *envelope* wird dann gewählt, wenn Sie auf SOAP-Header-Werte zugreifen möchten, z. B. um Sicherheitsmechanismen zu implementieren.

WSDL-Information ausgeben

Der WebService Starter 2 ermöglicht zusätzlich die Ausgabe der aktuellen Webservice-Definition im WSDL-Format. Rufen Sie hierzu den WebService Starter mit einer URL nach folgendem Schema auf:

```
http://<server>/<X4-Instanz>/httpstarter/WS2/<URL-Part>?wsdl
```

Beispiel: `http://localhost:8080/X4/httpstarter/WS2/testURLpart?wsdl`

Webservice-Informationen

Der über den Wizard exportierte Webservice auf Basis eines XML-Schema-Dokuments (XSD) ist nach einem Neustart des X4 Servers direkt verfügbar und wird in `<X4>/X4DB/0/WebServices` in jeweils einer Konfigurationsdatei mit sämtlichen relevanten Informationen abgelegt, z.B. `TestService.xml`:

```

<WebService urlpart="/test"
  namespace="http://www.example.org/TestService" name="TestService">
  <XSD-URL prefix="pre" namespace="http://www.example.org/Content">
    http://www.example.org/Content.xsd</XSD-URL>
  <Operation name="ping">
    <X4 user="1" process="/test/test.wrf"/>
    <Input element="pingRequest"/>
    <Output element="pingResponse"/>
    <IOModel type="content"/>
  </Operation>
</WebService>

```

Dabei gilt Folgendes:

- In Attribut `urlpart` wird der eindeutige URL-Part des Webservices definiert (immer mit führendem /).
- In Attribut `name` wird der Name des Webservices angezeigt.
- Element `<XSD-URL>` ist lediglich für die WSDL-Generierung relevant.
- Für jede Webservice-Operation (Methode) existiert ein Element `<Operation>`, in dem der Operation ein technischer Prozess zugeordnet wird.

Für jede Operation in Element `<Operation>` wird festgelegt:

- Element `<X4>` gibt den Repository-Benutzerordner und den Pfad zum gestarteten technischen Prozess an.
- Die Elemente `<Input>` und `<Output>` sind nur für die WSDL-Generierung relevant und bestimmen hierfür aus der XSD jeweils die Wurzel-Elemente der Input-Daten und der Ausgabe-Daten für den Webservice.
- Element `<IOModel>` definiert in Attribut `type` das *I/O Model* des WebServices.

- ① Je nach verwendetem I/O Model verhält sich der Webservice bei einem Request `<s:Envelope> <s:Header/> <s:Body><Data/> </s:Body> </s:Envelope>` wie folgt:
- I/O Model *content*: Prozess-Input ist `<Data/>` und SOAP-Response ist `<s:Envelope> <s:Header/> <s:Body><Prozess-Output/> </s:Body> </s:Envelope>`.
 - I/O Model *body*: Prozess-Input ist `<s:Body> <Data/> </s:Body>` und SOAP-Response ist `<s:Envelope><s:Header/> <Prozess-Output/> </s:Envelope>`.
 - I/O Model *envelope*: Prozess-Input ist `<s:Envelope> <s:Header/> <s:Body> <Data/> </s:Body> </s:Envelope>` und SOAP-Response ist `<Prozess-Output/>`.

6.19.7 X4 Streaming Support



X4 Streaming Support: Nimmt ein XML oder XML-Fragment (well-nested) und erzeugt daraus ein Dokument, das als Ausgabe für den ReST Starter verwendet werden kann. Innerhalb dieses XML(-Fragments) wird nach `Stream-Elementen` im Namensraum `x4feature:reststream` gesucht. Diese werden in die Ausgabe übernommen, während alle anderen Teile Base64-kodiert werden. Somit

lassen sich XML-Dokumente als ReST-Ausgabe erzeugen in denen ein Stream Base64-kodiert eingebettet ist (z. B. eine SOAP-Response, wenn nicht mit MTOM gearbeitet wird).

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: convertXML: XML konvertieren
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.reststreamsupport.RestStreamSupportAdapter</i> : Hauptklasse (Standard)
<i>inputEncoding</i>	Zeichenkodierung des Input-Dokuments Mögliche Werte: Beliebige gültige Zeichenkodierung (z. B. <i>UTF-8</i> (Standard))

Input

Der Adapter-Input kann wie folgt aufgebaut sein:

```
<?xml version="1.0" encoding="UTF-8"?>
<RootElement>
  <SomeElementBefore>Text</SomeElementBefore>
  <ContainerForTheStream>
    <Stream xmlns="x4feature:reststream" url="xstore://SomeProject/someData.xml"
transcoding="base64-UTF-8"/>
  </ContainerForTheStream>
  <SomeElementAfter>More Text</SomeElementAfter>
</RootElement>
```

Ausgabe

Folgende XML-Struktur wird vom Adapter ausgegeben:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <Status code="200"/>
  <Headers/>
  <Content encoding="base64">PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz4KPF
Jvb3RFbGVtZW50PgoJPFNv
bWVfBgVtZW50QmVmb3JlPlRleHQ8L1NvbWVfBgVtZW50QmVmb3JlPgoJPENvbnRhaW5lckZvc1Ro
ZVN0cmVhbT4=<Stream xmlns="x4feature:reststream" url="xstore://SomeProject/
someData.xml" transcoding="base64-UTF-8"></Stream>
PC9Db250YWluZXJGb3JUaGVtZDhJlYW0+Cgk8U29tZUVsZW1lbnRBZnRlcj5Nb3JlIFRleHQ8L1Nv
bWVfBgVtZW50QWZ0ZXI+CjwvUm9vdEVsZW1lbnQ+
  </Content>
</Response>
```

Diese Ausgabe kann sofort an den ReST-Starter zurückgegeben werden, oder erst noch mit entsprechenden Headern versehen werden.

Inhalt von xstore://SomeProject/someData.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<Hallo><Welt/></Hallo>
```

Ergebnis der ReST-Antwort

```
<?xml version="1.0" encoding="UTF-8"?>
<RootElement>
  <SomeElementBefore>Text</SomeElementBefore>
  <ContainerForTheStream>PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz4NCjxIYW
  xsbz48V2VsdC8+PC9IYWxs4=</ContainerForTheStream>
  <SomeElementAfter>More Text</SomeElementAfter>
</RootElement>
```

i Derzeit gibt es keine Unterstützung wenn ein Stream in ein Attribut eingesetzt werden soll. In diesem Fall muss das entsprechende Element und Attribut mit Base64-Strings im Content-Element zusammengebaut werden.

6.19.8 XOP (MTOM)



XOP (MTOM): Erzeugt XOP/MTOM-Pakete aus XML-Dokumenten, löst XOP-Referenzen aus einem XOP/MTOM-Paket auf und gibt ein XML-Dokument aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>CleanupXml</i>: Entfernt <?xop-candidate?>-Anweisungen und normalisiert deren Elternelement • <i>MakeSimplepart</i>: Entfernt <?xop-candidate?>-Anweisungen und normalisiert deren Elternelement • <i>MakeMultipart</i>: aus einem XML-Dokument ein XOP/MTOM-Paket erzeugen; Die Ausgabestruktur ist ein MIME-XML (siehe auch MIME Multipart Converter) • <i>UnpackMultipart</i>: XOP-Referenzen aus einem MIME-XML, welches ein XOP/MTOM-Paket darstellt, auflösen; Ausgabe ist ein XML-Dokument (die effektive Nutzlast)
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.xop.Adapter</code>: Hauptklasse (Standard)</p>
<i>RequiredMinimumPartSavings</i>	<p>Anzahl der Bytes, die pro Element eingespart werden müssen, damit ein Element via XOP/MTOM ausgelagert wird</p> <ul style="list-style-type: none"> • Beliebige ganze Zahl, z.B. <i>300</i> • 0 (Standard) • <0: Ineffizienzen werden akzeptiert, d.h. es wird akzeptiert, dass mehr Bytes übertragen werden, als wenn nicht ausgelagert wird • <= -300: Die Auslagerung soll unabhängig von den Ersparnissen stattfinden (Auslagern hat einen Overhead von ca. 260–270 Bytes pro Auslagerung)
<i>RequiredMinimumTotalSavings</i>	<p>Legt fest wie viele Bytes insgesamt durch Auslagerungen eingespart werden müssen, damit die Auslagerungen durchgeführt werden.</p> <ul style="list-style-type: none"> • 0 (Standard) • <0: Das Ergebnis-MIME-Dokument darf größer sein als ein Ergebnisdokument, für welches keine Auslagerungen durchgeführt wurden. Mit diesem Parameter kann ein hoher negativer Wert in <i>RequiredMinimumPartSavings</i> relativiert werden.

Statuswerte

1	Die Operation wurde ohne Probleme ausgeführt
0	Die Operation wurde nicht oder nicht vollständig ausgeführt
-1	Fehler bei der Verarbeitung; Wird typischerweise durch ungültige/unvollständige Input-Dokumente ausgelöst.

6.19.8.1

Operation CleanupXml

Die Adapter-Operation *CleanupXml* führt die Normalisierungsarbeiten für die Erstellung eines XOP-Pakets durch. Dabei wird weder ein Kandidat (identifiziert durch `<?xop-candidate?>`) ausgelagert noch wird ein Mime-XML-Dokument erzeugt.

Der *mimetype* des resultierenden X4-Dokuments (siehe [Document Properties Changer](#)) wird wie folgt ermittelt.

Eingehender <i>mimetype</i>	Eingehendes <i>encoding</i>	XML-Dokument ist ...	resultierender <i>mimetype</i>	resultierendes <i>encoding</i>
<i>leer</i>	<i>beliebig</i>	kein SOAP	text/xml	UTF-8
<i>some/type</i>	<i>beliebig</i>	kein SOAP	<i>some/type</i>	UTF-8
<i>beliebig</i>	<i>beliebig</i>	SOAP 1.1	text/xml	UTF-8
<i>beliebig</i>	<i>beliebig</i>	SOAP 1.2	application/soap+xml	UTF-8

Beispiel

Input

```

<SomeXml>
  <blob1>
    <?xop-candidate?>
      SGFs
      bG8=
    </blob1>
    <blob2>
      <?xop-candidate text/plain?>
        SGFs
        bG8g
        SGFs
        bG8=
      </blob2>
    </SomeXml>

```

Output

```

<SomeXml>
  <blob1>SGFsbG8=</blob1>
  <blob2>SGFsbG8gSGFsbG8=</blob2>
</SomeXml>

```

6.19.8.2 Operation MakeSimplepart

Die Adapter-Operation `MakeSimplepart` führt die Normalisierungsarbeiten für die Erstellung eines XOP-Pakets durch, wobei kein Kandidat (identifiziert durch `<?xop-candidate?>`) ausgelagert wird. Das Ergebnis ist ein Mime-XML-Dokument mit nur einem Part (kein Multipart).

Der Mime-Part hat zwei Header:

- `content-transfer-encoding`: `binary`
- `content-type`: Wert siehe unten

Eingehender <i>mimetype</i>	Eingehendes <i>encoding</i>	XML-Dokument ist ...	resultierender <code>content-type</code> Header
<i>leer</i>	<i>beliebig</i>	kein SOAP	<code>text/xml; charset=UTF-8</code>
<i>some/type</i>	<i>beliebig</i>	kein SOAP	<code>some/type; charset=UTF-8</code>
<i>beliebig</i>	<i>beliebig</i>	SOAP 1.1	<code>text/xml; charset=UTF-8</code>
<i>beliebig</i>	<i>beliebig</i>	SOAP 1.2	<code>application/soap+xml; charset=UTF-8</code>

Beispiel

Input

```

<SomeXml>
  <blob1>
    <?xop-candidate?>
      SGFs
      bG8=
    </blob1>
    <blob2>
      <?xop-candidate text/plain?>
        SGFs
        bG8g
        SGFs
        bG8=
      </blob2>
    </SomeXml>

```

Output

```

<Part>
  <Header>
    <Name>content-type</Name>
    <Value>text/xml; charset=UTF-8</Value>
    <Type>text/xml</Type>
    <Parameter>
      <Name>charset</Name>
      <Value>UTF-8</Value>
    </Parameter>
  </Header>
  <Header>
    <Name>content-transfer-encoding</Name>
    <Value>binary</Value>
  </Header>
  <Content encoding="base64">PD94bWwgdGVyc2lvd... more base64 ...g0K</Content>
</Part>

```

6.19.8.3 Operation MakeMultipart

Die Adapter-Operation *MakeMultipart* wird verwendet, um die Übertragung von XML-Dokumenten mit größeren Binär(Base64)-Inhalten zu optimieren. Haupteinsatzgebiet sind SOAP-Webservices, aber auch jede andere XML.

Input: Beispiel einer Input-XML für die Operation MakeMultipart

```
<SomeXml>
  <blob1>
    <?xop-candidate?>
      SGFsbG8=
    </blob1>
  <blob2>
    <?xop-candidate text/plain?>
      SGFsbG8gSGFsbG8=
    </blob2>
</SomeXml>
```

Erläuterung

- Mit der XML-Verarbeitungsanweisung <?xop-candidate?> wird angegeben, welcher Textknoten ausgelagert werden soll
- Wird ein Medien-Typ als Parameter angegeben, z.B.
 <?xop-candidate text/plain?>
 , so wird dieser Medien-Typ verwendet
- Wird kein Medien-Typ als Parameter angegeben, so wird immer application/octet-stream als Medien-Typ angenommen
- Über die Adapter-Parameter RequiredMinimumPartSavings und RequiredMinimumTotalSavings kann gesteuert werden, wie groß die Ersparnisse sein müssen, damit eine Auslagerung aus dem Input-XML stattfindet.

Output: Beispiel eines erzeugten XML-Dokuments mit MakeMultipart

```

<Multipart>
  <Header>
    <Name>content-transfer-encoding</Name>
    <Value>binary</Value>
  </Header>
  <Header>
    <Name>content-type</Name>
    <Value>multipart/related; type="application/xop+xml";
      start="&lt;0@2eba3b7d-7eab-448d-be94-e5dd2bc6e902&gt;";
      start-info="application/xml";
      boundary=3f032702-07f2-46fa-8108-f2258033727e</Value>
    <Type>multipart/related</Type>
    <Parameter>
      <Name>type</Name>
      <Value>application/xop+xml</Value>
    </Parameter>
    <Parameter>
      <Name>start</Name>
      <Value>&lt;0@2eba3b7d-7eab-448d-be94-e5dd2bc6e902&gt;</Value>
    </Parameter>
    <Parameter>
      <Name>start-info</Name>
      <Value>application/xml</Value>
    </Parameter>
    <Parameter>
      <Name>boundary</Name>
      <Value>3f032702-07f2-46fa-8108-f2258033727e</Value>
    </Parameter>
  </Header>
  <Prolog />
  <Part>
    <Header>
      <Name>content-type</Name>
      <Value>application/xop+xml; charset=UTF-8; type="application/xml"
    </Value>
      <Type>application/xop+xml</Type>
      <Parameter>
        <Name>charset</Name>
        <Value>UTF-8</Value>
      </Parameter>
      <Parameter>
        <Name>type</Name>
        <Value>application/xml</Value>
      </Parameter>
    </Header>
    <Header>
      <Name>content-transfer-encoding</Name>
      <Value>binary</Value>
    </Header>
    <Header>
      <Name>content-id</Name>
      <Value>&lt;0@2eba3b7d-7eab-448d-be94-e5dd2bc6e902&gt;</Value>
      <Id asUri="cid:0@2eba3b7d-7eab-448d-be94-e5dd2bc6e902">

```

```

002eba3b7d-7eab-448d-be94-e5dd2bc6e902
</Id>
</Header>
<Content encoding="base64">PD94bWwgdMvYc2lvcj0iMS4wIiB1bmNvZGluZz0iVVRGLTgiPz
4NCjxTb21lWG1sPgoJPGJsb2Ixpjx4b3A6SW5jbHVkZSBocmVmPSJjaWQ6MUAxOGE0ODI3ZC1lZWY3LTQ5NjU
tYmY3OS1lN2YwZDBjYTI5OTQ5IHhtbG5zOnhvcD0iaHR0cDovL3d3dy53My5vcmcvMjAwNC8wOC94b3AvaW5j
bHVkZSIvPjwvYmxvYjE+Cgk8YmxvYjI+PHhvcDpJbmNsdWRlIGhyZWY9ImNpZDoyQDkyYzhlnjM5LTkwZjktN
DJjNC1hMGNhLTNkOTdiODQzZTQ4YSIgeG1sbnM6eG9wPSJodHRwOi8vd3d3LnczLm9yZy8yMDA0LzA4L3hvcC
9pbmNsdWRlIi8+PC9ibG9iMj4KPC9Tb21lWG1sPg0K</Content>
</Part>
<Part>
  <Header>
    <Name>content-type</Name>
    <Value>application/octet-stream</Value>
    <Type>application/octet-stream</Type>
  </Header>
  <Header>
    <Name>content-transfer-encoding</Name>
    <Value>binary</Value>
  </Header>
  <Header>
    <Name>content-id</Name>
    <Value><1@18a4827d-eef7-4965-bf79-e7f0d0ca2994></Value>
    <Id asUrl="cid:1@18a4827d-eef7-4965-bf79-e7f0d0ca2994">
      1@18a4827d-eef7-4965-bf79-e7f0d0ca2994
    </Id>
  </Header>
  <Content encoding="base64">SGFsbG8=</Content>
</Part>
<Part>
  <Header>
    <Name>content-type</Name>
    <Value>text/plain</Value>
    <Type>text/plain</Type>
  </Header>
  <Header>
    <Name>content-transfer-encoding</Name>
    <Value>binary</Value>
  </Header>
  <Header>
    <Name>content-id</Name>
    <Value><2@92c8e639-90f9-42c4-a0ca-3d97b843e48a></Value>
    <Id asUrl="cid:2@92c8e639-90f9-42c4-a0ca-3d97b843e48a">
      2@92c8e639-90f9-42c4-a0ca-3d97b843e48a
    </Id>
  </Header>
  <Content encoding="base64">SGFsbG8gSGFsbG8=</Content>
</Part>
<Epilog />
</Multipart>

```

Dieses Output-XML entspricht einem MIME-Multipart-Dokument in dem sich als erster Part das modifizierte Eingangs-XML befindet, und alle weiteren Parts die extrahierten Base64-Einschlüsse des Input-XMLs sind.

Output: Entsprechendes MIME-Multipart-Dokument

```

content-transfer-encoding: binary
content-type: multipart/related; type="application/xop+xml";
start="<0@2eba3b7d-7eab-448d-be94-e5dd2bc6e902>"; start-info="application/xml";
boundary=3f032702-07f2-46fa-8108-f2258033727e

--3f032702-07f2-46fa-8108-f2258033727e
content-type: application/xop+xml; charset=UTF-8; type="application/xml"
content-transfer-encoding: binary
content-id: <0@2eba3b7d-7eab-448d-be94-e5dd2bc6e902>

<SomeXml>
  <blob1><xop:Include href="cid:1@18a4827d-eef7-4965-bf79-e7f0d0ca2994"
xmlns:xop="http://www.w3.org/2004/08/xop/include"/></blob1>
  <blob2><xop:Include href="cid:2@92c8e639-90f9-42c4-a0ca-3d97b843e48a"
xmlns:xop="http://www.w3.org/2004/08/xop/include"/></blob2>
</SomeXml>

--3f032702-07f2-46fa-8108-f2258033727e
content-type: application/octet-stream
content-transfer-encoding: binary
content-id: <1@18a4827d-eef7-4965-bf79-e7f0d0ca2994>

Hallo
--3f032702-07f2-46fa-8108-f2258033727e
content-type: text/plain
content-transfer-encoding: binary
content-id: <2@92c8e639-90f9-42c4-a0ca-3d97b843e48a>

Hallo Hallo
--3f032702-07f2-46fa-8108-f2258033727e--


```

Wenn keine Auslagerungen stattgefunden haben, weil z.B. keine Markierungen existieren, oder nicht genug Ersparnisse zusammen gekommen sind, liefert der Adapter den Status 0 für diese Operation.

6.19.8.4 Operation UnpackMultipart

Die Adapter-Operation *UnpackMultipart* entspricht der inversen Operation zu *MakeMultipart*. Sie löst aus einem MIME-XML, das ein XOP/MTOM-Paket darstellt, die *xop:Include* Elemente auf. Als Ergebnis kommt das Nutzlast-XML des XOP-Pakets (z.B. der SOAP-Envelope) heraus, wobei die *xop:Include* Elemente durch die entsprechenden Base64-Strings ersetzt wurden.

Die Operation akzeptiert auch ein MIME-XML, welches nur ein Part ist. In diesem Fall wird das Part einfach nur ausgepackt. Dann wird jedoch nicht garantiert, dass das Ergebnis ein XML ist, z.B. wird ein Part reinschicken das ein PDF darstellt, ist das Ergebnis das PDF.

 Bei den *xop:Include*-Referenzen wird nur das URL-Protokoll *cid*: unterstützt. Die so referenzierten Teile müssen sich im selben Multipart befinden wie das Root-Part, d.h. sie müssen Geschwister-Knoten des Root-Parts sein.

Input: Beispiel Input für die Operation UnpackMultipart

```

<Multipart>
  <Header>
    <Name>content-transfer-encoding</Name>
    <Value>binary</Value>
  </Header>
  <Header>
    <Name>content-type</Name>
    <Value>multipart/related; type="application/xop+xml";
      start="&lt;0@2eba3b7d-7eab-448d-be94-e5dd2bc6e902&gt;";
      start-info="application/xml";
      boundary=3f032702-07f2-46fa-8108-f2258033727e</Value>
    <Type>multipart/related</Type>
    <Parameter>
      <Name>type</Name>
      <Value>application/xop+xml</Value>
    </Parameter>
    <Parameter>
      <Name>start</Name>
      <Value>&lt;0@2eba3b7d-7eab-448d-be94-e5dd2bc6e902&gt;</Value>
    </Parameter>
    <Parameter>
      <Name>start-info</Name>
      <Value>application/xml</Value>
    </Parameter>
    <Parameter>
      <Name>boundary</Name>
      <Value>3f032702-07f2-46fa-8108-f2258033727e</Value>
    </Parameter>
  </Header>
  <Prolog />
  <Part>
    <Header>
      <Name>content-type</Name>
      <Value>application/xop+xml; charset=UTF-8; type="application/xml"
    </Value>
      <Type>application/xop+xml</Type>
      <Parameter>
        <Name>charset</Name>
        <Value>UTF-8</Value>
      </Parameter>
      <Parameter>
        <Name>type</Name>
        <Value>application/xml</Value>
      </Parameter>
    </Header>
    <Header>
      <Name>content-transfer-encoding</Name>
      <Value>binary</Value>
    </Header>
    <Header>
      <Name>content-id</Name>
      <Value>&lt;0@2eba3b7d-7eab-448d-be94-e5dd2bc6e902&gt;</Value>
      <Id asUrl="cid:0@2eba3b7d-7eab-448d-be94-e5dd2bc6e902">

```

```

002eba3b7d-7eab-448d-be94-e5dd2bc6e902
</Id>
</Header>
<Content encoding="base64">PD94bWwgdmVyc2lvcj0iMS4wIiB1bmNvZGluZz0iVVRGLTgiPz
4NCjxTb21lWG1sPgoJPGJsb2Ixpjx4b3A6SW5jbHVkZSBocmVmPSJjaWQ6MUAxOGE0ODI3ZC1lZWY3LTQ5NjU
tYmY3OS1lN2YwZDBjYTI5OTQ5IHhtbG5zOnhvcD0iaHR0cDovL3d3dy53My5vcmcvMjAwNC8wOC94b3AvaW5j
bHVkZSIvPjwvYmxvYjE+Cgk8YmxvYjI+PHhvcDpJbmNsdWRLIGhyZWY9ImNpZDoyQDkyYzhlnjM5LTkwZjktN
DJjNC1hMGNhLTNkOTdiODQzZTQ4YSIgeG1sbnM6eG9wPSJodHRwOi8vd3d3LnczLm9yZy8yMDA0LzA4L3hvcC
9pbmNsdWRLIi8+PC9ibG9iMj4KPC9Tb21lWG1sPg0K
</Content>
</Part>
<Part>
  <Header>
    <Name>content-type</Name>
    <Value>application/octet-stream</Value>
    <Type>application/octet-stream</Type>
  </Header>
  <Header>
    <Name>content-transfer-encoding</Name>
    <Value>binary</Value>
  </Header>
  <Header>
    <Name>content-id</Name>
    <Value><1@18a4827d-eef7-4965-bf79-e7f0d0ca2994></Value>
    <Id asUrl="cid:1@18a4827d-eef7-4965-bf79-e7f0d0ca2994">
      1@18a4827d-eef7-4965-bf79-e7f0d0ca2994
    </Id>
  </Header>
  <Content encoding="base64">SGFsbG8=</Content>
</Part>
<Part>
  <Header>
    <Name>content-type</Name>
    <Value>text/plain</Value>
    <Type>text/plain</Type>
  </Header>
  <Header>
    <Name>content-transfer-encoding</Name>
    <Value>binary</Value>
  </Header>
  <Header>
    <Name>content-id</Name>
    <Value><2@92c8e639-90f9-42c4-a0ca-3d97b843e48a></Value>
    <Id asUrl="cid:2@92c8e639-90f9-42c4-a0ca-3d97b843e48a">
      2@92c8e639-90f9-42c4-a0ca-3d97b843e48a
    </Id>
  </Header>
  <Content encoding="base64">SGFsbG8SGFsbG8=</Content>
</Part>
<Epilog />
</Multipart>

```

i Output ist ein XML (oder ein beliebiges Dokument, wenn ein MIME-Part-XML hineingeschickt wird)

Output: Beispiel einer Output-XML

```
<SomeXml>
  <blob1>SGFsbG8=</blob1>
  <blob2>SGFsbG8gSGFsbG8=</blob2>
</SomeXml>
```

Wenn nicht alle `xop:Include`-Element ersetzt werden konnten, weil die Referenzen nicht aufgelöst werden konnten, gibt der Adapter den Status 0 zurück und die Ausgabe ist das Nutzlast-XML mit allen Ersetzungen, die durchgeführt werden konnten.

Ein Status von -1 wird verwendet, um verschiedene Fehler in dem MIME-XML anzuzeigen.

6.19.9 YAML Converter

YAML Converter: Konvertiert ein YAML-Dokument in ein XML-Dokument und umgekehrt.

Eigenschaften

Operation	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • Convert YAML to XML: Konvertiert ein YAML-Dokument in ein XML-Dokument • Convert XML to YAML: Konvertiert ein XML-Dokument in ein YAML-Dokument
-----------	--

Parameter

Adapter	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.x4.adapter.yaml.YamlAdapter</i> : Hauptklasse (Standard)
processStrings	Definiert, ob numerische Werte und boolean Werte vom Adapter gelesen und normiert werden (<i>true</i>) oder als literale Zeichenketten behandelt werden (<i>false</i>). Mögliche Werte: <ul style="list-style-type: none"> • <i>true</i>: Numerische Werte und boolean Werte werden gelesen und normiert • <i>false</i>: Numerische Werte und boolean Werte werden als literale Zeichenketten behandelt

Statuswerte

1	Die Konvertierung war erfolgreich und eine Ausgabe wird erzeugt
-1	Während der Konvertierung kam es zu einem Fehler und es wird keine Ausgabe erzeugt

Input

Je nach gewählter Operation erwartet der Adapter folgende Input-Struktur:

- Convert YAML to XML:

YAML Input

```
example:
  element1:
    - value1
    - value2
    - childValue1: 1
      childValue2: 2
  element2: value3
```

- Convert XML to YAML:

XML Input

```
<Object>
  <Object name="example">
    <Array name="element1">
      <Value>value1</Value>
      <Value>value2</Value>
      <Object>
        <Value name="childValue1">1</Value>
        <Value name="childValue2">2</Value>
      </Object>
    </Array>
    <Value name="element2">value3</Value>
  </Object>
</Object>
```

Ausgabe

Je nach gewählter Operation gibt der Adapter folgend Struktur aus:

- Convert YAML to XML:

XML Output

```
<Object>
  <Array name="element1">
    <Value>value1</Value>
    <Value>value2</Value>
    <Object>
      <Value name="childValue1">1</Value>
      <Value name="childValue2">2</Value>
    </Object>
  </Array>
  <Value name="element2">value3</Value>
</Object>
```

- Convert XML to YAML:

YAML Output

```
element1:
- value1
- value2
- {childValue1: 1, childValue2: 2}
element2: value3
```

6.20 XML Tools

6.20.1 Inline Content Expander



Inline Content Expander: Klappt Datensegmente in einem XML-Dokument auf Basis von definierten Parametern ein und aus.

Eigenschaften

Operation	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <ul style="list-style-type: none"> • Expand: In einem XML-Dokument eingebettete Datensegmente ausklappen. Die Datensegmente enthalten XML-Daten als reinen Text oder Base64-kodierte Zeichenketten und werden innerhalb des Input-XML-Dokuments ausgeklappt und eingebettet. • Collapse: Teile eines XML-Dokuments als reinen Text oder Base64-kodierte Zeichenketten einklappen. Die eingeklappten Datensegmente werden in das Input-XML-Dokument eingebettet.
-----------	---

Parameter

Adapter	<p>Main adapter class (do not change!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.inlinecontentexpanderAdapter</code>: Hauptklasse (Standard)</p>
select	<p>XPath-Ausdruck zur Auswahl aus den Input-XML-Dokument</p> <p>Mögliche Werte: Beliebiger gültiger XPath-Ausdruck (z.B. <code>Root/Element[1]</code>)</p>
Format	<p>Format der Datensegmente</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • BASE64: Base64-kodierte Daten (Standard) • STRING: Textdaten
Output	<p>Ausgabeformat der Datensegmente</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • XML_ONLY: Es werden nur gültige XML-Strukturen verarbeitet. Bei ungültigen Strukturen wird ein Fehler ausgegeben. (Standard) • XML_OR_STRING: XML-Strukturen werden verarbeitet, wenn sie gültig sind, andernfalls werden die Daten als Zeichenkette ausgegeben. • STRING_ONLY: Es werden nur Zeichenketten verarbeitet.
encoding	<p>Zeichenkodierung der Datensegmente</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige gültige Zeichenkodierung (z.B. <code>windows-1252</code>) • <code>UTF-8</code>: UTF-8 (Standard)

Statuswerte

0	Der XPath-Ausdruck referenziert kein Element. Das Dokument wird unverändert ausgegeben.
1	Der Adapter wurde erfolgreich ausgeführt.
-1	Ein Fehler trat bei der Ausführung des Adapters auf. Je nach Operation wird entweder das Dokument unverändert ausgegeben (Collapse) oder die entsprechende Fehlermeldung ausgegeben (Expand).

Input

Der Adapter erwartet ein beliebiges gültiges XML-Dokument als Input.

Beispiel Input-XML-Dokument

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Headers>
    <Header>
      <Name>Content-Type</Name>
      <Value>application/xml</Value>
    </Header>
    <Header>
      <Name>Accept</Name>
      <Value>application/xml</Value>
    </Header>
  </Headers>
  <Body><Content>
    <Description>Adress succesfully submitted!</Description>
    <Data>
      <root>
        <address>
          <name>John Doe</name>
          <street>42 main avenue</street>
          <zip>13579</zip>
          <city>example town</city>
          <state>example state</state>
          <country>example country</country>
        </address>
      </root>
    </Data>
  </Content></Body>
</Request>
```

i Gesetzte Adapter-Parameter:

- Operation: *Collapse*
- select: *//Body*
- Format: *BASE64*
- Output: *XML_ONLY*
- encoding: *UTF-8*

Ausgabe


Der Adapter gibt ein XML-Dokument mit eingebetteten verarbeiteten Daten aus.

Beispiel-Output

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Headers>
    <Header>
      <Name>Content-Type</Name>
      <Value>application/xml</Value>
    </Header>
    <Header>
      <Name>Accept</Name>
      <Value>application/xml</Value>
    </Header>
  </Headers>
  <Body>
    >PENvbnRlbnQ+DQoJCQk8RGVzY3JpcHRpb24+QWRyZXNzIHN1Y2Nlc2Z1bGx5IHN1Ym1pdHRlZCE8L0Rlc2Ny
    aXB0aW9uPg0KCQkJPERhdGE+DQoJCQkJPHJvb3Q+DQoJCQkJCTxhZGRyZXNzPg0KCQkJCQkJPG5hbWU+Sm9ob
    iBEb2U8L25hbWU+DQoJCQkJCQk8c3RyZWV0PjQyIG1haW4gYXZlbnVlPC9zdHJlZXQ+DQoJCQkJCQk8emlwPj
    EzNTc5PC96aXA+DQoJCQkJCQk8Y2l0eT5leGFtcGxlIHRvd248L2NpdHk+DQoJCQkJCQk8c3RhZGU+ZXhnbXB
    sZSBzdGF0ZTwwc3RhZGU+DQoJCQkJCQk8Y291bnRyeT5leGFtcGxlIGNvdW50cnk8L2NvdW50cnk+DQoJCQkJ
    CTwwYWRkcmVzcz4NCgkJCQk8L3Jvb3Q+DQoJCQk8L0RhdGE+DQoJCTwwQ29udGVudD4=
  </Body>
</Request>
```

6.20.2 Record Counter



 **Record Counter:** Zählt Datensätze in den Input-Daten, die anhand eines Trennzeichens bestimmt werden. Dieser Adapter behandelt alle Input-Daten als Text und gibt ein XML-Dokument aus, das die Anzahl der Datensätze enthält.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Count</i> : Datensätze in den Input-Daten zählen
------------------	---

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.RecordCounter</i>: Hauptklasse (Standard)</p>
<i>RecordID</i>	<p>Verwendetes Trennzeichen (oder Escape-Sequenz)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige(s) Zeichen (z.B. ,) • \n: Zeilenumbruch als Trennzeichen interpretieren (Standard)

Ausgabe

Der erfolgreich ausgeführte *Record Counter* gibt ein XML-Dokument mit folgendem Muster zurück:

```
<RecordCount><!--Anzahl der Datensätze--></RecordCount>
```

6.20.3 XHTML Form Extractor



XHTML Form Extractor: Extrahiert Formularfelder aus XML-kompatiblen (X)HTML-Dokumenten, um das Mapping von Prozessdaten auf Formularfelder übersichtlicher zu gestalten. Der Adapter erwartet als Input ein (X)HTML-Dokument (in jedem Fall ein wohlgeformtes XML-Dokument!) und gibt ein XML-Dokument mit der Struktur des extrahierten (X)HTML-Formulars aus. Zudem kann der Adapter über Operation `FillInFields` die adapterspezifische XML-Struktur, die gesetzte Formularfelder und ggf. veränderte Eigenschaften für Optionsfelder enthält, in eine (X)HTML-Formularvorlage einfüllen und das fertige (X)HTML-Formulardokument ausgeben.

i Dieser Adapter richtet sich an Entwickler von *Human Workflow*-Prozessen, die bestehende (X)HTML-Formularvorlagen zur Laufzeit mit Daten befüllen möchten. Hierzu wird in einem XSL-Mapping die Input-Datenstruktur (z. B. die Ergebnisse einer Datenbankabfrage) über den Mapping Editor mit den Feldern des (X)HTML-Formulars verknüpft.

Um (X)HTML-Dokumentstrukturen für das Mapping übersichtlicher zu gestalten, extrahiert der XHTML Form Extractor Adapter alle Formularelemente, sodass im Mapping Editor nur diese vereinfachte Formularstruktur und die Daten verknüpft werden müssen. Zur Konvertierung in die XML-Formularstruktur ist im *X4 Designer* die Kontextmenüfunktion **Extract XHTML Form** für (X)HTML-Repository-Dokumente verfügbar, die diesen Adapter nutzt.

Um die XML-Formularstruktur nach dem Mapping wieder in ein vollständiges (X)HTML-Formular umzuwandeln, den XHTML Form Extractor Adapter als nächsten Prozess-Schritt platzieren, die Operation `FillInFields` wählen und in Adapter-Parameter `templateUrl` die URL zur (X)HTML-Formularvorlage angeben. Bei der Ausführung des Prozesses werden somit die veränderten Formularelemente mit der (X)HTML-Formularvorlage zusammengeführt und zur Laufzeit ausgegeben, z. B. zur Anzeige in Web-Anwendungen. Über Attribute in der XML-Formularstruktur lassen sich Optionsfelder (de)selektieren und (de)aktivieren sowie CSS-Direktformatierungen und CSS-Klassenzuordnungen definieren.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none">• <i>ExtractFields</i>: Formularfelder aus einem Input-(X)HTML-Dokument (wohlgeformtes XML!), extrahieren• <i>FillInFields</i>: Formularfelder aus der adapterspezifischen XML-Struktur in ein XML-konformes (X)HTML-Dokument einfügen, das in Parameter <code>templateUrl</code> definiert wurde; zusätzliche Felder aus dem (X)HTML-Vorlagendokument werden unverändert übernommen
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.htmlformextractor.FormAdapter</code>: Hauptklasse (Standard)</p>
<i>templateUrl</i>	<p>URL zum Vorlagendokument des (X)HTML-Formulars (benötigt für Operation <i>FillInFields</i>)</p> <p>Mögliche Werte: Beliebige gültige URL (z.B. <code>xstore://Project/Folder/Form.xhtml</code>)</p>

Beispiel

In diesem Beispiel dient ein HTML5-Formular (wohlgeformtes XML) als Input:

```

<!DOCTYPE HTML>
<html>
  <head>
    <title>My Form</title>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8"/>
    <style type="text/css">
      body { background-color:#F0F0F0; font-family:sans-serif }
    </style>
  </head>
  <body>
    <h1>Form</h1>
    <form action="ACTION" method="post" id="myID">
      <p>
        <input type="text" name="myName" value=""/>
        <select name="myDropdown" size="1">
          <option>1</option>
          <option>2</option>
          <option>3</option>
        </select>
        <input type="radio" name="myOptions" value="either" /> Either
        <input type="radio" name="myOptions" value="or" /> Or
        <input type="submit" value="submit" name="mySend"/>
      </p>
    </form>
  </body>
</html>

```

Bei Operation ExtractFields wandelt der Adapter das HTML5-Formular in folgende XML-Struktur um:

```

<HtmlFields>
  <title>My Form</title>
  <form_myID id="myID">
    <text_myName name="myName" type="text"/>
    <submit_mySend name="mySend" type="submit">submit</submit_mySend>
    <singleselect_myDropdown name="myDropdown">
      <option disabled="false" selected="false">1</option>
      <option disabled="false" selected="false">2</option>
      <option disabled="false" selected="false">3</option>
    </singleselect_myDropdown>
  </form_myID>
</HtmlFields>

```

Über ein XSL-Mapping wurde nun folgende XML-Struktur erzeugt, in der Feldwerte und der Seitentitel geändert sowie verschiedene Eigenschaften für Optionsfelder gesetzt sind und ein zusätzliches Optionsfeld ergänzt wurde.


```

<HtmlFields>
  <title>X4 Experience</title>
  <form_myID id="myID">
    <text_myName type="text" name="myName">X4 Experience</text_myName>
    <submit_mySend type="submit" name="mySend">submit</submit_mySend>
    <singleselect_myDropdown name="myDropdown">
      <option>Version 4.0</option>
      <option selected="false" disabled="true">Version 4.1</option>
      <option selected="false" disabled="false" cssStyle="color:blue">Version 4.2
    </option>
      <option selected="true" disabled="false">Version 4.3</option>
    </singleselect_myDropdown>
  </form_myID>
</HtmlFields>

```

Mit Operation FillInFields wandelt der Adapter die oben genannte modifizierte XML-Struktur wieder in ein vollständiges, XML-konformes HTML5-Formular um:

```

<!DOCTYPE HTML>
<html>
  <head>
    <title>X4 Experience</title>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8"/>
    <style type="text/css">
      body { background-color:#F0F0F0; font-family:sans-serif }
    </style>
  </head>
  <body>
    <h1>Form</h1>
    <form action="ACTION" method="post" id="myID">
      <p>
        <input type="text" name="myName" value="X4 Experience"
          style="color:blue"/>
        <select name="myDropdown" size="1">
          <option>Version 4.0</option>
          <option disabled="disabled">Version 4.1</option>
          <option style="color:blue">Version 4.2</option>
          <option selected="selected">Version 4.3</option>
        </select>
        <input type="radio" name="myOptions" value="either"/> Either
        <input type="radio" name="myOptions" value="or"/> Or
        <input type="submit" value="submit" name="mySend"/>
      </p>
    </form>
  </body>
</html>

```

6.20.3.1

Operation: ExtractFields

Mit Operation ExtractFields lassen sich Formularfelder aus einem Input-(X)HTML-Dokument (wohlgeformtes XML!), extrahieren.

Input

Bei Operation `ExtractFields` erwartet der Adapter ein (X)HTML-Dokument, das eine wohlgeformte XML-Struktur aufweist (korrekte Verschachtelung, selbstschließende leere Elemente wie `
` etc.) und dessen Formularelemente über Attribut `id` oder `name` eindeutig sind.

Ausgabe

Bei Operation `ExtractFields` gibt der Adapter ein XML-Dokument mit Formulardaten aus, das Sie im Mapping als Vorlage für die Zielstruktur verwenden können.

6.20.3.2 Operation: `FillInFields`

Mit Operation `FillInFields` lassen sich Formularfelder aus der adapterspezifischen XML-Struktur in ein XML-konformes (X)HTML-Dokument einfügen, das in Parameter `templateUrl` definiert wurde; zusätzliche Felder aus dem (X)HTML-Vorlagendokument werden unverändert übernommen.

Input

Bei Operation `FillInFields` erwartet der Adapter ein XML-Dokument, das eine Struktur nach folgendem Muster aufweist und für alle Formularfelder (außer Checkboxen und Radio-Buttons) ein XML-Element erzeugt, das `Feldtyp` und `Feldname` im Namen enthält.

In der XML-Struktur können `<option/>`-Felder über spezifische Attribute mit CSS-Direktformatierungen (`cssStyle="..."`), CSS-Klassenzuordnungen (`cssClass="..."`) und Labels (`label="..."`) versehen sowie (de)selektiert (`selected="true/false"`), (de)aktiviert (`disabled="true/false"`) und auf nur lesend gesetzt (`readonly="true/false"`) werden. Werte von `<input/>` und `<option/>` lassen sich über ein Attribut `value="..."` setzen.

```
<HtmlFields>
  <title><!--Seitentitel des (X)HTML-Dokuments--></title>
  <!-- Beliebig viele Formulare (eindeutig über ID identifiziert) -->
  <form_id id="Formular-ID">
    <!-- Beliebig viele Formularfelder (identifiziert über ID oder Name) -->
    <Feldtyp_Feldname/-ID name="Feldname" type="Feldtyp"><!--Feldwert--></
Feldtyp_Feldname/-ID>
  </form_id>
</HtmlFields>
```

Ausgabe

Bei Operation `FillInFields` gibt der Adapter ein XML-kompatibles (X)HTML-Dokument aus, das die Feldwerte (und bei Optionsfeldern ggf. geänderte Eigenschaften) enthält, die im Input-Dokument definiert wurden und dessen Struktur der in Parameter `templateUrl` definierten Vorlage entspricht. Checkboxen und Radio-Buttons aus der Vorlage werden unverändert in das Ergebnisdokument übernommen.

6.20.4 XML Checker



XML Checker: Prüft, ob das Input-Dokument ein XML-Dokument ist, und gibt das Dokument sowie einen entsprechenden Status aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>checkXML</i> : Input-Dokument prüfen
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.typechecker.TypeCheckerAdapter</i> : Hauptklasse (Standard)
----------------	---

Statuswerte

0	Das geprüfte Input-Dokument ist kein XML-Dokument.
1	Das geprüfte Input-Dokument ist ein wohlgeformtes XML-Dokument.

6.20.5 XML Element Wrapper



XML Element Wrapper: Umschließt die Input-Daten mit einem bestimmten XML-Wurzelement und gibt ein UTF-8-kodiertes XML-Dokument aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>wrap</i> : Input-Daten als XML-Dokument mit Wurzelement aus Parameter name ausgeben
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.elementwrapper.ElementWrapperAdapter</i> : Hauptklasse (Standard)
<i>name</i>	Name des Wurzelements Mögliche Werte: <ul style="list-style-type: none"> • Beliebiger gültiger XML-Elementname • <i>Data</i>: Wurzelement <i>Data</i> (Standard)

6.20.6 XML Formatter



XML Formatter: Stellt Formatierungsfunktionen für XML-Dokumente bereit, z.B. Zeilenumbrüche und Einrückungen.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: • <i>PrettyPrint</i> : XML-Dokument formatieren
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.xmlformatter.XMLFormatterAdapter</i> : Hauptklasse (Standard)
<i>indentSize</i>	Anzahl der Einrückungen innerhalb des XML-Dokuments Mögliche Werte: Beliebige ganze Zahl, z. B. 4

Statuswerte

<i>1(successful)</i>	Die Anfrage war erfolgreich und ein Ergebnis wurde zurückgegeben
<i>-1(failed)</i>	Die Anfrage schlug aufgrund eines technischen Fehlers fehl

Input

Der Adapter erwartet als Input ein unformatiertes XML-Dokument.

Ausgabe

Der Adapter gibt ein formatiertes XML-Dokument zurück.

6.20.7 XML Iterator



XML Iterator: Liest ein XML-Dokument von einer beliebigen gültigen URL schrittweise in einen technischen Prozess ein und gibt bei jeder Iteration ein Kindelement (inklusive sämtlicher Kindknoten) des Wurzel-Elements als XML-Dokument aus

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Iterate</i>: Kindknoten des Wurzel-Elements einlesen • <i>Clear</i>: Iterator eines <i>XML Iterators</i> zurücksetzen, der dieselbe ID besitzt wie der gewünschte <i>XML Iterator</i> mit der Operation <i>Iterate</i> (siehe Parameter <i>id</i>)
------------------	---

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <code>de.softproject.integration.adapter.xmliterator.XmlIterator</code>: Hauptklasse (Standard)</p>
<i>url</i>	<p>URL zum einzulesenden XML-Dokument</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige gültige URL (z.B. <code>xstore://Project/Folder/File.xml</code>) • URL zu einem UNC-Netzlaufwerk (z.B. <code>file://\\\\storage/share_data/Folder/File.xml</code>)
<i>id</i>	<p>Eindeutige ID des <i>XML Iterator</i>-Bausteins innerhalb des Prozess-Kontextes bzw. der Prozess-Instanz (wenn Sie mehrere voneinander unabhängig iterierende <i>XML Iterator Adapter</i> in demselben Prozess-Kontext verwenden möchten)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • \emptyset : Aktuelle Action-ID (Nummer des Prozess-Schritts des <i>XML Iterator Adapters</i> im aktuellen Prozess) als ID verwenden; Beachten Sie hierbei, dass ein <i>XML Iterator Adapter</i> in einem Subprozess die gleiche Action-ID besitzen könnte (Standard) • Beliebige ganze Zahl außer \emptyset, um den jeweiligen <i>XML Iterator</i>-Baustein innerhalb des Prozess-Kontextes eindeutig zu kennzeichnen; Empfehlung: einen Wert >1000 verwenden, um nicht zufällig eine bereits bestehende Action-ID zu "erwischen"

Statuswerte

<i>-1(error)</i>	Fehler bei der Ausführung des Adapters (Details siehe Server-Protokoll)
------------------	---

<i>0(empty)</i>	Der Adapter gibt keine Daten (mehr) aus. Dieser Status kann als Abbruchbedingung einer Schleife im technischen Prozess ausgewertet werden und gibt an, dass keine weiteren Datensätze verfügbar sind.
<i>1(successful)</i>	Der Adapter gibt ein Ergebnis aus. Dieser Status kann als Schleifeneintrittsbedingung im technischen Prozess ausgewertet werden und gibt an, dass ein weiterer Datensatz verfügbar ist.

6.20.8 XML Validation



XML Validation: Validiert ein Input-XML-Dokument gegen ein XML-Schema oder ein *Relax NG*-Schema.

Ist das geprüfte XML-Dokument gültig, gibt der Adapter dieses unverändert aus; Wenn das geprüfte XML-Dokument ungültig ist, wird ein XML-Dokument mit einer entsprechenden Fehlermeldung ausgegeben.

Eigenschaften (Properties)

Operation	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <code>Validate</code> : XML-Dokument validieren
-----------	---

Parameter

Adapter	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <code>de.softproject.integration.adapter.xmlvalidator.XMLValidator</code> : Hauptklasse (Standard)
SchemaUrl	Pfad zur Schema-Definition (XML-Schema oder Relax NG) Mögliche Werte: Beliebige <code>xstore://</code> -URL (z. B. <code>xstore://Project/Folder/Schema.xsd</code>)
Verbose	Angabe, ob das Ausgabedokument detaillierte Informationen zum Validierungsfehler ausgeben soll Mögliche Werte: <ul style="list-style-type: none"> <code>true</code>: Das Ausgabedokument enthält detaillierte Informationen zum Validierungsfehler (Standard) <code>false</code>: Der Adapter beendet die Validierung nach dem ersten Fehler und gibt keine Informationen über die Fehlerursache aus

Statuswerte

-1 (error)	Während der Ausführung des Adapters trat ein Fehler auf. Nähere Details finden Sie im Server-Log.
------------	---

1 (successful)	Das geprüfte XML-Dokument ist gültig gemäß dem angegebenen Schema; Der Adapter gab das geprüfte Input-XML-Dokument unverändert aus.
2	Das geprüfte XML-Dokument ist nicht gültig gemäß dem angegebenen Schema; Der Adapter gab eine Fehlermeldung aus (siehe "Output").

Ausgabe

Dieser Adapter gibt ein UTF-8-kodiertes XML-Dokument aus. War die Gültigkeitsprüfung erfolgreich, wird das Input-Dokument unverändert ausgegeben. Ist das geprüfte XML-Dokument ungültig, wird ein XML-Dokument mit einer entsprechenden Fehlermeldung ausgegeben, wobei für jeden Fehler ein XML-Element `<Error/>` erzeugt wird, das eine detaillierte Fehlerbeschreibung enthält, zum Beispiel:

```
<?xml version="1.0" encoding="UTF-8"?>
<ValidationErrors>
  <Error>cvc-maxLength-valid: Value 'abc123' with length = '6' is not facet-valid
with respect to maxLength '5' for type 'id'.</Error>
  <Error>cvc-type.3.1.3: The value 'abc123' of element 'Id' is not valid.</Error>
</ValidationErrors>
```

6.20.9 XPath Iterator



XPath Iterator: Selektiert eine Knotengruppe, einen Knoten oder Knoteninhalte über einen XPath-Ausdruck in einem Input-XML-Dokument und gibt je nach Ausdruck eine Knotengruppe als XML-Dokument oder die Werte als Text-Dokument aus. Der Adapter kann dabei alle zutreffenden Knoten durchlaufen und die Ergebnisse nacheinander ausgeben (ForEach-Schleife) oder das Ergebnis des ersten Treffers ausgeben.

- i** Wenn Sie innerhalb eines Prozess-Kontextes mehrere *XPath Iteratoren* definieren, dann vergeben Sie in Parameter *Id* eine eindeutige ID für den jeweiligen Adapter, z.B. *1001*, *1002* etc. So wird für das Selektieren im *forEach*-Modus und für das Zurücksetzen eines Iterators gewährleistet, dass nur der *XPath Iterator* mit der entsprechenden ID angesprochen wird (siehe Operation *Clear*).

Wird in Parameter *Id* der Standardwert *0* belassen, dann wird die Nummer des Prozess-Schritts (Action ID) im aktuellen Prozess als ID verwendet. Diese ID ist möglicherweise nicht eindeutig, wenn sich in einem Subprozess ein *XPath Iterator* mit der gleichen Action-ID befindet.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Select</i>: XPath-Ausdruck auf das Input-XML-Dokument anwenden und Ergebnis-Dokument ausgeben (Parameter Id beachten) • <i>Clear</i>: Iterator eines <i>XPath Iterators</i> zurücksetzen, der dieselbe ID besitzt wie der gewünschte <i>XPath Iterator</i> mit der Operation <i>Select</i> (siehe Parameter Id)
------------------	--

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.xpathselector.XPathSelector</i>: Hauptklasse (Standard)</p>
<i>XPath</i>	<p>XPath-Ausdruck, der auf das Input-XML-Dokument angewandt wird</p> <p>Mögliche Werte: Beliebiger XPath-Ausdruck (z.B. <i>/Element[@att="Value"]/text()</i>)</p>
<i>forEach</i>	<p>Den ersten oder alle zutreffenden Knoten selektieren</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Alle selektierten Knoten nacheinander durchlaufen und jeweils ein Dokument ausgeben (nützlich in Kombination mit einem Condition-Prozessbaustein) (Standard) • <i>nein</i>: Ersten Treffer ausgeben
<i>Id</i>	<p>Eindeutige ID des <i>XPath Iterator</i>-Bausteins innerhalb des Prozess-Kontextes bzw. der Prozess-Instanz (wenn Sie mehrere voneinander unabhängig iterierende <i>XPath Iterator Adapter</i> in demselben Prozess-Kontext verwenden möchten)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>0</i>: Aktuelle Action-ID (Nummer des Prozess-Schritts des <i>XPath Iterators</i> im aktuellen Prozess) als ID verwenden; Beachten Sie hierbei, dass ein <i>XPath Iterator</i> in einem Subprozess die gleiche Action-ID besitzen könnte (Standard) • Beliebige ganze Zahl außer <i>0</i>, um den jeweiligen <i>XPath Iterator</i>-Baustein innerhalb des Prozess-Kontextes eindeutig zu kennzeichnen; Empfehlung: einen Wert <i>>1000</i> verwenden, um nicht zufällig eine bereits bestehende Action-ID zu "erwischen"

Statuswerte

<i>-1(error)</i>	Fehler bei der Ausführung des Adapters (Details siehe Server-Log)
------------------	---

<i>0(empty)</i>	Der Adapter gibt keine Daten (mehr) aus. Dieser Status kann im Iterator-Modus als Abbruchbedingung einer Schleife im technischen Prozess ausgewertet werden und gibt an, dass keine weiteren Datensätze verfügbar sind.
<i>1(successful)</i>	Der Adapter gibt ein Ergebnis aus. Dieser Status kann im Iterator-Modus als Schleifeneintrittsbedingung im technischen Prozess ausgewertet werden und gibt an, dass ein weiterer Datensatz verfügbar ist.

6.21 X4 Tools

6.21.1 Case Management Documents


Case Management Documents: Bildet die Schnittstelle zur „Case Management“-Funktionalität der Produktkomponente X4 BPM. Der Adapter ermöglicht das Verwalten von Dokumenten (Dateien) an Vorgängen. Dokumente können zusätzlich zu Merkmalen des Fachdatenmodells direkt an Vorgängen hinterlegt werden.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • AttachDocument: Fügt dem Vorgang ein Dokument hinzu. • UpdateDocument: Aktualisiert ein vorhandenes Dokument. • UpdateDocumentMetadata: Aktualisiert die Metadaten eines vorhandenen Dokuments. • GetDocument: Gibt das Dokument als Output aus. • GetDocumentMetadata: Gibt die Metadaten eines Dokuments als Output aus. • ListAllDocumentMetadata: Gibt die Metadaten aller Dokumente des Vorgangs als Output aus. • RemoveDocument: Entfernt das Dokument aus dem Vorgang.
-----------	---

Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.x4.server.bmn.casemanagement.adapter.CaseDocumentAdapter</i>: Hauptklasse (Standard)</p>
caseId	<p>ID des zu verarbeitenden Vorgangs</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige (gültige) Case-ID • %CASE_ID%: aktuelle Case-ID (Standard)


title	Titel des Dokuments Mögliche Werte: Beliebige Zeichenkette <div>  Der Titel des Dokuments dient zur Identifizierung und muss innerhalb eines Prozesses eindeutig sein! </div>
fileName	Dateiname des Dokuments Mögliche Werte: Beliebige Zeichenkette, z. B. Anhang/Rechnung.pdf
encoding	Zeichenkodierung des Dokuments Mögliche Werte: Gültige Angabe der Zeichenkodierung, z. B. UTF-8
mimeType	MIME-Type des Dokuments Mögliche Werte: Gültige Angabe des MIME-Types, z. B. application/xml
base64	Gibt an, ob das Dokument als Base64-Zeichenkette übergeben wird. Mögliche Werte: <ul style="list-style-type: none"> • true: Dokument wird als Base64-Zeichenkette übergeben. • false: Dokument wird direkt übergeben (Standard).

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt.
-1	Die Operation konnte nicht ausgeführt werden.

Input

Der Adapter erwartet keinen speziellen Input.

 Wenn bei den Operationen AttachDocument und UpdateDocument der Parameter base64 auf *true* gesetzt ist, muss der Input eine Base64-kodierte Zeichenkette sein.

Output:

Je nach Operation gibt der Adapter Daten in verschiedenen XML-Strukturen aus:

- Bei folgenden Operationen wird der Input bei einer erfolgreichen Operation wieder als Output ausgegeben:
 - AttachDocument

- UpdateDocument
- UpdateDocumentMetadata
- RemoveDocument
- GetDocumentMetadata-Operation

Beispiel-Output für die Operation GetDocumentMetadata

```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>
<Document updatedOn="2019-03-28T13:34:46.233+01:00" updatedBy="x4" createdOn="2019-03-28T13:34:45.690+01:00" createdBy="x4">
  <Title>Rechnung</Title>
  <FileName>Rechnung.xml</FileName>
  <MimeType>application/xml</MimeType>
  <Encoding>UTF-8</Encoding>
</Document>
```

- ListAllDocumentMetadata-Operation

Beispiel-Output für die Operation ListAllDocumentMetadata

```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>
<Documents>
  <Document createdOn="2019-03-28T13:35:04.543+01:00" createdBy="x4">
    <Title>Kommentar</Title>
    <FileName>Kommentar.txt</FileName>
    <MimeType>text/plain</MimeType>
    <Encoding>UTF-8</Encoding>
  </Document>
  <Document createdOn="2019-03-28T13:34:45.690+01:00" createdBy="x4" updatedOn="2019-03-28T13:34:46.233+01:00" updatedBy="x4">
    <Title>Rechnung</Title>
    <FileName>Rechnung.xml</FileName>
    <MimeType>application/xml</MimeType>
    <Encoding>UTF-8</Encoding>
  </Document>
</Documents>
```

- GetDocument-Operation:
Wenn der Parameter base64 auf *true* gesetzt ist, wird das Dokument als Base64-kodierte Zeichenkette ausgegeben.

6.21.2 Case Management

Case Management: Bildet die Schnittstelle zur „Case Management“-Funktionalität der Produktkomponente X4 BPM. Der Adapter ermöglicht beispielsweise das Setzen und Auslesen von Vorgangsdaten wie Fachstatus oder Merkmalen des Fachdatenmodells.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • SetFeature: Schreibt ein Feature • SetFeatures: Schreibt mehrere Features in einem Vorgang • GetFeatures: Gibt alle Features des Vorgangs zurück • SetState: Setzt den Status des Vorgangs • GetStates: Gibt alle möglichen Status des Vorgangs zurück • GetComments: Gibt alle Kommentare des Vorgangs zurück • GetInformation: Gibt alle Informationen des Vorgangs zurück
-----------	---

Parameter

Adapter	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.x4.server.bmn.casemanagement.adapter.CaseManagementAdapter</i>: Hauptklasse (Standard)</p>
caseId	<p>ID des zu verarbeitenden Vorgangs</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige gültige Benutzer-ID • <i>%CASE_ID%</i>: Aktuelle Case-ID (Standard)
caseProcess	<p>Repository-Pfad zu einem Geschäftsprozess, in dem der entsprechende Vorgang definiert ist.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebiger kompletter Repository-Pfad, z. B. <i>Project/Folder/Process.bpm</i> • Beliebiger gültiger Platzhalter, z. B. <i>%START_PROCESS%</i> • <i>%START_PROCESS%</i>: Aufrufender Geschäftsprozess (Standard)
caseUserId	<p>Benutzer-ID des Besitzers eines Vorgangs</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige gültige Benutzer-ID • <i>%USER_ID%</i>: Aktuelle Benutzer-ID (Standard)

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt.
0	Die Operation wurde durchgeführt, es wurde aber kein Ergebnis gefunden oder keine Daten geändert.
-1	Die Operation konnte nicht ausgeführt werden.

Input

Der Adapter erwartet eine vordefinierte XML-Struktur als Input. Die Struktur ist abhängig von der verwendeten Operation.

- SetFeature-Operation:

Erwartete Struktur für die Operation SetFeature

```
<?xml version="1.0" encoding="UTF-8" ?>
<Feature name="Name">Max</Feature>
```

- SetFeatures-Operation:

Erwartete Struktur für die Operation SetFeatures

```
<?xml version="1.0" encoding="UTF-8" ?>
<Features>
  <Feature name="Name">Max</Feature>
  <Feature name="Nachname">Mustermann</Feature>
</Features>
```

- SetState-Operation:

Erwartete Struktur für die Operation SetState

```
<?xml version="1.0" encoding="UTF-8" ?>
<State>NEW</State>
```

Output:

Je nach Operation, gibt der Adapter Daten in verschiedenen XML-Strukturen aus:

- GetFeatures-Operation

Beispiel-Output für die Operation GetFeatures

```
<?xml version="1.0" encoding="UTF-8" ?>
<Features>
  <Feature name="Name">Max/Feature>
  <Feature name="Nachname">Mustermann</Feature>
</Features>
```

- GetStates-Operation

Beispiel-Output für die Operation GetStates

```
<?xml version="1.0" encoding="UTF-8" ?>
<CaseStates>
  <State>NEW</State>
  <State>DONE</State>
</CaseStates>
```

- GetInformation-Operation

Beispiel-Output für die Operation GetInformation

```
<?xml version="1.0" encoding="UTF-8" ?>
<Information>
  <Case creator="x4" created="2018-12-14T16:39:48.100+01:00">
    <Comments />
    <Features>
      <Feature type="String" name="Name">Max</Feature>
      <Feature type="String" name="Nachname">Mustermann</Feature>
    </Features>
    <State>NEW</State>
  </Case>
  <PossibleCaseStates>
    <State>NEW</State>
    <State>DONE</State>
  </PossibleCaseStates>
</Information>
```

6.21.3 Case Management Statistics

Case Management Statistics: Stellt Statistiken für Vorgänge (Geschäftsprozesse) und Human Tasks bereit.

Eigenschaften

Operation	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • GetAllCaseMetadata: Alle Case-Metadaten bereitstellen, die dem Filterkriterium entsprechen • GetAllTaskMetadata: Alle Task-Metadaten bereitstellen, die dem Filterkriterium entsprechen
-----------	--

Parameter

Adapter	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.x4.server.bpm.casemanagement.adapter.CaseManagementStatisticsAdapter</i> Hauptklasse (Standard)
---------	---

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt.
-1	Die Operation ist aufgrund eines technischen Fehlers fehlgeschlagen.

Input

Je nach Operation erwartet der Adapter ein spezielles XML-Dokument als Input.

Operation GetAllCaseMetadata

```
<?xml version="1.0" encoding="UTF-8"?>
<CaseMetadataFilter>
  <!-- Filtert alle Cases nach dem angegebenen Startdatum -->
  <From>2019-10-01T08:00:00.000</From>
  <!-- Filtert alle Cases nach dem angegebenen Enddatum -->
  <To>2019-10-07T08:00:00.000</To>
  <!-- Filtert alle Cases nach dem angegebenen Prozessnamen -->
  <ProcessName>Invoice/Processes/InvoiceApproval.bpm</ProcessName>
  <!-- Filtert alle Cases nach dem angegebenen Case Status -->
  <CaseState>Approved</CaseState>
  <!-- Filtert alle Cases nach dem angegebenen technischen Status
       mögliche Werte: RUNNING, CRASHED, WAITING, FINISHED, FAILED, UNKNOWN -->
  <TechnicalState>FINISHED</TechnicalState>
</CaseMetadataFilter>
```

Operation GetAllTaskMetadata

```
<?xml version="1.0" encoding="UTF-8"?>
<TaskMetadataFilter>
  <!-- Filtert alle Human Tasks nach dem angegebenen Startdatum -->
  <From>2019-10-01T08:00:00.000</From>
  <!-- Filtert alle Human Tasks nach dem angegebenen Enddatum -->
  <To>2019-10-07T08:00:00.000</To>
  <!-- Filtert alle Human Tasks nach dem angegebenen Prozessnamen -->
  <ProcessName>Invoice/Processes/InvoiceApproval.bpm</ProcessName>
  <!-- Filtert alle Human Tasks nach dem angegebenen Task Status
       mögliche Werte: OPEN, CLOSED, IN_PROGRESS, DEADLINE_REACHED -->
  <TaskState>OPEN</TaskState>
  <!-- Filtert alle Human Tasks nach dem angegebenen Case Status -->
  <CaseState>To be approved by clerk</CaseState>
  <!-- Filtert alle Human Tasks nach der angegebenen Frist-->
  <Deadline>2019-10-07T08:00:00.000</Deadline>
  <!-- Filtert alle Human Tasks nach dem angegebenen Empfänger (User oder Gruppe) -->
  <Assignment type="GROUP">clerks</Assignment>
  <!-- Filtert alle Human Tasks nach dem angegebenen Besitzer -->
  <Owner>emil</Owner>
</TaskMetadataFilter>
```

Ausgabe

Je nach Operation liefert der Adapter ein spezielles XML-Dokument als Output.

Operation GetAllCaseMetadata

```
<?xml version="1.0" encoding="UTF-8"?>
<CaseManagementStatistics>
  <CaseMetadata caseId="86757d17-5a8c-47c8-84b1-3fc9f241bf9f">
    <CaseState>Approved</CaseState>
    <ProcessName>Invoice/Processes/InvoiceApproval.bpm</ProcessName>
    <StartDate>2019-10-02T09:27:47.432+02:00</StartDate>
    <EndDate>2019-10-02T09:27:47.695+02:00</EndDate>
    <Duration>263</Duration>
    <TechnicalState>FINISHED</TechnicalState>
  </CaseMetadata>
  <CaseMetadata caseId="a4f12abd-97ad-48cb-a969-40ad84e53ecd">
    <CaseState>Approved</CaseState>
    <ProcessName>Invoice/Processes/InvoiceApproval.bpm</ProcessName>
    <StartDate>2019-10-04T14:27:47.146+02:00</StartDate>
    <EndDate>2019-10-07T14:27:47.842+02:00</EndDate>
    <Duration>696</Duration>
    <TechnicalState>FINISHED</TechnicalState>
  </CaseMetadata>
  ...
</CaseManagementStatistics>
```

Operation GetAllTaskMetadata

```
<?xml version="1.0" encoding="UTF-8"?>
<CaseManagementStatistics>
  <TaskMetadata taskId="fa24c1d0-e9f0-4a72-bca0-760c1cb1d653">
    <ProcessName>Invoice/Processes/InvoiceApproval.bpm</ProcessName>
    <TaskState>OPEN</TaskState>
    <CaseState>To be approved by clerk</CaseState>
    <Deadline>2019-10-05T16:00:00.000+02:00</Deadline>
    <Assignment type="GROUP">clerks</Assignment>
    <Owner>emil</Owner>
    <Start>2019-10-01T08:14:41.395+02:00</Start>
    <End>2019-10-02T10:48:12.753+02:00</End>
  </TaskMetadata>
  <TaskMetadata taskId="9b330bc3-310a-4638-ac1b-07f115d564d7">
    <ProcessName>Invoice/Processes/InvoiceApproval.bpm</ProcessName>
    <TaskState>OPEN</TaskState>
    <CaseState>To be approved by clerk</CaseState>
    <Deadline>2019-10-06T16:00:00.000+02:00</Deadline>
    <Assignment type="USER">clerks</Assignment>
    <Owner>emil</Owner>
    <Start>2019-10-01T13:27:53.593+02:00</Start>
    <End>2019-10-04T11:56:09.852+02:00</End>
  </TaskMetadata>
  ...
</CaseManagementStatistics>
```


6.21.4 X4 Variable Collector



X4 Variable Collector: Sammelt Ergebnisse aus Prozessschleifen und speichert diese in einer Variablen.

Eigenschaften (Properties)

Operation	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • Collect: Daten Base64-encodieren, siehe Operation Collect • Get: Inhalt der Prozess-Variablen auslesen, siehe Operation Get • GetAndRemove: Inhalt der Prozess-Variablen auslesen und die Prozess-Variable anschließend entfernen, siehe Operation GetAndRemove
-----------	---

Parameter

Adapter	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.merge.Collector</i> : Hauptklasse (Standard)
name	Name der Prozess-Variablen
mode	Merge-Modus; Bestimmt wie der Input in die Prozess-Variable zusammengeführt wird. Mögliche Werte: <ul style="list-style-type: none"> • BINARY: Der Input wird als Byte-Strom zusammengeführt • TEXT: Der Input wird als String zusammengeführt • XML: Der Input wird als XML-Dokument zusammengeführt

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt
-1	Bei der Ausführung der Operation kam es zu technischen Problemen
0	Es wurde keine Prozess-Variable gefunden (bei Get und GetAndRemove)

6.21.4.1 Operation Collect

6.21.4.1.1 Modus: BINARY

Folgendes gilt für den Modus BINARY:

- Adapter-Input wird als Byte-Array interpretiert und an den als Byte-Array interpretierten Inhalt der Prozess-Variablen angehängt.
- NULL wird als Byte-Array der Länge 0 betrachtet.
- Der Content-Type des Ergebnisses ist `application/octet-stream`.
- Das Ergebnis wird in die Prozess-Variable zurückgeschrieben und als Adapter-Ergebnis zurückgeliefert.

6.21.4.1.2 Modus: TEXT

Folgendes gilt für den Modus TEXT:

- Der Adapter-Input wird als String interpretiert (das Encoding wird hierzu ausgewertet) und an den als String interpretierten Inhalt der Prozess-Variablen (hier wird ebenfalls das Encoding verwendet) angehängt.
- NULL wird als String der Länge 0 betrachtet.
- Der Content-Type des Ergebnisses ist `text/plain` mit dem Encoding UTF-8.
- Das Ergebnis wird in die Prozess-Variable zurückgeschrieben und als Adapter-Ergebnis zurückgeliefert.

6.21.4.1.3 Modus: XML

Folgendes gilt für den Modus XML:

- Der Adapter-Input wird als XML interpretiert und an den als XML interpretierten Inhalt der Prozess-Variablen eingefügt.
- Der Content-Type des Ergebnisses ist `text/xml` und das Encoding UTF-8.
- Das Ergebnis wird in die Prozess-Variable zurückgeschrieben und als Adapter-Ergebnis zurückgeliefert.
- Falls der Adapter-Input oder der Inhalt der Prozess-Variablen nicht als XML interpretiert werden kann (NULL ist erlaubt), wird der Adapter mit Status -1 und NULL als Ausgabe beendet. Der Inhalt der Prozess-Variablen wurde dann nicht verändert.

Angehängt wird wie folgt:

- Es wird ein neues Dokument mit dem Wurzelement Merge erstellt.
- Ist der Inhalt der Prozess-Variablen nicht NULL, wird ihr Inhalt wie folgt zusammengeführt. Ist das Wurzelement der Prozess-Variablen Merge, werden alle Kind-Elemente als Kind-Element des Ausgabe-Merge-Elements übernommen. Andernfalls wird das Wurzelement als Kind des Ausgabe-Merge-Elements übernommen.
- Anschließend wird dieselbe Behandlung auf den Adapter-Input angewendet. Nach einem erfolgreichen Collect im XML-Modus befindet sich damit in der Prozess-Variablen auf jeden Fall ein XML-Dokument mit einem Wurzelement Merge. Aus Sicht der Collect-Operation im Modus XML, sind NULL (Nichts) und das XML-Dokument `<Merge/>` gleichwertig.

6.21.4.2 Operation Get

Die Operation Get liefert den Inhalt der Prozess-Variablen zurück. Ist die Variable gefüllt, wird der Status 1 zurückgegeben, ist die Variable leer, der Status 0. Die Operation Get verhält sich damit wie die Operation Test des Variablen-Bausteins.

i Für den Fall, dass die Variable leer war, wird das neutrale Element in die Variable gesetzt und als Adapterausgabe verwendet.

Modus	Entsprechendes neutrales Element
BINARY	Byte-Array der Länge 0
TEXT	String der Länge 0
XML	Ein XML-Dokument, das ausschließlich aus dem Wurzelknoten mit dem Namen Merge besteht

6.21.4.3 Operation GetAndRemove

Die Operation GetAndRemove liefert den Inhalt der Prozess-Variablen zurück. Ist die Variable gefüllt, wird der Status 1 zurückgegeben, ist die Variable leer, der Status 0. Die Operation GetAndRemove kombiniert damit die Operationen Remove und Test des [Variablen-Bausteins](#).

i Für den Fall, dass die Variable leer war, wird das neutrale Element in die Variable gesetzt und als Adapterausgabe verwendet.

Modus	Entsprechendes neutrales Element
BINARY	Byte-Array der Länge 0
TEXT	String der Länge 0
XML	Ein XML-Dokument, das ausschließlich aus dem Wurzelknoten mit dem Namen Merge besteht

Nach dieser Operation ist die Variable leer.

6.21.5 Commandline



CommandLine: Ruft eine externe Anwendung oder einen externen Befehl über eine Kommandozeile des Betriebssystems mit beliebig vielen Argumenten auf. Der Adapter erwartet eine adapterspezifische Input-XML-Struktur, die Base64-kodierte Nutzdaten enthalten kann und gibt ein XML-Dokument mit den Base64-kodierten Ergebnissen des Aufrufs von *StdOut* und/oder *StdErr* sowie dem zurückgegebenen *Exit Code* zurück.

Ehemalige Bezeichnung: *OS Process Execution Adapter*

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt. Mögliche Werte: <i>Execute</i> : Anwendung/Befehl über eine Kommandozeile aufrufen
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.osprocess.OsProcessAdapter</i> : Hauptklasse (Standard)
----------------	---

<i>Working Directory</i>	<p>Arbeitsverzeichnis für den ausgeführten Anwendungsaufwurf; wird ggf. durch das im Input-XML gesetzte Arbeitsverzeichnis überschrieben; falls beides leer ist, wird das <i>X4 Server</i>-Arbeitsverzeichnis verwendet</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Beliebige gültige Pfadangabe (z.B. <i>C:\X4\temp\</i>) • <i>(leer)</i>: Kein Standard-Arbeitsverzeichnis (über Input-XML manuell setzen oder das <i>X4 Server</i>-Arbeitsverzeichnis wird verwendet) (Standard)
<i>CollectStdOut</i>	<p><i>Standard Output</i>-Strom des aufgerufenen Programms erfassen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: <i>Standard Output</i>-Strom des aufgerufenen Programms im Ergebnis-XML-Dokument in Element <code><StdOut></code> Base64-kodiert speichern (Standard) • <i>nein</i>: <i>Standard Output</i> nicht berücksichtigen (Element <code><StdOut></code> im Ergebnis-XML-Dokument bleibt leer)
<i>CollectStdErr</i>	<p><i>Standard Error</i>-Strom des aufgerufenen Programms erfassen</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: <i>Standard Error</i>-Strom des aufgerufenen Programms im Ergebnis-XML-Dokument in Element <code><StdErr></code> Base64-kodiert speichern (Standard) • <i>nein</i>: <i>Standard Error</i> nicht berücksichtigen (Element <code><StdErr></code> im Ergebnis-XML-Dokument bleibt leer)
<i>MergeStdErrIntoStdOut</i>	<p><i>Standard Error</i>-Strom zusammen mit dem <i>Standard Output</i>-Strom in Element <code><StdOut></code> Base64-kodiert ausgeben</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: <i>Standard Error</i> und <i>Standard Output</i> zusammen ausgeben • <i>nein</i>: Datenströme separat ausgeben (Standard)

Input

Der Adapter erwartet ein XML-Dokument, das nach folgendem Muster aufgebaut ist:


```

<Command>
  <WorkingDirectory><!--Arbeitsverzeichnis (optional)--></WorkingDirectory>
  <Application><!--Name der Kommandozeile oder kompletter Pfad der Anwendung--></
Application>
  <!-- Beliebig viele Argumente (Reihenfolge beachten!) -->
  <Argument><!--Argument/Parameter--></Argument>
  <Environment clear="nur mit folgenden Umgebungsvariablen arbeiten (true/false)">
    <!-- beliebig viele Umgebungsvariablen -->
    <Entry name="Umgebungsvariablenname"><!--Wert--></Entry>
  </Environment>
  <Input><!--Base64-kodierte Daten zur Verarbeitung durch die externe Anwendung--></
Input>
  <Copy><!--Beliebige Zeichenkette/XML-Struktur zum unveränderten Durchschleifen--></
Copy>
</Command>

```

Erläuterung:

- **<WorkingDirectory>** (optional): Arbeitsverzeichnis in dem die aufgerufene Anwendung operiert; überschreibt den Wert in Parameter `WorkingDirectory`; entspricht *nicht* dem Pfad der Anwendung (siehe unten)!
- **<Application>**: Name der Kommandozeile oder Anwendungsname mit oder ohne komplettem Pfad (falls der Anwendungspfad bereits in den Umgebungsvariablen verfügbar ist)
Alternativ unter Windows `cmd.exe` verwenden und die Anwendung als erstes Argument angeben
- **<Environment>**: Definiert Umgebungsvariablen; Über das Attribut `clear="true"` können Sie steuern, ob der Befehl nur mit den Umgebungsvariablen, die innerhalb von Element `<Environment>` definiert wurden, ausgeführt wird. Mit `clear="false"` werden die innerhalb von Element `<Environment>` definierten Umgebungsvariablen zu den bestehenden hinzugefügt.
- **<Input>** (optional) Enthält Base64-kodierte Input-Daten
- **<Copy>** (optional): Enthält eine XML-Struktur/beliebige Zeichenkette zum unveränderten Durchschleifen

 Im Gegensatz zu den Adapter-Parametern können Sie im Input-XML keine Platzhalter wie z.B. `%USER_COLLECTION%` verwenden!

Beispiel

In der folgenden Input-XML-Struktur wird die Standardkommandozeile von Windows, `cmd.exe`, im Arbeitsverzeichnis `C:\` aufgerufen und es werden mit dem Befehlsargument `set` die Umgebungsvariablen `name` und `age` gesetzt. Über das Argument `/c` wird dabei Befehl in der Zeichenfolge ausgeführt und dann beendet. Über `clear="true"` wird der Befehl nur mit den hier definierten Umgebungsvariablen ausgeführt.

```
<Command>
  <WorkingDirectory>C:\</WorkingDirectory>
  <Application>cmd.exe</Application>
  <Argument>/c</Argument>
  <Argument>set</Argument>
  <Environment clear="true">
    <Entry name="name">Theo Tester</Entry>
    <Entry name="age">35</Entry>
  </Environment>
  <Input></Input>
  <Copy></Copy>
</Command>
```

Ausgabe

Der Funktions-Adapter gibt ein XML-Dokument aus, das nach folgendem Muster aufgebaut ist.

```
<CommandResult>
  <ExitCode>ExitCode der Anwendung</ExitCode>
  <StdOut>Base64-kodierte Ausgabedaten der Anwendung</StdOut>
  <StdErr>Base64-kodierte Fehlerausgabe der Anwendung</StdErr>
  <Copy>Unverändert durchgeschleifte Zeichenkette/XML-Struktur</Copy>
</CommandResult>
```

Statuswerte

1	Aufruf wurde ausgeführt; der Exitcode des Programms ist 0.
0	Aufruf wurde ausgeführt; der Exitcode des Programms ist ungleich 0.
-1	Aufruf konnte nicht ausgeführt werden (z. B. weil Programm nicht gefunden wurde).

6.21.6 Custom Placeholder Initializer

Custom Placeholder Initializer: Verwaltet die Platzhalter im aktuellen Prozess. Das Laden und Entladen von Platzhaltern aus dem Prozesskontext kann über Adapter-Parameter oder ein XML-Dokument gesteuert werden.

Eigenschaften (Properties)

Operation	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • Load: Lädt die im Parameter groups definierten Platzhalter in den Prozesskontext (<i>Input-XML = Output-XML</i>) • Unload: Entfernt die im Parameter groups definierten Platzhalter aus dem Prozesskontext (<i>Input-XML = Output-XML</i>) • LoadFromXML: Lädt die Parameter aus dem XML-Dokument in den Prozesskontext • UnloadFromXML: Entfernt die Parameter aus dem XML-Dokument aus dem Prozesskontext • Export: Gibt alle aktuell gesetzten Platzhalter im Prozesskontext aus (benötigt kein Input-Dokument)
-----------	--

Parameter

Adapter	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.placeholder.ProcessPlaceholderAdapter</i> : Hauptklasse (Standard)
groups	Liste der zu ladenden Platzhalter-Gruppen aus dem Platzhalter-Speicher. Werte können mit Komma oder Semikolon getrennt werden. Ist der Parameter leer, wird die DEFAULT-Platzhalter-Gruppe geladen. <i>Beispiel: DEFAULT, Prod, Dev, Test</i>
maxAge	Gibt in Sekunden an, wie alt die Platzhalter maximal sein dürfen, wenn sie aus dem Cache gelesen werden. Wird der Wert überschritten, werden die Platzhalter neu eingelesen. Bei Werten kleiner gleich Null (≤ 0) wird immer direkt aus dem Platzhalter-Speicher gelesen. Mögliche Werte: ganze Zahlen

Statuswerte

1(successful)	Die Operation war erfolgreich
0(empty)	Es wurden keine Platzhalter in den Prozesskontext geladen
-1(failed)	Die Operation schlug aufgrund eines technischen Fehlers fehl

Input

Die Operationen LoadFromXML und UnLoadFromXML erwarten folgendes XML-Format:

LoadFromXML

```
<?xml version="1.0" encoding="UTF-8"?>
<Placeholder>
  <Group name="DEV">
    <Key name="message">Hallo Entwickler!</Key>
  </Group>
  <Group name="DEFAULT">
    <Key name="message">Hallo Welt!</Key>
  </Group>
</Placeholder>
```

Es werden die zwei Platzhalter-Gruppen *DEFAULT* und *DEV* mit dem jeweiligen Platzhalter *message* im Platzhalterkontext abgelegt.

UnloadFromXML

```
<?xml version="1.0" encoding="UTF-8"?>
<Placeholder>
  <Group name="DEV" />
  <Group name="DEFAULT">
    <Key name="message" />
  </Group>
</Placeholder>
```

Alle Platzhalter der Gruppe *DEV* und der Platzhalter *message* aus der *DEFAULT* Platzhalter-Gruppe werden aus dem Prozesskontext entfernt.

Ausgabe

Die Operation *Get* gibt alle aktuellen Platzhalter im Prozesskontext aus. Die Operationen *LoadFromXML* und *UnLoadFromXML* geben die verarbeiteten Elemente aus.

```
<?xml version="1.0" encoding="UTF-8"?>
<Placeholder>
  <Group name="DEV">
    <Key name="message">Hallo Entwickler!</Key>
  </Group>
  <Group name="DEFAULT">
    <Key name="message">Hallo Welt!</Key>
  </Group>
</Placeholder>
```

6.21.7 Custom Placeholder Storage Manager

Custom Placeholder Storage Manager: Verwaltet die Platzhalter im definierten Platzhalterspeicher.

Eigenschaften (Properties)

Operation	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • AddKeys: Fügt alle Platzhalter in den jeweiligen Platzhaltergruppen hinzu • Export: Exportiert alle Platzhalter aus dem Platzhalterspeicher (benötigt kein Input-Dokument) • Get: Gibt alle Platzhalter der angefragten Platzhaltergruppen unter Berücksichtigung des Caches aus • Import: Alle Platzhalter in den jeweiligen Platzhaltergruppen werden hinzugefügt oder ersetzt • Refresh: Lädt die Platzhalter aus der Datenbank neu • RemoveKeys: Entfernt alle Platzhalter in den jeweiligen Platzhaltergruppen • RemoveGroups: Entfernt alle Platzhaltergruppen • ReplaceGroups: Ersetzt alle Platzhalter in den jeweiligen Platzhaltergruppen
-----------	--

Parameter

Adapter	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.placeholder.ProcessPlaceholderAdapter</i> : Hauptklasse (Standard)
---------	--

Statuswerte

1 (successful)	Die Operation war erfolgreich
-1 (failed)	Die Operation schlug aufgrund eines technischen Fehlers fehl

Input

Für die Operationen *Get*, *AddKeys*, *RemoveKeys*, *RemoveGroups*, *ReplaceGroups* und *Import* werden folgende Input-XML-Dokumente erwartet:

Operation: Get

Folgende Input-XML-Struktur lädt die Gruppen DEV und DEFAULT aus dem Platzhalter-Speicher:

```
<?xml version="1.0" encoding="UTF-8"?>
<Placeholder>
  <Group name="DEV"/>
  <Group name="DEFAULT"/>
</Placeholder>
```

Operation: AddKeys & ReplaceGroups & Import

Folgendes XML-Dokument beschreibt Platzhalter zum Hinzufügen, Ersetzen und Importieren:

```
<?xml version="1.0" encoding="UTF-8"?>
<Placeholder><!-- Bei Operation Import optionales Attribut (default=false):
deleteAndImport=true|false -->
  <Group name="DEV">
    <Key name="message" type="String">Hallo X4 Entwickler!</Key><!-- gültige
Typen: String|Boolean|Integer|Decimal|Date -->
  </Group>
  <Group name="DEFAULT">
    <Key name="message" type="String">Hallo Welt!</Key>
    <Key name="endOfTheWorld" type="Date">2012-12-21</Key>
    <Key name="sayHello" type="Boolean">true</Key>
  </Group>
</Placeholder>
```

❗ Eine leere Platzhaltergruppe beim Aufruf der Operation Replace führt dazu, dass die gesamte Platzhaltergruppe geleert wird. Der Aufruf entspricht somit der Operation Remove. Bei der Operation Import ist es optional möglich, die entsprechenden Platzhaltergruppen vor dem Import zu Löschen. Hierfür muss im Wurzel-Knoten das Attribut deleteAndImport auf *true* gesetzt werden.

Operation: RemoveKeys

Folgendes XML-Dokument entfernt einzelne Platzhalter einer Gruppe:

```
<?xml version="1.0" encoding="UTF-8"?>
<Placeholder>
  <Group name="DEFAULT">
    <Key name="endOfTheWorld"/>
  </Group>
</Placeholder>
```

Operation: RemoveGroups

Folgendes XML-Dokument entfernt eine ganze Platzhaltergruppe:

```
<?xml version="1.0" encoding="UTF-8"?>
<Placeholder>
  <Group name="DEFAULT" />
</Placeholder>
```

Ausgabe

Die Operation Refresh gibt den aktuellen Status aller Platzhalter in einem XML-Dokument zurück.

Die Operationen geben die angefragten oder durch die Anfrage geänderten Platzhaltergruppen zurück.

```
<?xml version="1.0" encoding="UTF-8"?>
<Placeholder>
  <Group name="DEV">
    <Key name="message" type="String">Hallo X4-Entwickler!</Key>
  </Group>
  <Group name="DEFAULT">
    <Key name="message" type="String">Hallo Welt!</Key>
    <Key name="endOfTheWorld" type="Date">2012-12-21</Key>
    <Key name="sayHello" type="Boolean">true</Key>
  </Group>
</Placeholder>
```

6.21.8 Document Comparison



Document Comparison: Vergleicht eine Repository-Datei mit einer Input-Datei und gibt als Prozess-Ergebnis einen Statuswert aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt
	Mögliche Werte: <i>Compare</i> : Dateien miteinander vergleichen

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.comparison.X4ComparisonAdapter</i> : Hauptklasse (Standard)
<i>comparisonMethod</i>	Vergleichsmethode Mögliche Werte: <ul style="list-style-type: none"> • BYTE: Dateien byte-weise miteinander vergleichen (Standard) • ZIP: ZIP-Dateien anhand ihrer Inhalte miteinander vergleichen • XML: XML-Dateien anhand ihrer Inhalte miteinander vergleichen • TEXT: Text-Dateien anhand ihrer Inhalte miteinander vergleichen • EMPTY: Input-Dateien mit leeren Byte-Arrays vergleichen
<i>expectedResult</i>	Pfad zum Dokument das mit dem Adapter-Input verglichen werden soll. Bezieht sich auf die Vergleichsmethoden BYTE, ZIP, XML und TEXT Mögliche Werte: Repository-Pfad, z. B. <i>X4Experience/BPM/1-Examples/InvoiceReceipt/BPM/Data/Invoice.xml</i> (Pflichtfeld)

<i>ignoreWhitespace</i>	<p>Gibt an, ob Leerzeichen beim Vergleichen ignoriert werden sollen; Bezieht sich auf die Vergleichsmethoden XML und TEXT</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ja: Leerzeichen ignorieren • nein: Leerzeichen berücksichtigen (Standard)
<i>normalizeWhitespace</i>	<p>Gibt an, ob Leerzeichen am Anfang und Ende einer Zeichenkette entfernt werden sollen; Bezieht sich auf die Vergleichsmethoden XML und TEXT</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ja: Leerzeichen wird am Anfang und Ende einer Zeichenkette entfernt (Standard) • nein: Leerzeichen am Anfang und Ende einer Zeichenkette bleibt erhalten
<i>ignoreComments</i>	<p>Gibt an, ob Kommentare innerhalb des XML-Dokuments beim Vergleichen ignoriert werden sollen; Bezieht sich auf die Vergleichsmethode XML</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ja: Kommentare ignorieren • nein: Kommentare berücksichtigen (Standard)
<i>compareSimilar</i>	<p>Gibt an, ob die Reihenfolge der Elemente und Attribute (Tags) innerhalb des XML-Dokuments beim Vergleichen ignoriert werden soll, z. B. wenn die Inhalte innerhalb des XML-Dokuments zwar gleich, aber ihre Reihenfolge unterschiedlich ist; Bezieht sich auf die Vergleichsmethode XML</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ja: Reihenfolge ignorieren • nein: Reihenfolge berücksichtigen (Standard)
<i>ignoreCDATAAndTextDiff</i>	<p>Gibt an, ob der Inhalt innerhalb des Elementes CDATA als Klartext interpretiert werden soll; Bezieht sich auf die Vergleichsmethode XML</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • ja: Inhalt von CDATA als Klartext interpretieren (Standard) • nein: Inhalt von CDATA nicht als Klartext interpretieren

Input

Als Input erwartet der Adapter eine beliebige Datei, die mit einer gegebenen Repository-Datei verglichen wird.

Ausgabe

Als Prozess-Ergebnis können verschiedene Statuswerte ausgegeben werden:

1	Die verglichenen Dokumente sind gleich
0	Die verglichenen Dokumente sind nicht gleich

-1	Die Dokumente konnten nicht miteinander verglichen werden, da der Input entweder leer, nicht wohlgeformt oder falsch war.
----	---

6.21.9 Document Properties Changer



Document Properties Changer: Liest und verändert Dokumenteigenschaften eines im X4 ESB für den Prozessdaten-Austausch verwendeten sogenannten X4Document. Dabei werden lediglich die intern verwendeten Dokumenteigenschaften verändert, was Einfluss auf die Ausführung von einigen Adaptoren (nicht aber von XSL-Mappings) haben kann. Die eigentlichen Daten des Dokuments bleiben unverändert.

Ehemalige Bezeichnung: *Document Properties Adapter*

- i** Um die Zeichenkodierung der Dokumentdaten zu ändern, den [Charset Changer](#) verwenden. Um die Zeichen der Dokumentdaten in eine andere Zeichenkodierung zu konvertieren, den [Encoding Converter](#) verwenden. Beachten Sie zudem: Änderungen des Zeichenkodierung mit dem *Document Properties Changer* haben keinen Effekt auf XSL-Transformationen. Wenn sich keine XML-Deklaration mit Angabe der Zeichenkodierung im Dokument befindet, dann wird *UTF-8* gemäß XML-Spezifikation als Standard-Zeichenkodierung verwendet.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Change</i> : MIME-Typ und/oder Zeichenkodierung des X4Document ändern
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.documentproperties.DocumentPropertiesAdapter</i> : Hauptklasse (Standard)
<i>mimeType</i>	Neuer MIME-Typ des X4Document Mögliche Werte: Beliebiger MIME-Typ (z.B. <i>text/xml</i>)
<i>encoding</i>	Neue Zeichenkodierung des X4Document Mögliche Werte: Beliebige Zeichenkodierung (z.B. <i>ISO-8859-1</i>)

6.21.10 Empty Output Adapter



Empty Output Adapter : Leert den Datenstrom im technischen Prozess und dient so als Datensenkke, um einen wohldefinierten Zustand innerhalb des technischen Prozesses zu schaffen.

Insbesondere im Produktionsmodus des *X4 Servers* ist dieser Adapter hilfreich, um eine leere Datei zu erzeugen, ohne dass Zugriff auf ein Dateisystem erforderlich ist. Dieser verarbeitet beliebige Input-Daten und gibt ein 0 Byte langes Binärdokument oder nichts (Null) aus.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>Empty</i>: Dokument mit 0 Byte Länge ausgeben • <i>Null</i>: Nichts (Null) ausgeben
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.sink.Sink</i> : Hauptklasse (Standard)
----------------	---

6.21.11 Global Context Manipulation



Global Context Manipulation: Bietet die Möglichkeit, ähnlich wie bei Prozess-Variablen beliebige Dokumente aus einem technischen Prozess in einer globalen Hashtabelle zwischenspeichern, sodass auch von anderen Prozessen auf diesen Kontext zugegriffen werden kann, siehe *Handbuch X4 Designer – Variablen-Baustein*.

Die im globalen Kontext abgelegten Dokumente werden unter einem Schlüssel gespeichert und sind über die gesamte Laufzeit des *X4 Servers* verfügbar. Der Funktions-Adapter kann Einträge in der Hash-Tabelle erstellen, überschreiben und löschen sowie alle oder ausgewählte Schlüssel der Hashtabellen-Einträge auflisten.

- i** Da ein Kontext *X4 Server*-weit gilt und alle technischen Prozesse auf dieselbe Hashtabelle zugreifen, ist es möglich, dass andere Threads parallel ein Dokument unter demselben Schlüssel einstellen. Zudem können geringe Zeitdifferenzen zwischen den Operationen Put und PutGet uneindeutige (z. B. leere) Ergebnisse liefern. Innerhalb von XSL-Mappings können Sie über das `globalcontext:-URL`-Protokoll zur Laufzeit auf bereits definierte Kontextschlüssel zugreifen, siehe *X4 Developer Guide – globalcontext:-URL-Protokoll*.

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Get</i>: Eintrag, dessen Schlüssel in Parameter <i>Key</i> festgelegt wurde, aus dem globalen Kontext auslesen und das zuletzt vollständig gespeicherte Dokument ausgeben • <i>Put</i>: Input-Dokument mit dem in Parameter <i>Key</i> festgelegten Schlüssel im globalen Kontext speichern und liefert den gesetzten Eintrag zurück • <i>PutGet</i>: Input-Dokument mit einem Schlüssel im globalen Kontext speichern und das zuletzt vollständig gespeicherte Dokument ausgeben (dass das aktuellste Dokument ausgegeben wird, kann nicht garantiert werden!) • <i>Remove</i>: Eintrag, dessen Schlüssel in Parameter <i>Key</i> festgelegt wurde, aus dem Globalen Kontext löschen und gibt den gelöschten Eintrag zurück • <i>List</i>: Alle Schlüssel des globalen Kontexts in einem XML-Dokument ausgeben • <i>RemoveAllMatching</i>: Alle Einträge des globalen Kontexts löschen, deren Schlüssel dem Regex-Filterausdruck in Parameter <i>Filter</i> entsprechen • <i>KeepAllMatching</i>: Nur Einträge des globalen Kontexts behalten, deren Schlüssel dem Regex-Filterausdruck in Parameter <i>Filter</i> entsprechen (alle anderen Einträge löschen) • <i>ListAllMatchingKeys</i>: Alle Schlüssel des globalen Kontexts, die dem Regex-Filterausdruck in Parameter <i>Filter</i> entsprechen, in einem XML-Dokument ausgeben • <i>ListAllNotMatchingKeys</i>: Nur Schlüssel des globalen Kontexts, die nicht dem Regex-Filterausdruck in Parameter <i>Filter</i> entsprechen, in einem XML-Dokument ausgeben
------------------	---

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.globalcontext.GlobalContextAdapter</i>: Hauptklasse (Standard)</p>
<i>Key</i>	<p>Schlüssel, unter dem ein Eintrag im globalen Kontext gespeichert ist</p> <p>Mögliche Werte: Beliebige Zeichenkette</p>
<i>Filter</i>	<p>Regulärer Ausdruck (Regex), um nach Schlüssel im globalen Kontext zu filtern, gemäß <i>java.util.Pattern</i>, siehe http://java.sun.com/j2se/1.5.0/docs/api/java/util/regex/Pattern.html</p> <p>Mögliche Werte: Beliebiger Regulärer Ausdruck (z.B. <i>.*Test.*</i>)</p>

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt.
0	Die Operation wurde erfolgreich ausgeführt, das Ergebnis ist jedoch Null oder eine leere Liste.
-1	Der Adapter konnte nicht ausgeführt werden.

6.21.12 Log4J Logging

Log4J Logging: Gibt Daten aus dem Prozess über *Log4J* in das Server-Log aus. Als Input-Daten können alle Daten dienen, die sich in eine Zeichenkette umwandeln lassen. Im *AdvancedMode* können in einem XML-Dokument mit adapterspezifischer Struktur Daten auf verschiedenen Log-Stufen direkt ausgegeben werden.


Dieser Adapter gibt seine Input-Daten unverändert an den nächsten Prozess-Schritt weiter.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Execute</i> : Adapter ausführen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.log4j.Log4jAdapter</i> : Hauptklasse (Standard)
<i>logger</i>	Verwendeter Logger, an den die Log-Nachricht gesendet werden soll Mögliche Werte: <ul style="list-style-type: none"> • Beliebiger Logger-Name • <i>X4Logger</i>: Ausgabe als <i>X4Logger</i> in der Server-Konsole über Log4j

<i>level</i>	<p>Stufe der ausgegebenen Log-Nachricht. In der Log4j-Konfiguration ist eine Schwelle (Parameter <code>Threshold</code>) definiert, ab welcher Stufe Log-Nachrichten ausgegeben werden; meist ist dies <i>INFO</i>.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>DEBUG</i>: Debug-Informationen ausgeben (falls dies die Logging-Konfiguration zulässt) • <i>INFO</i>: Allgemeine Informationen oder verarbeitete Prozessdaten ausgeben • <i>WARN</i>: Warnung für eine unerwartete Situation ausgeben • <i>ERROR</i>: Fehlermeldung (abgefangene Exception) ausgeben • <i>FATAL</i>: Kritischen Systemfehler ausgeben <div style="border: 1px solid #add8e6; padding: 10px; margin-top: 10px;"> <p> Log-Nachrichten mit dem Level <i>FATAL</i> werden technisch bedingt mit dem Level <i>ERROR</i> ausgegeben. Die Ausgabe erhält <i>FATAL</i>: als Präfix.</p> </div>
<i>advancedMode</i>	<p>Erweiterten Modus aktivieren, in dem ein Input-XML-Dokument mit einer adapterspezifischen Struktur gesondert verarbeitet wird. Ein solches XML-Dokument kann Log-Nachrichten enthalten, die auf einer angegebenen Log-Stufe ausgegeben werden (siehe Beispiel unten)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ja</i>: Input-XML-Dokumente parsen • <i>nein</i>: Input-Daten nicht parsen (Standard)
<i>modifyMessage</i>	<p>Log-Nachricht verändern/erweitern</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Nothing</i>: Log-Nachricht mit unverändertem Layout ausgeben • <i>AddPid</i>: In der Log-Nachricht zusätzlich die aktuelle Prozess-Instanz-ID ausgeben (Standard)

Beispiel

Wenn der Parameter `advancedMode` aktiviert ist: Dieses adapterspezifische Input-XML-Dokument enthält Angaben zur Log-Stufe und die auszugebenden Log-Nachrichten. Die Nachrichten werden direkt ins Server-Log ausgegeben:


```

<logs>
  <log logger="X4Logger">
    <message level="DEBUG">This is a debug message.</message>
    <message level="INFO">This is a info message. </message>
    <message level="WARN">This is a warn message. </message>
    <message level="ERROR">This is an error message. </message>
    <message level="FATAL">This is a fatal message. </message>
  </log>
</logs>

```

6.21.13 Parallel Process Starter

Parallel Process Starter: Führt einen *X4*-Prozess in einem beliebigen Benutzerordner in *<X4>* / *X4DB* beliebig oft parallel aus und ermöglicht es, die Ergebnisse anschließend wieder zu synchronisieren. Der Adapter erwartet ein Input-XML-Dokument mit zu verarbeitenden Daten, jeweils abgelegt innerhalb von beliebig benannten Kindelementen unterhalb des Wurzel-Elements. Der Adapter gibt je nach Konfiguration ein XML-Dokument oder eine Base64-kodierte Zeichenkette aus.

 Mit dem *Parallel Process Starter* können Sie über *dynamische Parameter* den Prozessnamen (und -pfad) zur Laufzeit setzen. Außerdem besitzt jeder über den *Parallel Process Starter* gestartete *X4*-Prozess eine eigene Prozessinstanz-ID, wird in einem separaten Thread ausgeführt und er kann daher nicht auf Variablen des aufrufenden Prozesses zugreifen.

Allgemeine Eigenschaften und Parameter

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt
	Mögliche Werte: <i>Execute</i> : Prozess(e) ausführen

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.parallelprocessstarter.Adapter</i> : Hauptklasse (Standard)
<i>X4Process URL</i>	Pfad zum auszuführenden <i>X4</i> -Prozess Mögliche Werte: Beliebige gültige <i>xstore</i> - oder <i>x4db</i> :-URL zum <i>X4</i> -Prozess im <i>X4 Repository</i> (z. B. <i>xstore://Project/Process.wrf</i> bzw. <i>x4db:/1/Project/Folder/Process.wrf</i>)

<i>MaxThread</i>	Anzahl der Threads, die genutzt werden sollen Mögliche Werte: <ul style="list-style-type: none"> • Beliebige positive ganze Zahl (z.B. 3) • 0 oder negative ganze Zahl: parallele Ausführung nicht beschränken (z.B. -1)
<i>InputIsBase64</i>	Gibt das Datenformat für den Adapter-Input an Mögliche Werte: <ul style="list-style-type: none"> • True: Der Inhalt jedes Elements ist eine Base64-kodierte Zeichenkette, die dekodiert als Input an den Kind-Prozess weitergegeben wird. • False: Der XML-Elementinhalt wird direkt an den Kind-Prozess weitergegeben.
<i>expandOutput</i>	Steuert das Ausgabeformat des Adapters Mögliche Werte: <ul style="list-style-type: none"> • True: Das Ergebnis wird komplett als XML-Dokument ausgegeben. • False: Das Ergebnis wird als Base64-kodierte Zeichenkette ausgegeben.

Input

Im Input-XML-Dokument kann ein beliebig benanntes Wurzel-Element beliebig viele Kind-Elemente enthalten. Für jedes Kind-Element wird ein X4-Prozess aufgerufen und der Elementknoten als Input übergeben.

Beispiel-Input

```
<SomeRootElement>
  <SomeInputElement/>
  <SomeInputElement>with some content</SomeInputElement>
  <SomeInputElement someAttribute="someValue">more content</SomeInputElement>
</SomeRootElement>
```

Mit dem Parameter *InputIsBase64:False* wird das oben genannte Beispiel wie folgt ausgeführt:

Prozessaufruf 1

```
<SomeInputElement/>
```

Prozessaufruf 2

```
<SomeInputElement>with some content</SomeInputElement>
```

Prozessaufruf 3

```
<SomeInputElement someAttribute="someValue">more content</SomeInputElement>
```

Ausgabe

Das Ergebnis der Prozess-Ausführung könnte im obigen Beispiel wie folgt aufgebaut sein:

Output

```
<Results>
  <Result childProcessStatus="1" childProcessPid="4711" encoding="UTF-8" content-
type="text/plain">...some base64 string...</Result>
  <Result childProcessStatus="1" childProcessPid="4712" encoding="UTF-8" content-
type="text/plain">...some base64 string...</Result>
  <Result someAttribute="someValue" childProcessStatus="1" childProcessPid="4713"
encoding="UTF-8" content-type="text/plain">...some base64 string...</Result>
</Results>
```

Im Falle eines Fehlers wird kein Result-Element erzeugt, sondern z.B. folgendes Ergebnis ausgegeben:

Output im Fehlerfall

```
<Error attributesFromInput="">
  <TopException class="java.lang.NullPointerException">exception message</
TopException>
  <CausedByException class="some.other.Exception">exception message</
CausedByException>
</Error>
```

6.21.14 Process Starter

Process Starter: Führt einen technischen Prozess in einem beliebigen Benutzerordner in <X4>/X4DB mit einer Prozessinstanz-ID synchron oder asynchron aus. Als Input erhält der gestartete Prozess den Input des Adapters. Wenn der Funktions-Adapter *asynchron* ausgeführt wird, dann gibt der Adapter unmittelbar ein XML-Dokument aus, das die Prozessinstanz-ID des gestarteten Prozesses enthält; Wenn der Funktions-Adapter *synchron* ausgeführt wird, dann wartet er auf das Ergebnis des gestarteten technischen Prozesses (falls vorhanden) und gibt dieses aus.

i Mit dem *Process Starter* können Sie über *dynamische Parameter* den Prozessnamen (und -pfad) zur Laufzeit setzen. Außerdem besitzt jeder über den *Process Starter Adapter* gestartete technische Prozess eine eigene Prozessinstanz-ID, wird in einem separaten Thread ausgeführt und er kann daher nicht auf Variablen des aufrufenden Prozesses zugreifen.

Allgemeine Eigenschaften und Parameter

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Run</i> : Prozess starten
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.processstarter.ProcessStarter</i> : Hauptklasse (Standard)
<i>User</i>	Benutzer-ID (X4DB-Benutzerordner, in dem sich der zu startende Prozess befindet) Mögliche Werte: <ul style="list-style-type: none"> • Beliebige gültige Benutzer-ID (z.B. <i>999</i>) • <i>1</i>: Standard-Benutzerordner (Standard)
<i>Process</i>	Repository-Pfad zu einem technischen Prozess, der die Eigenschaft <i>In service: true</i> besitzt Mögliche Werte: Beliebiger Pfad im X4 Repository ohne führenden Schrägstrich (z.B. <i>Project/Folder/Process.wrf</i>)
<i>synchron</i>	Prozess synchron oder asynchron ausführen Mögliche Werte: <ul style="list-style-type: none"> • <i>True</i>: Warten, bis die gestartete Prozessinstanz beendet ist, und ggf. deren Ergebnis ausgeben, siehe unten (Standard) • <i>False</i>: Die Prozessinstanz-ID in einem XML-Dokument ausgeben und unmittelbar den darauf folgenden Prozess-Schritt ausführen, siehe unten

<i>forceNewThread</i>	<p>Auch bei synchroner Ausführung einen neuen Thread starten</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>True</i>: Bei synchroner Ausführung einen neuen Thread starten; Diese Option ermöglicht, den mit den Threads assoziierten Zustand besser zu verwalten, z.B. bei Transaktionen. (Standard) • <i>False</i>: Bei synchroner Ausführung keinen neuen Thread starten (ursprüngliches Verhalten)
-----------------------	---

Synchron gestartete technische Prozesse

Wenn Sie technische Prozesse synchron starten (mit Parameter *synchron*: *True*), dann wartet der Adapter, bis der gestartete technische Prozess vollständig ausgeführt wurde und gibt ggf. dessen Ergebnis inklusive Statuscode aus, bevor der nächste Prozess-Schritt ausgeführt wird.

Statuswerte

<i>1(successful)</i>	<p>(Oder jeder andere positive Wert) Der Adapter wurde erfolgreich <i>synchron</i> ausgeführt. Der Statuscode des synchron gestarteten technischen Prozesses wird vom <i>Process Starter</i> übernommen oder ggf. verändert ausgegeben (siehe Hinweis unten).</p> <p>Wenn der Adapter Status <i>1</i> ausgibt, heißt das nicht notwendigerweise, dass der gestartete Prozess ein Ergebnis ausgibt.</p>
<i>0(successful without result)</i>	<p>Der zu startende technische Prozess ist entweder nicht aktiv (Eigenschaft <i>In service</i>: <i>false</i>) und wurde daher nicht ausgeführt, oder im gestarteten technischen Prozess werden leere Bausteine verwendet, oder der gestartete Prozess gibt den Status <i>0</i> aus.</p>
<i>-1(error)</i>	<p>Der Funktions-Adapter konnte nicht ausgeführt werden (der zu startende Prozess wurde nicht gefunden oder der gestartete technische Prozess wirft Fehler).</p>
<i>(Jeder negative Wert)</i>	<p>Der gestartete technische Prozess wurde möglicherweise ausgeführt und erzeugte einen negativen Statuswert, z.B. über ein XSL-Mapping mit <i>ConditionState</i>, siehe Dynamische Statuswerte mit <i>ConditionState</i>.</p> <p>Bei asynchron gestarteten Prozessen wird deren Status vom <i>Process Starter</i> übernommen. Die weitere Prozessaufführung wird abgebrochen, falls die Prozess-Eigenschaft <i>Stop on error</i>: <i>true</i> ist.</p>

Beachten Sie:

- Wenn ein synchron gestarteter Prozess einen Statuswert zwischen *950* und *999* zurückgibt, wird dieser Status vom *Process Starter* verändert (Statuswert minus 50) und es werden Statuswerte zwischen *900* und *949* ausgegeben. Dies ist erforderlich, um die *Save Point*-Funktion des *X4 Servers* nicht zu beeinflussen.
- Wenn ein synchron gestarteter Prozess einen Statuswert zwischen *-999* und *-950* zurückgibt, wird dieser Status vom *Process Starter* verändert (Statuswert plus 50) und es werden Statuswerte zwischen *-900* und *-949* ausgegeben. Dies ist erforderlich, um Fehler in der Prozess-Ausführung zu vermeiden.

Asynchron gestartete X4-Prozesse

Wenn Sie X4-Prozesse asynchron starten (Parameter *synchron*: *False*), dann wird nach dem Start des aufgerufenen X4-Prozesses dessen Prozessinstanz-ID in einem XML-Dokument ausgegeben und der unmittelbar darauf folgende Prozess-Schritt ausgeführt.

Statuswerte

<i>1(successful)</i>	Der Funktions-Adapter wurde erfolgreich <i>asynchron</i> ausgeführt. Wenn der gestartete Prozess nicht aktiv ist (Eigenschaft <i>In service</i> : <i>false</i>) oder Fehler wirft, dann wird dennoch Status <i>1</i> ausgegeben; Das Ergebnis-XML-Dokument enthält in diesen Fällen die Prozessinstanz-ID <i>-1</i> .
<i>-1(error)</i>	Der Funktions-Adapter konnte nicht ausgeführt werden (der zu startende X4-Prozess wurde nicht gefunden).

6.21.15 Process Tester

Process Tester: Startet einen oder mehrere technische Prozesse mit frei definierbaren Inputdaten und vergleicht die Ergebnisse mit beliebigen Dokumenten. Der Adapter erwartet hierzu eine adapterspezifische Input-Struktur und erzeugt als Ergebnis ein XML-Dokument, in dem sämtliche getesteten Prozesse und deren Testergebnisse aufgelistet werden.

- Beachten Sie:** Dieser Adapter erfordert Zugriff auf das *X4 Repository* über das Dateisystem-Plug-in des *X4 Servers*. Technische Prozesse, die über das Classloader-Plug-in des *X4 Servers* geladen werden, können mit diesem Adapter also nicht verarbeitet werden, siehe *Handbuch X4 Server – Repository-Zugriffsmethode konfigurieren*.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <i>Execute</i> : Adapter ausführen
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.ProcessTester</i> : Hauptklasse (Standard)
----------------	--

Input

Der Adapter erwartet ein XML-Dokument mit beliebigem Wurzelementnamen, das nach folgendem Muster aufgebaut wurde:

```
<Input>
  <Process>
    <User><!--Benutzer-Repository ("user collection") --></User>
    <Name><!--Pfad zum technischen Prozess --></Name>
    <Files>
      <Input><!--Pfad zum Input-Dokument --></Input>
      <Output><!--Pfad zum Vergleichsdokument --></Output>
      <Result><!--Pfad zum Ergebnisdokument des technischen Prozesses --></
Result>
    </Files>
  </Process>
</Input>
```

Erläuterung:

- <Process> definiert einen auszuführenden technischen Prozess.
- <User> und <Name> definieren das Benutzerverzeichnis und den Pfad zum technischen Prozess.
- <Files> referenziert verschiedene Dokumente, siehe unten.
- <Input> referenziert ein Input-Dokument für den technischen Prozess.
- <Output> (optional) referenziert ein Vergleichsdokument.
- <Result> referenziert ein Dokument, in dem das tatsächliche Ergebnis des technischen Prozesses gespeichert werden soll.

Ausgabe

Der Adapter erzeugt ein XML-Dokument, in dem für jeden getesteten Prozess dessen Testergebnisse und Metadaten in einem Element <Process> in folgenden Attributen hinterlegt sind:

```
<Data>
  <Process
    Name="Pfad zum technischen Prozess"
    Input="Pfad zum Input-Dokument"
    ExecutionTime="Ausführzeit in Millisekunden"
    Result="Pfad zum Ergebnisdokument des technischen Prozesses"
    Equal="Übereinstimmend mit Vergleichsdokument: true/false"
    PID="Prozess-Instanz-ID" State="Ausführstatuswert"
    TimeNeededToProcessAFile="Testausführzeit in Millisekunden" />
</Data>
```


6.21.16 Project Access

6.21.16.1 Project Access


Project Access: Ermöglicht den Zugriff auf Dateien innerhalb eines Projektes, in Prozessbibliotheken und Adaptern.

 Dieser Adapter löst den [Repository Explorer \(veraltet\)](#) ab.

Eigenschaften (Properties)

Operation	Vom Adapter ausgeführte Operation
	Mögliche Werte: <ul style="list-style-type: none"> • ReadFile: Liest den Inhalt einer Datei aus. • WriteFile: Schreibt den Output in eine Datei. • Exists: Prüft, ob die angegebene Datei vorhanden ist • CreateDirectory: Erstellt ein Verzeichnis. • Delete: Löscht die Datei oder das Verzeichnis. • ListDirectory: Gibt den Inhalt des Verzeichnisses aus. • Move: Verschiebt die Datei • Copy: Kopiert die Datei

Parameter

Adapter	Hauptklasse des Adapters (Nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.system.repository.ProjectAccessAdapter:</i> Hauptklasse (Standard)
path	Pfad zur Datei bzw. dem Ordner, der verarbeitet werden soll. Mögliche Werte: Verzeichnispfad <div>  Der Projektpfad muss ohne den Projektnamen angegeben werden. Beispiel: <i>Processes/MyFolder</i> </div>
targetPath	Für Operationen copy und move: Ziel der zu kopierenden oder zu verschiebenden Datei/Ordner

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt.
0	Für die Operation Exists wurde keine Datei gefunden.
-1	Bei der Ausführung des Adapters ist ein Fehler aufgetreten.

Eingabe

- Operation `writeFile`: Inhalt der zu schreibenden Datei

Output

Der Adapter liefert abhängig von der ausgeführten Operation unterschiedliche Arten von Dokumenten zurück.

- Operation `readFile`: Inhalt der Datei
- Operation `writeFile`: Input entspricht Output
- Operation `delete`: Input entspricht Output
- Operation `createDirectory`: Input entspricht Output
- Operation `move`: Input entspricht Output
- Operation `copy`: Input entspricht Output
- Operation `listDirectory`:


Beispiel-Output für die Operation `listDirectory`


```
<Entries>
  <Entry isDirectory="true" name="MyFolder" path="Processes/MyFolder" />
  <Entry isDirectory="false" name="MyProcess.wrf" path="Processes/
MyProcess.wrf" />
</Entries>
```

6.21.16.2 Repository Explorer (veraltet)

Repository Explorer: Listet Dateien und Ordner eines bestimmten Ordnerpfads im X4 Repository (z.B. im Ordner `C:\X4\X4DB`) oder des gesamten Repositories.

Der Adapter gibt ein XML-Dokument aus, das jeweils die Datei- und Ordnernamen auflistet – je nach gewählter Operation lassen sich die Ordnerinhalte auch rekursiv (einschließlich Unterordnern) auflisten. Optional kann zudem der Inhaltstyp ermittelt werden, wobei bei XML-Dateien der jeweilige Wurzel-Elementname mit ausgegeben wird.

 Der *Repository Explorer* wird mit Version 6.2.0 der X4 Suite abgekündigt und in zukünftigen Versionen der X4 Suite nicht mehr unterstützt. Bitte verwenden Sie stattdessen den [Project Access](#) oder [X4 Projects](#) Adapter.

 Wenn über Parameter `DiscoverContent` der Inhaltstyp für die Repository-Dateien ermittelt wird, kann dies bei umfangreicheren Ordnerstrukturen mit zahlreichen Dateien längere Zeit in Anspruch nehmen!

Eigenschaften (Properties)

<i>Operation</i>	<p>Bestimmt, welche Operation der Adapter durchführt</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>List</i>: Datei- und Ordnernamen auflisten, die sich innerhalb von <i>StartPath</i> befinden • <i>DeepListFlat</i>: Datei- und Ordnernamen einschließlich aller Unterordner auflisten, die sich innerhalb von <i>StartPath</i> befinden • <i>DeepListNested</i>: wie <i>DeepListFlat</i>; das erzeugte XML-Dokument bildet jedoch die Ordnerstruktur hierarchisch ab • <i>DeepListFlatFromRoot</i>: Datei- und Ordnernamen des gesamten X4 Repositorys (X4DB) einschließlich aller Unterordner auflisten • <i>DeepListNestedFromRoot</i>: wie <i>DeepListFlatFromRoot</i>; das erzeugte XML-Dokument bildet jedoch die Ordnerstruktur hierarchisch ab
------------------	---

Parameter

<i>Adapter</i>	<p>Hauptklasse des Adapters (nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.repository.RepositoryAdapter</i>: Hauptklasse (Standard)</p>
<i>StartPath</i>	<p>Pfad im Repository</p> <p>Mögliche Werte: Beliebige Pfadangabe zum Repository-Ordner einschließlich des Benutzerordners (z.B. <i>1/Project/Folder</i>)</p>
<i>DiscoverContent</i>	<p>Inhaltstyp der Repository-Dateien prüfen (XML oder nicht-XML)</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>Ja</i>: Inhaltstyp von Repository-Dateien prüfen und im Ergebnisdokument jedem Element <i>File</i> ein Kindelement mit dem ermittelten Inhaltstyp hinzugefügt (kostet Performance!) • <i>Nein</i>: Keine Inhaltsprüfung (Standard)

Input

Dieser Adapter benötigt keinen Input.

Ausgabe

Der Adapter gibt je nach Operation ein XML-Dokument nach folgendem Muster aus, das Ordner in jeweils einem Element `<Directory>` und Dateien jeweils in einem Element `<File>` auflistet:

```
<Repository repositoryPath="Absoluter Repository-Pfad im Dateisystem">

  <!-- Bei Ordnern: -->
  <Directory name="Ordnername"
    path="Repository-Pfad des Ordners inkl. User Collection"
    fullpath="Absoluter Repository-Pfad des Ordners im Dateisystem" />

  <!-- Bei Dateien: -->
  <File name="Dateiname"
    path="Repository-Pfad der Datei inkl. User Collection"
    fullpath="Absoluter Repository-Pfad der Datei im Dateisystem">

    <!-- falls 'DiscoverContent' aktiviert ist: -->
    <Inhaltstyp: XML oder Non-XML>
      <!-- falls Inhaltstyp XML ist: -->
      <Wurzel-Elementname />
    </Inhaltstyp: XML oder Non-XML>
  </File>
</Repository>
```

Beispiel

Im folgenden Beispiel wird von einem Ordner Folder mit Operation DeepListNested die Ordnerstruktur hierarchisch ausgegeben und über Parameter DiscoverContent der Inhaltstyp von Dateien ermittelt. Für die XML-Datei File.xml wird hierbei dessen Wurzel-Element ausgegeben:

```
<Repository repositoryPath="c:\X4\X4DB">
  <Directory name="Folder" path="/1/Project/Data"
    fullpath="c:/X4/X4DB/1/Project/Data">
    <Directory name="EmptySubfolder" path="/1/Project/Data/Folder"
      fullpath="c:/X4/X4DB/1/Project/Data/Folder"/>
    <File name="File.txt" path="/1/Project/Data/Folder"
      fullpath="c:/X4/X4DB/1/Project/Data/Folder">
      <Non-XML/>
    </File>
    <File name="File.xml" path="/1/Project/Data/Folder"
      fullpath="c:/X4/X4DB/1/Project/Data/Folder">
      <XML><RootElement/></XML>
    </File>
  </Directory>
</Repository>
```

6.21.16.3 X4 Projects

X4 Projects: Ermöglicht den Zugriff auf Projekte innerhalb des X4 Repositorys.

 Dieser Adapter löst den [Repository Explorer \(veraltet\)](#) ab.

Eigenschaften (Properties)

Operation	<p>Vom Adapter ausgeführte Operation</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • Create: Erzeugt ein neues Projekt. Erforderliche Parameter: projectName, projectType • List: Erzeugt eine Liste aller vorhandenen Projekte. • Delete: Löscht ein Projekt. Erforderliche Parameter: projectName
-----------	---

Parameter

Adapter	<p>Hauptklasse des Adapters (Nicht verändern!)</p> <p>Mögliche Werte: <i>de.softproject.integration.adapter.system.repository.ProjectAdapter:</i> Hauptklasse (Standard)</p>
projectName	<p>Name des Projekts, das erstellt oder gelöscht werden soll.</p> <p>Mögliche Werte: Zeichenkette ohne Leerzeichen und Sonderzeichen.</p>
projectType	<p>Art des Projektes, das erstellt werden soll.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>ESB Project</i> • <i>Web App Project</i> • <i>Adapter Project</i> • <i>Web Portal Project</i>

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt.
-1	Bei der Ausführung des Adapters ist ein Fehler aufgetreten.

Input

Der Adapter erwartet keinen Input. Vorhandener Input wird unverändert ausgegeben.

Output


- Der Adapter liefert bei den Operationen Create und Delete nur im Fehlerfall einen Output.
- Operation List:

Beispiel-Output für die Operation List

```
<Projects>
  <Project type="ESB" name="MyESBProject" />
  <Project type="BPM" name="MyBPMProject" />
</Projects>
```

6.21.17 Task Management


Task Management: Ermöglicht den schnittstellenbasierten Zugriff auf Human Tasks in Geschäftsprozessen mittels Integration Flows. Er bildet eine Schnittstelle, die modellierte Aufgaben und menschliche Interaktion auf Prozessebene verfügbar macht. Dies ermöglicht die Anbindung von kundenspezifischen Web Apps an Vorgänge, die mit BPMN modelliert werden oder auch die direkte Anbindung beliebiger Drittsysteme an die zentrale Vorgangssteuerung der X4 Suite.


 Für den Zugriff mit dem Adapter wird der angemeldete Benutzer in X4 Activities verwendet.

Eigenschaften

Operation	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • GetAllTasks: Alle Tasks für den angemeldeten Benutzer zurückgeben • GetTask: Einen Task für den angemeldeten Benutzer zurückgeben • EditTask: Die Aufgaben für andere Benutzer sperren und die Merkmale aktualisieren • CloseTask: Die Aufgabe mit einem Fachstatus abschließen und den Vorgang fortsetzen • DeclineTask: Die Aufgabe ablehnen und an den Eigentümer zurückgeben • ForwardTask: Die Aufgabe an eine Person oder Gruppe weitergeben • CancelTask: Die Bearbeitung der Aufgabe abbrechen und sie für andere Benutzer freigeben
-----------	--

Parameter

Adapter	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.x4.server.bpm.casemanagement.adapter.TaskManagement</i> : Hauptklasse (Standard)
limit	Seitengröße für den Abruf der Task-Liste  Der Wert ist optional. Ist der Parameter nicht gesetzt, werden alle Tasks ausgegeben. Mögliche Werte: Beliebige Zahl > 0

offset	Offset für den Abruf der Task-Liste  Der Wert ist optional. Ist der Parameter nicht gesetzt, werden alle Tasks ausgegebenen. Mögliche Werte: Beliebige Zahl > 0
state	Erforderlich. Zustandsfilter für den Abruf der Task-Liste Mögliche Werte: <ul style="list-style-type: none"> • <i>OPEN</i>: Offene Tasks • <i>IN_PROGRESS</i>: Task in Bearbeitung • <i>CLOSED</i>: Geschlossene Tasks
taskId	UUID des Tasks Mögliche Werte: Gültige UUID, z. B. <i>443c498a-5868-4110-891f-42e5cb8fc24e</i>
userForDebugging	Benutzer der zum Debugging verwendet werden soll, wenn keine Anmeldung in X4 Activities vorhanden ist Mögliche Werte: Gültiges Benutzer-Login, z.B. <i>admin</i>

Statuswerte

1	Die Operation wurde erfolgreich ausgeführt
-1	Die Operation konnte nicht ausgeführt werden

Input

Bei folgenden Operationen wird kein Input benötigt:

- GetAllTasks
- GetTask
- CancelTask

Folgende Operationen benötigen als Input ein XML-Dokument mit folgendem Aufbau:

- Operation EditTask

Beispiel-Input für die Operation EditTask

```
<?xml version="1.0" encoding="UTF-8"?>
<EditTask>
  <Features>
    <Feature name="string" displayName="string" type="String">Wert</Feature>
  </Features>
</EditTask>
```

- Operation CloseTask

Beispiel-Input für die Operation CloseTask

```
<?xml version="1.0" encoding="UTF-8"?>
<CloseTask>
  <Features>
    <Feature name="string" displayName="string" readOnly="true" type="String">Wert</Feature>
  </Features>
  <Action>string</Action>
</CloseTask>
```

- Operation DeclineTask

Beispiel-Input für die Operation DeclineTask

```
<?xml version="1.0" encoding="UTF-8"?>
<DeclineTask>
  <Comment>string</Comment>
</DeclineTask>
```

- Operation ForwardTask

Beispiel-Input für die Operation ForwardTask

```
<?xml version="1.0" encoding="UTF-8"?>
<ForwardTask>
  <Comment>string</Comment>
  <Deadline>2019-04-15T11:46:02.782Z</Deadline>
  <Assignment type="USER|GROUP">string</Assignment>
</ForwardTask>
```

Ausgabe

Bei der Operation CancelTask wird als Output das Input-Dokument ausgegeben.

Folgende Operationen geben als Output ein XML-Dokument mit folgender Struktur aus:

- Operation GetAllTasks


```
<?xml version="1.0" encoding="UTF-8"?>
<Tasks>
  <Task id="3fa85f64-5717-4562-b3fc-2c963f66afa6" created="2019-04-15T11:50:56.145Z" creator="string">
    <Title>string</Title>
    <Reminder>2019-04-15T11:50:56.145Z</Reminder>
    <Deadline>2019-04-15T11:50:56.145Z</Deadline>
    <Owner>string</Owner>
    <Assignment type="USER|GROUP">string</Assignment>
    <State>OPEN|IN_PROGRESS|CLOSED</State>
    <Completed>2019-04-15T11:50:56.145Z</Completed>
  </Task>
</Tasks>
```

- Operationen GetTask, EditTask, CloseTask, DeclineTask und ForwardTask

```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>
<Tasks>
  <Task creator="antragssteller" created="2019-05-27T11:32:45Z" id="63febd9e-ae4c-463f-8e6f-eb25561b7850">
    <Deadline>2019-05-30T11:32:45Z</Deadline>
    <Owner>admin</Owner>
    <Assignment type="GROUP">AAM</Assignment>
    <State>OPEN</State>
    <KeyIdentifier>2019-001</KeyIdentifier>
    <CaseId>707bff74-cee3-41fb-92ab-704cb35beab0</CaseId>
  </Task>
</Tasks>
```

6.21.18 User Identity Adapter

User Identity Adapter: Ermöglicht es, Informationen über den aktuell angemeldeten Benutzer abzurufen, z. B. Benutzername, Vor- und Nachname, eingestellte Sprache, E-Mail-Adresse und Gruppen, denen dieser Benutzer angehört.

Eigenschaften (Properties)

Operation	Vom Adapter ausgeführte Operation Mögliche Werte: <ul style="list-style-type: none"> • GetUserIdentity: Gibt die Informationen über den Benutzer zurück.
-----------	---

Parameter

Adapter	Adapter Hauptklasse (nicht ändern!) Mögliche Werte: <ul style="list-style-type: none"> • <i>de.softproject.x4.adapter.useridentity.UserIdentityAdapter</i>: Hauptklasse (default)
---------	--

debug	<p>Setzt den Adapter in den Debug-Modus, in dem er keine echten Benutzerdaten verwendet, sondern Daten, die manuell in weitere Adapterparameter eingetragen werden.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Debug-Modus ein, Daten aus Parametern verwenden • <i>false</i>: Debug-Modus aus, echte Benutzerdaten verwenden
generateTestGroup	<p>Legt fest, ob ein Test-Gruppeneintrag im Debug-Modus generiert werden soll.</p> <p> Funktioniert ausschließlich im Debug-Modus.</p> <p>Mögliche Werte:</p> <ul style="list-style-type: none"> • <i>true</i>: Beispielhaften Gruppeneintrag generieren • <i>false</i>: Keinen beispielhaften Gruppeneintrag generieren
testGroupName	Technischer Name des Test-Gruppeneintrags, wenn debug und GenerateTestGroup auf <i>true</i> gesetzt sind.
testGroupDisplayName	Angezeigter Name des Test-Gruppeneintrags, wenn debug und GenerateTestGroup auf <i>true</i> gesetzt sind.
username	Verwendeter Benutzername, wenn debug und GenerateTestGroup auf <i>true</i> gesetzt sind.
lastname	Verwendeter Nachname, wenn debug und GenerateTestGroup auf <i>true</i> gesetzt sind.
firstname	Verwendeter Vorname, wenn debug und GenerateTestGroup auf <i>true</i> gesetzt sind.
email	Verwendete E-Mail-Adresse, wenn debug und GenerateTestGroup auf <i>true</i> gesetzt sind.
timezone	<p>Benutzer-Zeitzone gemäß ISO 8601, die verwendet wird, wenn der Parameter debug aktiviert ist.</p> <p>Mögliche Werte: Zeitzone gemäß ISO 8601 duration format, z. B. <i>PT5H</i></p>

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt.
-1	Bei der Ausführung des Adapters ist ein Fehler aufgetreten.

Ausgabe

- Benutzerinformationen wurden erfolgreich abgerufen:

```

<Result>
  <Username>admin</Username>
  <Firstname>SP</Firstname>
  <Lastname>Admin</Lastname>
  <Language>en</Language>
  <Groups>
    <Group>
      <Name>Users</Name>
      <DisplayName>X4 Control Center User</DisplayName>
    </Group>
    <Group>
      <Name>Administrators</Name>
      <DisplayName>X4 Control Center Administrator</DisplayName>
    </Group>
  </Groups>
  <Timezone>
    <Offset>PT1H</Offset>
  </Timezone>
</Result>

```

- Benutzerinformationen wurden im Debug-Modus abgerufen:

```

<Result>
  <Username>myUser</Username>
  <Firstname>anyName</Firstname>
  <Lastname>userLastName</Lastname>
  <EMail>mail@softproject.de</EMail>
  <Groups />
  <Timezone>
    <Offset>PT5H</Offset>
  </Timezone>
</Result>

```

- Benutzerinformation wurde nicht abgerufen:

```

<?xml version="1.0" encoding="UTF-8"?>
<Result>NO_USER</Result>

```

- Der Adapter befindet sich im Debug-Modus und erzeugt einen beispielhaften Gruppeneintrag:

```

<Result>
  <Username>myUser</Username>
  <Firstname>anyName</Firstname>
  <Lastname>userLastName</Lastname>
  <EMail>mail@softproject.de</EMail>
  <Groups>
    <Group>
      <Name>group</Name>
      <DisplayName>MyGroupName</DisplayName>
    </Group>
  </Groups>
  <Timezone>
    <Offset>PT5H</Offset>
  </Timezone>
</Result>

```


6.21.19 Web App Translation

Web App Translation: Gibt die lokalisierten Werte für definierte Translation Keys zurück.

Eigenschaften (Properties)

Operation	Vom Adapter ausgeführte Operation Mögliche Werte: <ul style="list-style-type: none"> GetTranslation: Gibt die lokalisierten Werte für definierte Translation Keys zurück.
-----------	--

Parameter

Adapter	Adapter Hauptklasse (nicht ändern!) Mögliche Werte: <ul style="list-style-type: none"> <i>de.softproject.x4.server.webapp.translation.adapter.TranslationAdapter</i>: Hauptklasse (Standard)
Language	Sprache, für die die Werte ausgelesen werden. Mögliche Werte: Zeichenkette mit dem definierten Sprachschlüssel aus der Web-App-Definition
keys	Translation Key/-s, der/die ausgelesen werden sollen.  Wenn mehrere Translation Keys ausgelesen werden, müssen diese durch ; getrennt werden. Mögliche Werte: Zeichenkette mit der/den definierten Translation Key/-s aus der Sprachdatei.

Statuswerte

1	Die Adapter-Operation wurde erfolgreich ausgeführt.
---	---

-1	Bei der Ausführung des Adapters ist ein Fehler aufgetreten.
----	---

Ausgabe

- Sprachdatei wurde erfolgreich ausgelesen:


1	<?xml version="1.0" encoding="UTF-8"?>
2	<Result>
3	<Output key="loginLabel" value="User"/>
4	<Output key="passwordLabel" value="Password"/>
5	</Result>

- Es wurde keine Sprache gesetzt:

1	<?xml version="1.0" encoding="UTF-8"?>
2	<Result>No language set</Result>

- Es wurde kein Translation Key gesetzt:

1	<?xml version="1.0" encoding="UTF-8"?>
2	<Result>No keys set</Result>

 Wenn ein Translation Key, der nicht definiert ist, abgefragt wird, wird statt des Wertes der Translation Key selbst zurückgegeben. Wenn eine Sprachdatei, die nicht existiert abgefragt werden soll, wird ebenfalls der Translation Key selbst zurückgegeben.

6.21.20 X4 BAM Logging


X4 BAM Logging: Gibt eine Log-Nachricht an die Monitoring-Oberfläche im X4 Control Center mit einem benutzerdefinierten Statustext aus oder speichert Daten im Data Log.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>LogStatus</i>: Benutzerdefinierten Statustext ausgeben, der in Parameter status angegeben wird • <i>LogMessage</i>: Log-Nachricht mit Daten aus dem vorangegangenen Prozess-Schritt im <i>Content Log</i> ausgeben • <i>LogBatchMessages</i>: Mehrere Nachrichten, die über eine spezifische Input-Struktur übergeben werden, protokollieren • <i>LogData</i>: Daten aus dem vorangegangenen Prozess-Schritt als Datei ins <i>Content Log</i> speichern, deren Dateiname in Parameter filename angegeben wird
------------------	---

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <code>de.softproject.integration.adapter.bam.BAMAdapter</code> : Hauptklasse (Standard)
<i>status</i>	Benutzerdefinierter Statustext, der in X4 BAM in der Spalte User Status ausgegeben wird (wenn Operation auf LogStatus gesetzt ist) Mögliche Werte: Beliebige Zeichenkette
<i>filename</i>	Dateiname einer im Content Log angehängten Datei (wenn Operation auf LogData gesetzt ist) Mögliche Werte: Beliebige Zeichenkette (z. B. <i>Data.xml</i>)
<i>key</i>	Kennzeichnet Logeinträge mit einem benutzerdefinierbaren Schlüssel, um später anhand von diesen Schlüsselwörtern Auswertungen der BAM-Datenbank erstellen zu können. Mögliche Werte: Beliebige Zeichenkette

 Wenn Sie Prozesse im X4 Designer im Debug-Modus ausführen, werden keine Daten mit X4 BAM geloggt.

Input

Dieser Funktions-Adapter erwartet für die Operation `LogBatchMessages` eine adapterspezifische Input-XML-Struktur, die Log-Nachrichten enthält:

```
<RootElementBAM>
  <MESSAGE level="Ebene" key="Schlüssel">
    <!--Log-Nachrichtentext-->
  </MESSAGE>
</RootElementBAM>
```

Beispiel

Im Folgenden ein Beispiel für ein Input-XML-Dokument für die Operation `LogBatchMessages`, das zwei Log-Nachrichten enthält, die vom Adapter direkt im *Content Log* von X4 BAM ausgegeben werden:

```
<RootElementBAM>
  <MESSAGE level="INFO" key="">Log Message 1</MESSAGE>
  <MESSAGE level="WARN" key="">Log Message 2</MESSAGE>
</RootElementBAM>
```

6.21.21 X4 Management



X4 Management: Ruft die Methoden des X4-spezifischen JMX-MBeans X4 Management auf. Verfügbar sind die Methoden *SetAllOutOfService*, *SetAllInService* und *ResetCache*.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Adapter durchführt Mögliche Werte: <ul style="list-style-type: none"> • <i>SetAllOutOfService</i>: Setzt bei <i>allen</i> Prozessen die Eigenschaft <i>In service</i> auf <i>false</i>. Dies bewirkt, dass keine Prozesse mehr gestartet werden. • <i>SetAllInService</i>: Setzt bei <i>allen</i> Prozessen die Eigenschaft <i>In service</i> auf <i>true</i>. Dies bewirkt, dass alle Prozesse (wieder) gestartet werden können. • <i>ResetCache</i>: Leert den Cache, der im Produktivmodus des X4 Servers Repository-Elemente zur Performance-Optimierung zwischenspeichert
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.x4management.X4ManagementCheckerAdapter</i> : Hauptklasse (Standard)
----------------	---

6.21.22 X4 Server Config Reader



X4 Server Config Reader: Gibt Java-Parameter aus dem X4-Startup-Skript sowie X4-Konfigurations-Eigenschaften aus der Datei *x4config.xml* in einem XML-Dokument aus. Die gewünschten Parameter werden dabei in einem spezifisch aufgebauten Input-XML-Dokument genannt und vom Adapter um die entsprechenden Werte ergänzt wieder ausgegeben.

Eigenschaften (Properties)

<i>Operation</i>	Bestimmt, welche Operation der Funktions-Adapter durchführt Mögliche Werte: <i>GetProperties</i> : Konfigurations-Eigenschaften ausgeben
------------------	--

Parameter

<i>Adapter</i>	Hauptklasse des Adapters (nicht verändern!) Mögliche Werte: <i>de.softproject.integration.adapter.ConfigReader:</i> Hauptklasse (Standard)
----------------	--

Input

Dieser Funktions-Adapter erwartet eine adapterspezifische Input-XML-Struktur. Es können beliebig viele `<Param/>`-Elemente eingetragen werden; das Element `<X4Config/>` oder `<JavaConfig/>` kann je nach Bedarf weggelassen werden.

```
<Config>
  <X4Config>
    <Param><Key><!--Element in X4Config.xml--></Key></Param>
    ...
  </X4Config>
  <JavaConfig>
    <Param><Key><!--Java-Option (in startX4.bat)--></Key></Param>
    <!-- weitere <Param/>-Elemente-->
  </JavaConfig>
</Config>
```

Ausgabe

Dieser Funktions-Adapter ergänzt das Input-XML-Dokument sofern möglich um die dynamisch ausgelesenen Werte, beispielsweise:

```
<Config>
  <X4Config>
    <Param>
      <Key><!--Element in X4Config.xml--></Key>
      <Value><!--Konfigurationswert des entsprechenden Elements--></Value>
    </Param>
  </X4Config>
  <JavaConfig>
    <Param>
      <Key><!--Java-Option (in startX4.bat)--></Key>
      <Value><!--Aktueller Wert der Java-Option--></Value>
    </Param>
  </JavaConfig>
</Config>
```

Beispiel

Im Folgenden ein Beispiel für ein ausgegebenes XML-Dokument, das *X4*-Konfigurationsschlüssel- und Werte sowie Java-Optionen und deren Werte enthält:


```
<Config>
  <X4Config>
    <Param>
      <Key>x4.xstore.repositoryPath</Key>
      <Value>C:/X4/X4DB</Value>
    </Param>
  </X4Config>
  <JavaConfig>
    <Param>
      <Key>x4.config.file</Key>
      <Value>X4config.xml</Value>
    </Param>
    <Param>
      <Key>javax.xml.transform.TransformerFactory</Key>
      <Value>net.sf.saxon.TransformerFactoryImpl</Value>
    </Param>
  </JavaConfig>
</Config>
```