

Workday Studio

Product Summary

March 8, 2026



Inhalt

Überlegungen zum Setup: Workday Studio.....	10
Grundlagen von Studio.....	12
Workday Studio für Windows herunterladen und installieren.....	12
Workday Studio für macOS herunterladen und installieren.....	14
Konzept: Workday Studio.....	16
Konzept: Lebenszyklus der Integrationsentwicklung.....	18
Konzept: Objekt „MediationContext“.....	19
Konzept: Integrationssysteme und Ereignisse.....	20
Referenz: Grundlagen der Benutzeroberfläche.....	20
Integrationsdesign.....	22
Assemblys.....	22
Assembly-Projekt erstellen.....	22
Schritte zu Komponenten hinzufügen.....	22
Untergeordnete Elemente hinzufügen.....	23
Assembly-Elemente verbinden.....	24
Eigenschaften der Assembly-Elemente konfigurieren.....	24
Sammlungen und Projekte verwalten.....	26
Namespace-Präfixe deklarieren.....	26
Pfadvariablen erstellen.....	27
Konzept: Assembly-Palette.....	27
Konzept: Komponenten und Schritte.....	29
Konzept: Anmerkungen zur Assembly.....	30
Konzept: Verknüpfte Ressourcendateien für Assembly-Projekte.....	31
Konzept: Schnellansicht der Gliederung.....	31
Referenz: Assembly-Eigenschaften.....	31
Referenz: Swimlane-Eigenschaften.....	32
Referenz: Eigenschaften der Verbindungslinie.....	33
Assembly-Transporte.....	33
Enumerationen als Startparameter in Studio verwenden.....	33
Konzept: Assembly-Transporte.....	34
Konzept: Enumerationen.....	36
Referenz: Eigenschaften des Transports „Workday-In“.....	37
Referenz: Eigenschaften des Transports „Local-In“.....	38
Referenz: Eigenschaften des Transports „Workday-Out-Rest“.....	40
Referenz: Eigenschaften des Transports „Workday-Out-Soap“.....	41
Referenz: Eigenschaften des Transports „Local-Out“.....	43
Referenz: Eigenschaften des Transports „HTTP-Out“.....	44
Referenz: Eigenschaften des Transports „FTP-Out“.....	48
Referenz: Eigenschaften für Transport „FTPS-Out“.....	53
Referenz: Eigenschaften des Transports „SFTP-Out“.....	59
Referenz: Eigenschaften des Transports „XMPP-Out“.....	64
Referenz: Eigenschaften des Transports „Custom-Out“.....	66
Referenz: Eigenschaften für Transport „Email-Out“.....	68
Untergeordnete Elemente des „HTTP-Out“-Transports.....	76
Konzept: Untergeordnete Elemente des „HTTP-Out“-Transports.....	76

Referenz: Eigenschaften von „HTTP-Basic-Auth“.....	76
Referenz: Eigenschaften von „HTTP-Custom-Auth“.....	77
Referenz: HTTPS-Eigenschaften.....	78
Untergeordnete Elemente des „FTPS-Out“-Transports.....	79
Konzept: Untergeordnete Elemente des „FTPS-Out“-Transports.....	79
Referenz: Eigenschaften der Proxy-Eigenschaften.....	80
Referenz: Eigenschaften der Firewall-Eigenschaften.....	80
Referenz: Eigenschaften von „Client-Key“.....	81
Referenz: Eigenschaften von „Server-Key“.....	81
Untergeordnete Elemente von SFTP-Out.....	82
Konzept: Untergeordnete Elemente von SFTP-Out.....	82
Referenz: Eigenschaften der Proxy-Eigenschaften.....	82
Referenz: Eigenschaften des privaten Schlüssels.....	83
Komponenten.....	83
Konzept: Komponenten.....	83
Referenz: Eigenschaften der Komponente „Async-Mediation“.....	84
Referenz: Eigenschaften der Komponente „Sync-Mediation“.....	85
Referenz: Eigenschaften der Komponente „Custom-Mediation“.....	86
Referenz: Eigenschaften der Komponente „Route“.....	87
Referenz: Eigenschaften der Komponente „Splitter“.....	87
Referenz: Eigenschaften der Komponente „Aggregator“.....	87
Untergeordnete Elemente der Komponente „Route“.....	88
Konzept: Untergeordnete Elemente der Komponente „Route“.....	88
Referenz: Eigenschaften „Custom-Strategy“.....	89
Referenz: Eigenschaften der Regex-Strategie.....	90
Referenz: Eigenschaften der MVEL-Strategie.....	90
Referenz: Eigenschaften der Round-Robin-Strategie.....	90
Referenz: Eigenschaften der XPath-Strategie.....	91
Referenz: Eigenschaften der Failover-Strategie.....	91
Referenz: Eigenschaften für alle Strategien.....	92
Referenz: Eigenschaften des „Doc-Iterator“.....	92
Referenz: Eigenschaften der Schleifenstrategie.....	92
Referenz: Eigenschaften „Choose-Route“.....	93
Referenz: Eigenschaften „Sub-Route“.....	93
Untergeordnete Elemente der Komponente „Splitter“.....	94
Konzept: Untergeordnete Elemente der Komponente „Splitter“.....	94
Referenz: Eigenschaften des Custom-Collater.....	95
Referenz: Eigenschaften des Standard-Splitter.....	95
Referenz: Eigenschaften des XPath-Splitters.....	95
Referenz: Eigenschaften des XML-Stream-Splitter.....	96
Referenz: Eigenschaften des JSON-Splitters.....	96
Referenz: Eigenschaften des Unzip-Splitters.....	96
Referenz: Eigenschaften des Mtable-Splitter.....	97
Untergeordnete Elemente des Standard-Splitters.....	97
Konzept: Untergeordnete Elemente des Standard-Splitters.....	97
Referenz: Eigenschaften für „Header-Ends-With“.....	98
Referenz: Eigenschaften von „Header-Fixed-Lines“.....	98
Referenz: Eigenschaften von „Content-Fixed-Lines“.....	98
Referenz: Eigenschaften von „Content-Starts-Ends“.....	98
Referenz: Eigenschaften von „Content-Starts-With“.....	98
Referenz: Eigenschaften von „Footer-Fixed-Lines“.....	99
Referenz: Eigenschaften von „Footer-Starts-With“.....	99
Untergeordnete Elemente der Komponente „Aggregator“.....	99
Konzept: Untergeordnete Elemente der Komponente „Aggregator“.....	99
Referenz: Eigenschaften der „Custom-Batch-Strategy“.....	100
Referenz: Eigenschaften der „Size-Batch-Strategy“.....	100

Referenz: Eigenschaften der „Time-Batch-Strategy“	100
Referenz: Eigenschaften des Custom-Collaters	101
Referenz: Eigenschaften des Message-Content-Collater	101
Referenz: Eigenschaften des XML-Message-Content-Collater	102
Referenz: Eigenschaften des JSON-Aggregators	104
Referenz: Eigenschaften des Zip-File-Collater	104
Referenz: Eigenschaften des Mtable-Collater	105
Allgemeine Komponenten	106
Konzept: Allgemeine Komponenten	106
Referenz: Eigenschaften der FTP-Subassembly	109
Referenz: Eigenschaften der Subassembly „GetEventConfigurations“	113
Referenz: Eigenschaften der Subassembly „GetEventDocuments“	115
Referenz: Eigenschaften der Subassembly „GetIntegrationEvent“	117
Referenz: Eigenschaften der Subassembly „GetIntegrationSystems“	119
Referenz: Eigenschaften für Subassembly „PagedGet“	120
Referenz: Eigenschaften der Subassembly „PagedGetLocalPaging“	124
Referenz: Eigenschaften der Subassembly „PdfPrintStep“	127
Referenz: Eigenschaften der Subassembly „PutIntegrationEvent“	130
Referenz: Eigenschaften der Subassembly „PutIntegrationMessage“	132
Referenz: Eigenschaften der Subassembly „SalesforceConnector“	136
Die PrismAnalytics-Untergruppe	138
Fehlerbehandlungskomponenten	152
Konzept: Fehlerbehandlungskomponenten	152
Referenz: Komponenteneigenschaften für benutzerdefinierte Fehlerbehandlung	153
Referenz: Eigenschaften der Komponente „Log-Error“	155
Referenz: Eigenschaften der Komponente „Send-Error“	157
Transformationsschritte	158
Text mit einer Textschemadatei in XML konvertieren	158
Konzept: Transformationsschritte	161
Referenz: Eigenschaften des Schritts „XSLT“	163
Referenz: Eigenschaften des Schritts „XSLT+“	166
Referenz: Eigenschaften des Schritts „FOP“	169
Referenz: Eigenschaften des Schritts „Text-Excel“	173
Referenz: Eigenschaften des Schritts „TextSchema“	175
Referenz: Eigenschaften des Schritts „Wrap-Soap“	178
Referenz: Eigenschaften des Schritts „XML-to-Java“	180
Referenz: Eigenschaften des JSON-Transformators	181
Referenz: Eigenschaften des Schritts „XML-to-JSON“	183
Referenz: Eigenschaften des Schritts „JSON-to-XML“	185
Referenz: Eigenschaften des Schritts „Java-to-XML“	187
Referenz: Eigenschaften des Schritts „XML-to-CSV“	189
Referenz: Eigenschaften des Schritts „CSV-to-XML“	192
Referenz: Eigenschaften des Schritts „Character-Conversion“	195
Referenz: Eigenschaften des Schritts „Mtable-Builder“	197
Referenz: Eigenschaften des Schritts „Mtable-Writer“	198
Referenz: Eigenschaften des Schritts „ETV“	199
Referenz: Eigenschaften des Schritts „XTT“	201
Untergeordnete Elemente des Mtable-Builder	203
Konzept: Untergeordnete Elemente des Mtable-Builders	203
Referenz: Eigenschaften des CSV-Mtable-Reader	203
Referenz: Eigenschaften des JSON-Mtable-Reader	204
Untergeordnete Elemente des Mtable-Writer	205
Konzept: Untergeordnete Elemente des Mtable-Writer	205
Referenz: Eigenschaften des CSV-Mtable-Writer	205
Referenz: Eigenschaften des JSON-Mtable-Writer	205
Protokollierungs- und Ereignisschritte	206

Konzept: Protokollierungs- und Ereignisschritte.....	206
Referenz: Eigenschaften des Schritts „Log“.....	207
Referenz: Eigenschaften des Schritts „Cloud-Log“.....	209
Schritte zur Nachrichtенbearbeitung.....	211
Konzept: Schritte zur Nachrichtенbearbeitung.....	211
Referenz: Eigenschaften des Schritts „Copy“.....	211
Referenz: Eigenschaften des Schritts „JavaScript“.....	214
Referenz: Eigenschaften des Schritts „Set-Headers“.....	216
Referenz: Eigenschaften des Schritts „Write“.....	217
Validierungsschritte.....	219
Konzept: Validierungsschritte.....	219
Referenz: Eigenschaften des Schritts „Eval“.....	220
Referenz: Eigenschaften des Schritts „Validieren“.....	221
Referenz: Eigenschaften des Schritts „Validate-Exp“.....	224
Referenz: Eigenschaften des Schritts „Xpath validieren“.....	225
Codierungsschritte.....	227
Konzept: Codierungsschritte.....	227
Konzept: PGP-Dateinamen.....	228
Referenz: Eigenschaften des Schritts „PGP-Encrypt“.....	230
Referenz: Eigenschaften des Schritts „PGP-Decrypt“.....	233
Referenz: Eigenschaften des Schritts „Base64-Decode“.....	235
Referenz: Eigenschaften des Schritts „Base64-Encode“.....	237
Referenz: Eigenschaften des Schritts „Zip“.....	239
Referenz: Eigenschaften des Schritts „Unzip“.....	241
Referenz: Eigenschaften des Schritts „Compress“.....	243
Referenz: Eigenschaften des Schritts „Decompress“.....	245
Andere Schritte.....	247
Konzept: Andere Schritte.....	247
Referenz: Eigenschaften des Schritts „Custom“.....	247
Referenz: Eigenschaften des Schritts „Set-Dynamic-Endpoint“.....	249
Referenz: Eigenschaften des Schritts „Store“.....	250
Referenz: Eigenschaften des Schritts „Retrieve“.....	254
Referenz: Eigenschaften des Schritts „Xmldiff“.....	256
Referenz: Eigenschaften des Schritts „Enqueue-Message“.....	260
Assembly-Module.....	262
Assembly-Module erstellen.....	262
Assembly-Module zu einem Projekt hinzufügen.....	263
Konzept: Assembly-Module.....	264
Subassemblies.....	265
Subassemblies erstellen.....	265
Konzept: Assemblies.....	265
Benutzerdefinierte Assembly-Komponenten.....	266
Benutzerdefinierte Java-Klassen und Spring-Beans erstellen.....	266
Konzept: Benutzerdefinierte Assembly-Elemente.....	268
Konzept: Verpackung von benutzerdefinierten Elementen.....	269
Spring Bean-Konfiguration.....	271
Konzept: Spring Bean-Konfiguration.....	271
Referenz: Globale Bean-Eigenschaften.....	272
Referenz: Individuelle Bean-Eigenschaften.....	272
Referenz: Alias-Eigenschaften.....	274
Referenz: Eigenschafts-Eigenschaften.....	274
Referenz: Constructor-Arg-Eigenschaften.....	274
Referenz: Referenzeigenschaften.....	275
Referenz: Listeneigenschaften.....	275
Referenz: Map-Eigenschaften.....	275
Message Builder.....	276

XML-Literale basierend auf Schema oder WSDLs erstellen.....	276
XPath-Ausdrücke basierend auf Schema oder WSDLs erstellen.....	277
Konzept: Workday Studio Message Builder.....	278
Referenz: Eigenschaften von Nachrichtenelementen.....	279
Transportbindungen.....	283
Konzept: Transportbindungen.....	283
Referenz: Eigenschaften für WSDL-HTTP-Bindung.....	284
Referenz: Eigenschaften für WSDL-SOAP-Bindung.....	285
Untergeordnete Elemente mit WSDL-SOAP-Bindung.....	285
Konzept: Untergeordnete Elemente mit WSDL-SOAP-Bindung.....	285
Referenz: Eigenschaften für benutzerdefinierte Richtlinie.....	286
Referenz: WS-Rm-Eigenschaften.....	286
Referenz: WS-Sicherheitseigenschaften.....	286
Referenz: WS-Ut-Eigenschaften.....	286
Schema Explorer.....	287
WSDL- und XSD-Dateien anzeigen.....	287
Workday Webservice-WSDLs anzeigen.....	288
Benutzerdefinierte RaaS-Berichtsschemas anzeigen.....	288
Konzept: Schema Explorer.....	289
MVEL.....	289
Konzept: MVEL-Grundlagen.....	289
Konzept: MVEL und Studio.....	293
Referenz: Studio-Hilfsobjekte.....	293
Referenz: Hilfsobjekt für den Dokumentenzugriff.....	298
Referenz: Helper-Objekt für Startparameter.....	303
Referenz: Util-Hilfsobjekt.....	306
Referenz: MVEL-Beispielausdrücke in Studio.....	308
Referenz: MVEL-Syntaxbeispiele für automatische Konvertierung in Studio.....	311
Referenz: MVEL-Methoden.....	314
Studio Integrationsservices.....	315
Attribut- und Map-Services erstellen.....	315
Übermittlungsservices erstellen.....	317
Listener-Services erstellen.....	318
Abrufservices erstellen.....	318
Sequenzgeneratorservice erstellen.....	319
Transaktionsprotokollservices erstellen.....	320
Berichtsservices erstellen.....	321
Services für benutzerdefinierte Objekte erstellen.....	322
Dateninitialisierungsservices erstellen.....	323
Servicereferenzen erstellen.....	323
Referenzen zu benutzerdefinierten Berichten über Transporte „Workday-Out-Rest“ hinzufügen.....	324
Referenzen zu benutzerdefinierten Objekten in Transporten „Workday-Out-Rest“ hinzufügen.....	325
CLAR-Dateien.....	326
Manifestdateien generieren.....	326
CLAR-Dateien importieren.....	327
CLAR-Dateien exportieren.....	327
Konzept: CLAR-Dateien.....	328
JAXB-Projekte.....	328
JAXB-Projekte erstellen.....	328
JAXB-Projekte als Abhängigkeiten in Assembly-Projekten hinzufügen.....	330
Konzept: JAXB-Unterstützung in Studio.....	331
Studio Ivy-Plugin.....	331
Ivy-Plugin konfigurieren.....	331
Workday Ivy-Arten zu Workday Studio-Projekten hinzufügen.....	331
Abhängigkeiten von Projekten mit Ivy-Definitionen auflösen.....	332

Abhängige JARs zu Projekten mit Ivy-Arten hinzufügen.....	332
Workday Ivy-Arten aus Studio-Projekten entfernen.....	333
Integrations-Deployment.....	334
Eine Verbindung hinzufügen.....	334
Eine Gruppe von Verbindungen hinzufügen.....	335
Eine Verbindung bearbeiten.....	335
Eine Gruppe von Verbindungen bearbeiten.....	336
Verbindungen entfernen.....	336
Verbindungen importieren.....	337
Verbindungen exportieren.....	337
Verbindung zu Workday herstellen.....	338
Integrationen in Workday bereitstellen.....	338
Bereitgestellte Integrationen in Workday anzeigen.....	339
Studio für die Verwendung eines HTTPS-Proxyservers konfigurieren.....	340
Konzept: Studio-Verbindungen.....	340
Konzept: Cloud Explorer.....	340
Referenz: Verbindungsfehler.....	341
Multi-Faktor-Authentifizierung in Workday Studio.....	342
Multi-Faktor-Authentifizierung in Workday Studio aktivieren.....	342
Integrationskonfiguration.....	344
Konto des Integrationssystembenutzers einer Integration zuweisen.....	344
Konzept: Sicherheit des Integrationssystems.....	344
Integration wird gestartet.....	345
Integrationen aus Studio starten.....	345
Start-URLs kopieren.....	345
Startkonfigurationen erstellen.....	346
Startkonfigurationen bearbeiten.....	347
Integrationsmonitoring und Problembehebung.....	348
Prozessmonitor.....	348
Konzept: Integrationsereignis-IDs.....	348
Referenz: Symboleleistenschaltflächen für Prozessmonitor.....	348
Betrachter für konsolidierte Berichte.....	349
Konsolidierte Berichte für Integrationen laden.....	349
Konsolidierte Berichte in Ihrem Arbeitsbereich speichern.....	349
Details zum Workday-Integrationsereignis mithilfe konsolidierter Berichte anzeigen.....	350
Anforderungsnachrichten für Integrationsereignisse mithilfe konsolidierter Berichte anzeigen....	351
Integrationen mit dem Ereignisprotokoll des Betrachters für konsolidierte Berichte überwachen.....	351
Integrationen mit dem Profilprotokoll des Betrachters für konsolidierte Berichte überwachen....	352
Assembly-Ausführungen für konsolidierte Berichte erneut wiedergeben.....	353
Auditereignisse für konsolidierten Bericht mit Assembly-Elementen verknüpfen.....	353
Anmerkungen zum konsolidierten Bericht in Assembly-Diagrammen anzeigen.....	354
Konzept: Betrachter für konsolidierte Berichte.....	354
Referenz: Fehlermeldungen im konsolidierten Bericht.....	355
AUnit-Assembly-Tests.....	358
Ein AUnit-Test-Skelett für ein Assembly-Projekt erstellen.....	358
Konzept: AUnit-Tests.....	359
Referenz: Anmerkungen zum AUnit-Test.....	361

Referenz: Aktionen des AUnit-Tests.....	362
Webservice-Tests.....	363
Webservices mit einer WSDL-Datei testen.....	363
Webservices mit einer XSD-Datei testen.....	364
Einzelne Webservices mit dem Schema Explorer testen.....	365
Get-Vorgänge und Anforderungsnachrichten mit dem Anforderungs-Builder-Assistenten testen.....	366
Authentifizierung für Webservice-Tester konfigurieren.....	366
X.509-Authentifizierung für Webservice-Tester konfigurieren.....	367
Webservice-Tester konfigurieren – Erweiterte WS-Sicherheit.....	368
Konzept: Webservice-Tests.....	369
Referenz: Schaltflächen der Symbolleiste des Webservice-Tester.....	369
Assembly-Debugging.....	371
Debug einer Assembly durchführen.....	371
Konzept: Assembly-Debugger.....	372

Management von Studio..... 373

Systemeinstellungen für Studio.....	373
Konzept: Systemeinstellungen des Assembly-Editors.....	373
Konzept: Systemeinstellungen der Verbindungen.....	374
Konzept: Deployment-Systemeinstellungen.....	375
Konzept: HTTPS-Systemeinstellungen.....	375
Konzept: Ivy-Systemeinstellungen.....	375
Konzept: Systemeinstellungen verknüpfen.....	375
Konzept: Systemeinstellungen für MVEL-Validierung.....	375
Konzept: Systemeinstellungen für Integrationspakete.....	376
Konzept: Systemeinstellungen „Laufzeit“.....	376
Konzept: Server-Systemeinstellungen.....	376
Konzept: Anmeldeeinstellungen für den Begrüßungsbildschirm.....	376
Konzept: Systemeinstellungen der Statusleiste.....	376
Konzept: Systemeinstellungen aktualisieren.....	376
Konzept: Systemeinstellungen des XPath-Explorer.....	376
Referenz: Systemeinstellungen für Debug-Laufzeit für Assembly.....	377
Referenz: Systemeinstellungen für Debug-Ansicht für Assembly.....	377
Referenz: Systemeinstellungen des Cloud Explorer.....	378
Referenz: Systemeinstellungen des Betrachters für konsolidierte Berichte.....	378
Referenz: Präferenzen für Java-Projekteinstellungen.....	379
Referenz: Benachrichtigungseinstellungen.....	379
Referenz: Systemeinstellungen des Prozessmonitors.....	380
Referenz: Systemeinstellungen des Webservice-Testers.....	380
Studio-Aktualisierungen.....	381
Studio-Aktualisierung.....	381
Laufzeitänderungen für Workday-Assembly.....	381
Konzept: Assembly-Versionen.....	381
Referenz: Versionsänderungen auf Schemaebene.....	382
Referenz: Versionsänderungen auf XML-Ebene.....	387
Referenz: Änderungen des Saxon-XSLT-Prozessors.....	388

Beispielkonfigurationen..... 390

Beispiel: Beispiel-Assembly „Hello Cloud“ ausführen.....	390
Beispiel: Beispiel-Assembly „Hello Workday Web Services“ ausführen.....	391
Beispiel: Beispiel-Assembly „Common Usage Patterns“ ausführen.....	392
Beispiel: Beispiel-Assembly „Debugging“ ausführen.....	394
Beispiel: Beispiel-Assembly „Workday-In Features“ ausführen.....	395

Beispiel: Beispiel-Assembly „Error Handling“ ausführen.....	397
Beispiel: Beispiel-Tests „AUnit“ ausführen.....	398
Beispiel: Beispiel-Assembly „FOPStep“ ausführen.....	399
Beispiel: Beispiel-Assembly „Organisationsdaten für einen Mitarbeiter extrahieren“ ausführen.....	400
Beispiel: Beispiel-Assembly „PrismAnalytics“ ausführen.....	402
Beispiel für Schnittstelle für Drittanbieter-Entgeltabrechnung.....	403
Schritte: Beispiel der Abrechnungsschnittstelle ausführen.....	403
Sicherheitsgruppe der Abrechnungsschnittstelle erstellen.....	404
PayData Extract-Integrationssystem erstellen.....	405
PayData Extract-Integrationssystem testen.....	406
Studio PayData-Integration bereitstellen.....	407
Integrationssystembenutzer der PayData-Integration zuweisen.....	408
PayData Extract- und PayData-Integrationen verknüpfen.....	408
Verknüpfte Integrationen starten.....	409

Überlegungen zum Setup: Workday Studio

Anmerkung: Die in diesem Abschnitt beschriebenen Lösungen sind nicht Teil des Workday-Service. Siehe [Rechtlicher Hinweis](#) für Details.

Dieses Thema kann als Entscheidungshilfe bei der Planung Ihrer Konfiguration und Verwendung von Workday Studio dienen. Darin wird Folgendes erläutert:

- Warum diese Integration eingerichtet werden sollte.
- Wie sie sich in Workday einfügt.
- Nachgelagerte Auswirkungen und produktübergreifende Interaktionen.
- Sicherheitsanforderungen und Geschäftsprozesskonfigurationen.
- Fragen und Einschränkungen, die vor der Implementierung zu berücksichtigen sind.

Ausführliche Konfigurationsdetails finden Sie in den detaillierten Anweisungen zur Aufgabe.

Worum es geht

Studio ist eine auf Eclipse™ basierende integrierte Entwicklungsumgebung (Integrated Development Environment, IDE), mit der Sie Folgendes ausführen können:

- Benutzerdefinierte Integrationen erstellen
- Massendaten mit externen Endpunkten austauschen
- Integrationen in Workday hochladen

Studio ist eine externe Softwareanwendung, die Sie separat von Workday installieren.

Vorteile und Nutzen

Studio verbessert die Produktivität, da es Ihnen folgende Möglichkeiten bietet:

- Integrationen erstellen, die eine größere Designflexibilität bieten als der EIB (Enterprise Interface Builder) oder Konnektorvorlagen.
- Den XML-Quellcode Ihrer benutzerdefinierten Integrationen debuggen.
- Große Datenmengen über Workday exportieren und importieren.
- Große Datensätze in ein bestimmtes Format transformieren. Beispiel: Sie können CSV-Datasets in das XML-Format umwandeln.

Anwendungsfälle

Sie haben folgenden Bedarf:

- Eine benutzerdefinierte Integration, die nicht über den EIB oder Konnektorvorlagen verfügbar ist. Beispiel: Eine Integration, die XTT-Transformationen (XML-to-Text) in einem Dataset implementiert.
- Eine Inbound-Integration, um Massendaten aus verschiedenen Datenquellen in Workday zu importieren.
- Eine Outbound-Integration, die Massendaten aus Workday in einem bestimmten Format exportiert.
- Zur Durchführung von Einheitentests im XML-Quellcode einer benutzerdefinierten Integration.

Fragestellungen

Frage	Überlegungen
Hat Ihre Organisation eine langfristige Supportvereinbarung mit Oracle?	Wenn ja, können Sie Studio mit dem Oracle JDK 1.8 installieren.

Frage	Überlegungen
	Wenn nein, können Sie Studio mit OpenJDK 8 installieren.
Welche Plattform verwenden Sie, um Studio zu installieren?	Installieren Sie die entsprechende Installationsdatei für Ihre Plattform: • MacOS (64-bit) • Windows (32-Bit) • Windows (64-Bit)
Haben Sie eine Schulung für Studio besucht?	Für Lernende, die an einem Studio-Training interessiert sind, bietet Workday den Kurs „Crestellung Integrations Using Workday Studio“ an. Weitere Informationen finden Sie im Trainingskatalog.

Empfehlungen

Workday empfiehlt, Ihre benutzerdefinierten Studio-Integrationen in Ihrer Vorschau-Sandbox-Umgebung bereitzustellen und zu testen, bevor Sie sie an Ihren Produktionsmandanten übertragen.

Anforderungen

Für Integrationen mit dem Studio ist ein Java™ Development Kit (JDK) erforderlich, das mit Java Version 8 kompatibel ist. Darüber hinaus erfordert die Eclipse-IDE ein JDK, das mit Java Version 17 kompatibel ist. Beachten Sie Folgendes:

- Sie haben eine Lizenzvereinbarung mit Oracle und verwenden Oracle JDK 1.8 und Oracle JDK 1.17.
- Wenn Sie keine Vereinbarung mit Oracle haben, verwenden Sie OpenJDK 8 und OpenJDK 17.

Beide JDKs sind erforderlich.

Einschränkungen

Die Größe der Ausgabedateien der Studio-Integration beträgt maximal:

- 1 GB (komprimiert) für einzelne Dateien.
- Insgesamt 3 GB (komprimiert) für alle Dateien.

Mandanten-Setup

Sie können auf den Quellcode von Studio-Integrationen zugreifen, indem Sie das Kontrollkästchen Quellcode erforderlich in der Aufgabe Mandanten-Setup bearbeiten - Integrationen in Workday aktivieren. Durch den Zugriff auf den Quellcode von Studio-Integrationen können Sie Fehler identifizieren und die Effizienz steigern.

Sicherheit

Es könnte sinnvoll sein, folgende Domänen im funktionalen Bereich „Integration“ einzurichten.

Domänen	Überlegungen
<i>Integration Build</i>	Stellen Sie Integrationen von Studio in Workday bereit.
<i>Integration Configure</i>	Bearbeiten Sie Studio-Integrationen in Workday.
<i>Integration Debug</i>	Debuggen Sie Studio-Integrationen in Workday.

Domänen	Überlegungen
<i>Integration Event</i>	Starten Sie Studio-Integrationen in Workday.
<i>Integration Security</i>	Erstellen Sie ein Konto des Integrationssystembenutzers (ISU) für Ihre Studio-Integrationen in Workday.

Geschäftsprozesse

Für Studio-Integrationen können Sie Folgendes konfigurieren:

- Geschäftsprozess *Integrationsprozessereignis* im funktionalen Bereich „Integration“, um zu steuern, wie eine Integration ausgeführt wird.
- Geschäftsprozessschritt *Integration*, um eine Integration während eines Geschäftsprozesses zu starten. Beispiel: Sie können einen Integrationsschritt zum Geschäftsprozess *Formular I-9 ausfüllen* hinzufügen. Dieser Schritt startet eine Integration, um Daten für den Mitarbeiter im Geschäftsprozess zu senden.

Reporting

Workday stellt folgende Berichte für Studio-Integrationen bereit:

Bericht	Beschreibung
Integrationsereignisse	Zeigt Folgendes an: • Eine Zusammenfassung der Integrationsereignisse, die in Bearbeitung oder abgeschlossen sind. • Den Status jedes Integrationsereignisses.
Integrationssystem anzeigen	Zeigt alle Integrationssysteme an, die in Ihrem Workday-Mandanten verfügbar sind.

Beziehungen und Touchpoints

Studio interagiert je nach Produktbereich, Zweck und Datenquellen Ihrer Integration mit verschiedenen Teilen von Workday:

- Inbound-Integrationen fügen Daten in Workday hinzu und aktualisieren sie.
- Outbound-Integrationen extrahieren Daten aus Workday, ändern die Daten jedoch nicht in Workday.

Workday bietet ein Touchpoints Kit mit Ressourcen, die Ihnen helfen, die Konfigurationsbeziehungen in Ihrem Mandanten besser zu verstehen. Weitere Informationen zum [Workday Touchpoints Kit](#) finden Sie in der Workday Community.

Grundlagen von Studio

Workday Studio für Windows herunterladen und installieren

Warum und wann dieser Vorgang ausgeführt wird

Workday Studio basiert auf der Eclipse-Entwicklungsplattform. Es umfasst mehr als 150 Eclipse-Plugins sowie Abhängigkeiten von Plattformen und Drittanbietern. Sie benötigen zwei Versionen des Java™ Development Kit (JDK):

- Für Eclipse ist ein JDK erforderlich, das mit Java, Version 17, kompatibel ist.
- Für die Laufzeit von Studio ist ein JDK erforderlich, das mit Java Version 8 kompatibel ist.

Installieren Sie beide JDKs und stellen Sie Folgendes sicher:

- Ihre Java-Laufzeitumgebung ist auf Java 8 gesetzt.
- Für Ihre JDK-Compiler-Compliance ist Java 1.8 festgelegt.

Prozedur

1. Laden Sie ein JDK herunter und installieren Sie es, das mit Java Version 17 kompatibel ist. Sie können die kostenlose Open-Source-Implementierung OpenJDK aus verschiedenen Quellen herunterladen.

Beispiel:

- Wählen Sie [Zulu OpenJDK 17](#) für Ihr Betriebssystem.

Anmerkung:

- Benutzer von Apple Verwalten, müssen ein Apple Resource Management JDK17 verwenden. Ein x86_64-JDK funktioniert nicht.
- Für Studio ist ein JDK erforderlich, das speziell mit Java 17 kompatibel ist, nicht mindestens mit Java 17.

2. Laden Sie ein JDK herunter und installieren Sie es, das mit Java Version 8 kompatibel ist. Auch hier können Sie die kostenlose Open-Source-Implementierung OpenJDK wählen.

Beispiel:

- Wählen Sie [Zulu OpenJDK 8](#) für Ihr Betriebssystem

Anmerkung:

- Benutzer von Apple Verwalten, müssen ein Apple Resource Management JDK8 verwenden. Ein x86_64-JDK funktioniert nicht.
- Für Studio ist ein JDK erforderlich, das speziell mit Java 8 kompatibel ist, nicht mindestens mit Java 8.

3. Laden Sie das [Installationsprogramm](#) für Workday Studio für Windows herunter.
4. Um die Installation zu starten, doppelklicken Sie auf die ausführbare Installationsdatei. Um Studio im Ordner C:\Programme zu installieren, müssen Sie die ausführbare Datei als Administrator ausführen, indem Sie sie mit der rechten Maustaste anklicken und Als Administrator ausführen auswählen.

Anmerkung:

Sie können die Anwendung „Eingabeaufforderung“ verwenden, um eine Studio-Installation im Hintergrund auszuführen. Navigieren Sie in der Eingabeaufforderung zu dem Verzeichnis, in dem Sie die Installationsdatei gespeichert haben, und geben Sie Folgendes ein:

```
<Installation_Filename>.exe /S /J=<Java_Home> /D=<Installation_Directory>
```

wobei das Argument:

- /S gibt die unbeaufsichtigte Installation an.
- /J gibt den JDK 17-Pfad an.
- /RD installiert den Workday Report Designer.
- /D Gibt den Installationsordner an (dieses Argument muss das letzte im Befehl sein).

Beispiel: `workday-studio-x86_64.exe /S /J=C:\Program Files\Zulu\zulu17.62.17-ca-jdk17.0.18-win_x64 /D=C:\Studio`

5. Lesen Sie die Lizenzvereinbarung und aktivieren Sie anschließend das Kontrollkästchen Annahme der Bedingungen. Klicken Sie auf Weiter.

6. Navigieren Sie zum Speicherort Ihrer JDK 17-Installation und klicken Sie auf Next. Beispiel: für JDK-Installationsorte: C:\Programme\Zulu\zulu17.62.17-ca-jdk17.0.18-win_x64.
7. Wenn die JVM-Details korrekt sind, klicken Sie auf "Bestätigen". Andernfalls klicken Sie auf Zurück.
8. Wählen Sie das Paket Report Designer, um es zusammen mit Workday Studio zu installieren. Sie können den Report Designer auch später aus Studio installieren.
9. Navigieren Sie zu dem Verzeichnis, in dem Sie Studio installieren möchten. Klicken Sie auf Weiter.
10. Wenn Studio die Meldung „Installation abgeschlossen“ anzeigt, klicken Sie auf "Weiter" und dann auf "Fertig stellen".

Anmerkung: Eclipse mit Java 17 kann keinen Arbeitsbereich verwenden, der von Eclipse mit Java 8 erstellt wurde. Beim Start können Sie Ihren vorhandenen Arbeitsbereich aktualisieren, damit er mit Java 17 kompatibel ist. Alternativ können Sie einen vollständig neuen kompatiblen Arbeitsbereich erstellen. Sobald Sie Ihren Arbeitsbereich aktualisiert haben, können Sie ihn nicht mehr mit früheren Versionen von Studio verwenden.

11. Um sicherzustellen, dass Java 8 als Java-Laufzeitumgebung in Studio verfügbar ist, navigieren Sie zum Bereich "Systemeinstellungen : Fenstereinstellungen > - > Java > Installed JREs".

Wenn Java 8 fehlt, klicken Sie auf Hinzufügen und wählen Sie *Standard VM* als JRE-Typ. Geben Sie das Verzeichnis an, in dem Sie JDK 8 installiert haben.

12. Stellen Sie sicher, dass das Kontrollkästchen neben Java 8 JRE aktiviert ist.
13. Prüfen Sie die Ausführungsumgebungen auf eine optimale Übereinstimmung: Fenstereinstellungen > Java > Installed JREs > Execution > Environments

Wählen Sie JavaSE-1.8 im Bereich „Ausführungsumgebungen“ und dann die Umgebung, die als „perrechnung“ für die Umgebung beschrieben wird, die im Bereich „Compatible JREs“ als „Periodisch“ beschrieben wird.

14. Navigieren Sie im Bereich "Einstellungen" von Studio zu "Java > Compiler".
15. Wählen Sie unter JDK-Compliance im Dropdown-Menü Compiler-Compliance-Ebene die *Option 1,8*.
16. Klicken Sie auf Anwenden und schließen.

Anmerkung: Lesen Sie diese Übersicht über [Workday-Data Center](#), bevor Sie Studio konfigurieren. Die Data Center-Seite für Ihre Mandanten enthält Informationen, mit denen Ihre IT-Abteilung Firewalls konfigurieren und ordnungsgemäßen Studio-Zugriff auf Workday sicherstellen kann.

Workday Studio für macOS herunterladen und installieren

Warum und wann dieser Vorgang ausgeführt wird

Workday Studio basiert auf der Eclipse-Entwicklungsplattform. Es umfasst mehr als 150 Eclipse-Plugins sowie Abhängigkeiten von Plattformen und Drittanbietern. Sie benötigen zwei Versionen des Java™ Development Kit (JDK):

- Für Eclipse ist ein JDK erforderlich, das mit Java, Version 17, kompatibel ist.
- Für die Laufzeit von Studio ist ein JDK erforderlich, das mit Java Version 8 kompatibel ist.

Installieren Sie beide JDKs und stellen Sie Folgendes sicher:

- Ihre Java-Laufzeitumgebung ist auf Java 8 gesetzt.
- Für Ihre JDK-Compiler-Compliance ist Java 1.8 festgelegt.

Prozedur

1. Laden Sie ein JDK herunter und installieren Sie es, das mit Java Version 17 kompatibel ist. Sie können die kostenlose Open-Source-Implementierung OpenJDK aus verschiedenen Quellen herunterladen.

Beispiel:

- Wählen Sie [Zulu OpenJDK 17](#) für Ihr Betriebssystem.

Anmerkung:

- Benutzer von Apple Verwalten, müssen ein Apple Resource Management JDK17 verwenden. Ein x86_64-JDK funktioniert nicht.
- Für Studio ist ein JDK erforderlich, das speziell mit Java 17 kompatibel ist, nicht mindestens mit Java 17.

2. Laden Sie ein JDK herunter und installieren Sie es, das mit Java Version 8 kompatibel ist. Auch hier können Sie die kostenlose Open-Source-Implementierung OpenJDK wählen.

Beispiel:

- Wählen Sie [Zulu OpenJDK 8](#) für Ihr Betriebssystem

Anmerkung:

- Benutzer von Apple Verwalten, müssen ein Apple Resource Management JDK8 verwenden. Ein x86_64-JDK funktioniert nicht.
- Für Studio ist ein JDK erforderlich, das speziell mit Java 8 kompatibel ist, nicht mindestens mit Java 8.

3. Laden Sie das richtige Installationsprogramm für macOS Studio herunter:

- [Apple Ansprüche 1](#)
- [x86, 64 Bit](#)

4. Um die Installation zu starten, öffnen Sie die .jar-Installationsdatei.
5. Lesen Sie die Lizenzvereinbarung und aktivieren Sie dann das Kontrollkästchen zur Annahme der Bedingungen. Klicken Sie auf Weiter.
6. Navigieren Sie zum Speicherort Ihrer JDK 17-Installation und klicken Sie auf Next. Example: / Library/Java/JavaVirtualMachines/zulu-17.jdk/Contents/Home.
7. Wenn die JVM-Details korrekt sind, klicken Sie auf "Bestätigen". Andernfalls klicken Sie auf Zurück.
8. Wählen Sie das Paket Report Designer, um es zusammen mit Workday Studio zu installieren. Sie können den Report Designer auch später aus Studio installieren.
9. Navigieren Sie zu dem Verzeichnis, in dem Sie Studio installieren möchten. Klicken Sie auf Weiter.
10. Wenn Studio die Meldung „Installation abgeschlossen“ anzeigt, klicken Sie auf "Weiter" und dann auf "Fertig stellen".

Anmerkung: Eclipse mit Java 17 kann keinen Arbeitsbereich verwenden, der von Eclipse mit Java 8 erstellt wurde. Beim Start können Sie Ihren vorhandenen Arbeitsbereich aktualisieren, damit er mit Java 17 kompatibel ist. Alternativ können Sie einen vollständig neuen kompatiblen Arbeitsbereich erstellen. Sobald Sie Ihren Arbeitsbereich aktualisiert haben, können Sie ihn nicht mehr mit früheren Versionen von Studio verwenden.

11. Um sicherzustellen, dass Java 8 als Java-Laufzeitumgebung in Studio verfügbar ist, navigieren Sie zum Bereich "Systemeinstellungen" für Workday Studio > Preferences > Java > Installed JREs

Wenn Java 8 fehlt, klicken Sie auf Hinzufügen und wählen Sie *Standard VM* als JRE-Typ. Geben Sie das Verzeichnis an, in dem Sie JDK 8 installiert haben.

12. Stellen Sie sicher, dass das Kontrollkästchen neben Java 8 JRE aktiviert ist.

13. Prüfen Sie die Ausführungsumgebungen auf eine optimale Übereinstimmung: Workday Studio > Preferences > Java > Installed JREs > Execution Environments

Wählen Sie JavaSE-1.8 im Bereich „Ausführungsumgebungen“ und dann die Umgebung, die als „perrechnung“ für die Umgebung beschrieben wird, die im Bereich „Compatible JREs“ als „Periodisch“ beschrieben wird.

14. Navigieren Sie im Bereich "Einstellungen" von Studio zu "Java > Compiler".

15. Wählen Sie unter JDK-Compliance im Dropdown-Menü Compiler-Compliance-Ebene die *Option 1,8*.

16. Klicken Sie auf Anwenden und schließen.

Anmerkung: Lesen Sie diese Übersicht über [Workday-Data Center](#), bevor Sie Studio konfigurieren. Die Data Center-Seite für Ihre Mandanten enthält Informationen, mit denen Ihre IT-Abteilung Firewalls konfigurieren und ordnungsgemäßen Studio-Zugriff auf Workday sicherstellen kann.

Konzept: Workday Studio

Workday bietet zwei Möglichkeiten, Integrationen durchzuführen:

- Konnektoren: Integrationspakete, die eine Verbindung zu einer Vielzahl gängiger Services von Drittanbietern herstellen.
- Enterprise Interface Builder (EIB): Ein einfaches Tool zum Erstellen von Integrationen ohne Programmieraufwand.

Workday Studio ist eine dritte Option, mit der Sie wesentlich komplexere Integrationen erstellen, selbst verwalten und unterstützen können, die Workday in Ihrem Auftrag hostet und ausführt. Im Gegensatz zu Konnektoren und EIB ist Studio nicht Teil von Workday. Es handelt sich um eine Eclipse-basierte integrierte Entwicklungsumgebung (Integrated Development Environment, IDE), die Sie separat herunterladen können. Wie bei allen anderen Integrationen in Ihrem Mandanten können Sie die Integrationen, die Sie mit Studio erstellen, starten, planen und auditieren. Sie können von Studio erstellte Integrationen in Studio selbst oder in der Workday-Anwendung starten.

Der Zielbenutzer für Studio:

- Verfügt über Kenntnisse in der Entwicklung.
- Ist mit Konzepten der Integrationsprogrammierung vertraut.
- Ist mit der Verwendung einer IDE vertraut.

Die Studio-Plattform besteht aus vier Hauptkomponenten:

- Die Studio-IDE
- Integrationen, Sammlungen und das Cloud Repository
- Die Workday-Anwendung
- Cloud-Laufzeiten

Die Studio-IDE

Studio ist eine Eclipse-basierte IDE. Sie verwenden es, um Integrationsabläufe zu erstellen, die:

- Auf Informationen aus Workday zugreifen.
- Es bearbeiten.
- Es (optional) irgendwo hinsenden.

Diese Abläufe werden als Assemblys bezeichnet. Sie erstellen Assemblys, indem Sie im Assembly-Editor von Studio grafische Komponenten und Schritte hinzufügen und miteinander verbinden.

Studio enthält Komponenten und Schritte für viele Aktionen, einschließlich Splitten, Transformieren und Routing. Workday-spezifische Komponenten haben eine integrierte Beschreibung dafür, wie Workday Informationen darstellt.

Sie können Ihre Integration in Studio testen, bevor Sie sie direkt in Ihrem Workday-Mandanten bereitstellen.

Studio unterstützt die Verwendung von Java-Erweiterungen in Assembly-Projekten. Sie können Ihren eigenen Java-Code bereitstellen und ihn als Spring Bean konfigurieren. Darüber hinaus können Sie Java-Bibliotheken von Drittanbietern in Ihre Integrationen einbinden.

Integrationen, Sammlungen und das Cloud Repository.

Das Cloud Repository ist der Begriff von Workday für seinen Studio-Integrationsspeicher. Vor der Bereitstellung weist Studio jede Integration einer Sammlung zu. Eine Sammlung kann mehrere Integrationen enthalten. Eine Integration kann zu mehr als einer Sammlung gehören. Das bedeutet, dass Sie Assemblys und Subassemblys problemlos für Integrationen und Mandanten wiederverwenden können.

Es kann immer nur eine Instanz einer bestimmten Sammlung ausgeführt werden. Wenn Sie ein Assembly-Projekt innerhalb einer Sammlung ändern und das Projekt erneut bereitstellen, ersetzt es alle zuvor bereitgestellten Versionen.

Die Binärdatei, die Studio für das Workday Cloud Repository bereitstellt, wird als CLAR (CLoud ARchive) bezeichnet. Eine CLAR enthält Folgendes:

- Alle Artefakte der enthaltenen Integrationsprojekten, z. B. deren Dateien „assembly.xml“, „XSLT“ und „Textschema“.
- Alle kompilierten Java-Codes oder Java-Bibliotheken von Drittanbietern, die Entwickler zu den Integrationsprojekten hinzugefügt haben.

Als binäre Objekte enthalten CLARs keine Quellcode-Artefakte wie Java-Quelldateien.

Das Cloud Repository speichert CLARs in einem verschlüsselten Format. Speichern Sie Informationen wie Kennwörter jedoch nicht direkt in Ihren Integrationen. Konfigurieren Sie sensible Details stattdessen in der Workday-Anwendung, damit Sie diese dynamisch verwalten können.

Die Workday-Anwendung

Workday verwendet eine Abstraktion, die als Integrationssystem bezeichnet wird, um Integrationen zu modellieren. In Workday können Sie für jede Integration Folgendes angeben:

- Welche Benutzer die Berechtigung zum Starten der Integration haben und welche Rechte die Integration während der Ausführung hat.
- Wie und wann verschiedene Workday-Benutzer benachrichtigt werden, wenn die Integration ausgeführt wird.

Da Sie Ihre Studio-Integrationen separat entwerfen und konfigurieren, können Sie eine Integration erneut bereitstellen, ohne dass sich dies auf ihre Konfiguration auswirkt.

Mit Workday können Integrationen in einer Reihe verschiedener Szenarien gestartet werden. Sie können:

- Integrationen manuell starten.
- Integrationen so planen, dass sie automatisch als Ereignis ausgeführt werden.
- Eine Integration automatisch als Teil eines Geschäftsprozesses starten.
- Den Start remote über einen Webservice-Client auslösen, um einen Workday-Webservice aufzurufen.

Sie können den Workday-Bericht Prozessmonitor verwenden, um die Ausführung einer Integration zu überwachen.

Cloud-Laufzeiten

Wenn eine Integration gestartet wird, führt Workday folgende Schritte aus:

- Ein Integrationsereignis wird generiert.
- Die Integration wird aus dem Cloud Repository abgerufen.

- Ein geschützter Satz von Verarbeitungsressourcen wird zugewiesen, der als Cloud-Laufzeit bezeichnet wird.
- Führt die Integration zur Cloud-Laufzeit aus.

Wenn Workday eine Integration nicht ausführen kann, wird der Prozessmonitor mit dem Startstatus der Integration aktualisiert `Failed`.

Wenn Workday während der Ausführung ein Problem feststellt, z. B. die Nichtverfügbarkeit einer externen Ressource, wird der Prozessmonitor beendet und der Prozessmonitor mit dem Integrationsstartstatus "" aktualisiert `Completed with errors`.

Anmerkung: Es liegt in der Verantwortung des Integrationsentwicklers, sich mit Integrationen zu befassen, die als markiert sind `Completed with errors`.

Workday beendet Integrationen, deren Ausführung länger als 2 Stunden dauert. Ein vom Entwickler bereitgestellter Java-Code kann im Allgemeinen nicht aus dem Dateisystem oder den Java-Systemeigenschaften lesen oder schreiben. Ein vom Entwickler bereitgestellter Java-Code kann keine neuen Java-Threads generieren.

Sicherheit und Datenschutz

Studio unterstützt automatisch zwei Cloud-Umgebungen:

- Sandbox
- Produktion

Sie können Verbindungen zu anderen Workday-Umgebungen hinzufügen. Es liegt in Ihrer alleinigen Verantwortung, sicherzustellen, dass Sie die richtigen Integrationen in der entsprechenden Umgebung bereitstellen. Die Standard-Sicherheitsrichtlinien für Workday Webservices gelten für alle Daten, auf die über Studio-Integrationen zugegriffen wird.

Sie können Assemblys offline in Studio erstellen, aber Sie müssen über die entsprechende Sicherheitskonfiguration verfügen, um sie in Workday zu integrieren. Nach der Bereitstellung werden Integrationen mit den Zugangsdaten und Berechtigungen des Benutzers, der sie ausführt, gestartet und ausgeführt.

Workday hat keine Einsicht in das Aussehen und die Funktionsweise Ihrer Integration. Es liegt in Ihrer Verantwortung, sicherzustellen, dass sich die Integration wie erwartet verhält. Die Interaktionen Ihrer Integration mit Ihrem Workday-Mandanten erfolgen über die öffentlichen APIs von Workday, sodass Sie alle zugehörigen Stabilitäts- und Sicherheitsgarantien übernehmen.

Konzept: Lebenszyklus der Integrationsentwicklung

Sie entwickeln und überwachen Studio-Integrationen in den folgenden verschiedenen Phasen, indem Sie je nach Bedarf entweder Workday Studio oder die Workday-Anwendung verwenden:

Phase	Beschreibung
Integrationsdesign	Entwickeln Sie in Studio die Assemblys, die Ihre Integration bilden. Optional können Sie Assembly-Module erstellen, mit denen Sie Integrationsmuster und -konfigurationen wiederverwenden und teilen können. Jede Assembly hat mindestens einen <code>workday-in-Transport</code>, der angibt, wo die Ausführung beginnt. Jeder <code>workday-in-Transport</code> kann optional Folgendes definieren: • Startparameter, die es einem Client ermöglichen, Informationen als Eingabe

Phase	Beschreibung
	an die Integration zu übergeben. Beispiel: Eine Entgeltabrechnungsintegration kann angeben, dass zwei Startparameter erforderlich sind: <code>payPeriod</code> und <code>payGroup</code>. Dadurch können die Benutzer der Integration den Monat und die Mitarbeitergruppe angeben, die bei der Berechnung der Entgeltabrechnung verwendet werden sollen. • Workday Services, die die Assembly zur Laufzeit verwendet. Workday bietet generische Integrationsservices für Attribut-Mapping, Sequenzgenerierung, Transaktionsprotokollierung, Dokumentabruf und -übermittlung. Außerdem werden ein Listener-Service, ein Berichtsservice und ein Service für benutzerdefinierte Objekte bereitgestellt.
Integrations-Deployment	Stellen Sie in Studio eine Verbindung zum Workday Cloud Repository her, um Ihre Integration bereitzustellen. Für jeden workday-in-Transport, der in einer Integration definiert ist, registriert Workday eine Abstraktion, die als Integrationssystem bezeichnet wird. Ein Integrationssystem umfasst alle erforderlichen Startparameter und Services.
Integrationskonfiguration	In Workday können Sie bestimmte Aspekte Ihrer Integration konfigurieren, einschließlich Berechtigungen und Benachrichtigungen.
Integration wird gestartet	Mit dem Integrationssystem kann Workday Ihre Integration starten und verfolgen. Um die Entwicklung und das Testen zu vereinfachen, können Sie den Start auch in Studio durchführen.
Integrationsmonitoring und Problembehebung	Verfolgen Sie den Fortschritt Ihrer Integration in Studio oder Workday. In Studio können Sie die Ausführung der Integration im Prozessmonitor überwachen. Sie können auch einen konsolidierten Bericht über den Integrationsprozess aufrufen, nachdem dieser abgeschlossen ist.

Konzept: Objekt „MediationContext“

Während der Batch-Ausführung erstellt Workday ein temporäres Inhaberobjekt mit dem Namen `MediationContext`. Die `MediationContext` enthält 3 Hauptarten von Informationen:

- `MediationMessage`: Ein Objekt, das die Text- oder Binärdatennachricht enthält, die Studio endgültig über einen Out-Transport an ein externes Ziel sendet. Beispiele: HTTP-Out, FTP-Out.
- Variablen: temporäre Inhaber für Nachrichtendaten.
- Eigenschaften: Eine Karte mit Name-Objekt-Paaren, die Build-Elemente zum Speichern vorübergehender Daten außerhalb einer Meldung verwenden.

Jede Verarbeitung in Builds umfasst entweder das Lesen von oder das Schreiben in `MediationContext`.

Anmerkung: Wenn Sie auf zugreifen `MediationContext` -Objekt unter Verwendung von MVE, der Accessor ist `context`. Beispiel: `context.getErrorMessage()`. In Java ist es `mediationContext`. Beispiel: `mediationContext.getErrorMessage()`.

Konzept: Integrationssysteme und Ereignisse

Integrationssysteme

In Workday ist ein Integrationssystem eine Abstraktion einer Integration mit einem externen System oder einem Workday-Endpunkt. Integrationssysteme stellen das Framework bereit, nach dem Workday Integrationen startet und verfolgt. Mit einem Integrationssystem können Sie Folgendes konfigurieren:

- Sicherheit: Geben Sie an, wer die Integration starten kann und welche Rechte die Integration bei der Ausführung hat.
- Benachrichtigungen: Geben Sie an, wer bei der Ausführung einer Integration benachrichtigt werden soll.
- Parameter: Sie können den Aufruf einer Integration parametrisieren. Beispiel: Sie können die Endpunkte und Benutzerzugangsdaten konfigurieren, die eine Integration benötigt, um Daten auf eine externe SFTP-Site hochzuladen.
- Trigger: Sie können die Integration so konfigurieren, dass sie wie folgt getriggert wird:
 - Manuell
 - Im Rahmen eines Abonnements.
 - Mit einer API.
 - Nach einem Zeitplan.
 - Als Teil eines Geschäftsprozesses.

Workday stellt alle Integrationen, unabhängig von ihrem Ursprung, als Integrationssysteme dar.

Wenn Sie eine Integration mit Workday Studio entwickeln und in Workday bereitstellen, registrieren Sie automatisch ein neues Integrationssystem für jeden workday-in-Transport in dieser Integration.

Integrationsereignis

Wenn eine Integration gestartet wird, erstellt Workday ein Objekt, das als Integrationsereignis bezeichnet wird. Dieses Objekt ist für jeden Aufruf eindeutig und enthält eine ID, die als Integrationsereignis-ID bezeichnet wird. Workday verwendet diese eindeutige ID, um alle ausgeführten Integrationen zu verfolgen. Sie können Integrationsereignisse mit dem Prozessmonitor in Workday anzeigen.

Workday erstellt nicht nur ein Integrationsereignisobjekt, sondern auch eine Anforderungsnachricht für das Integrationsereignis, und sendet diese an die Integration. Diese Anforderung enthält Informationen zum Integrationssystem und zur Integrationsereignis-ID. Sie enthält außerdem Werte für alle Parameter, die zum Startzeitpunkt bereitgestellt werden. Die Integration kann diese Informationen dann verwenden, wenn sie Informationen von Workday-Webservices anfordert. Die Integration kann diese Informationen auch verwenden, um Aktualisierungsinformationen an Workday zurückzusenden. Beispiel: eine Statusmeldung, die dem Benutzer mitteilt, dass 100 Mitarbeiterdatensätze erfolgreich verarbeitet wurden. Workday fügt diese Aktualisierungen zum Integrationsereignisobjekt hinzu.

Referenz: Grundlagen der Benutzeroberfläche

Die Oberfläche von Workday Studio besteht aus verschiedenen Ansichten. Um Ansichten ein- oder auszublenden, wählen Sie **Fenster > Ansicht anzeigen > Andere ...** und blenden Sie dann den Ordner Workday ein.

Die am häufigsten verwendeten Ansichten in Studio sind:

Anzeigen	Beschreibung
Project Explorer	Der Projektnavigator von Studio. Zeigt Projekte an, die im aktuellen Arbeitsbereich gespeichert wurden. Jeder Ordner enthält alle relevanten Dateien für dieses Projekt und verweist auf die Einzüge, zu denen das Projekt gehört.
Cloud Explorer	Zeigt die Integrations-Cloud-Server an, auf denen Studio bereitstellen kann, sowie die aktuell bereitgestellten Einzüge. Sie können Verbindungen zu mehreren Umgebungen hinzufügen. Wenn Verbindungs-, Authentifizierungs- oder Autorisierungsfehler auftreten, zeigt Studio einen Fehler oder eine Warnung als Decorator auf dem Verbindungssymbol an.
Palette	Zeigt alle Assembly-Komponenten an, die Sie hinzufügen können. Verfügt über drei Unteransichten mit Registern: • Design: das Assembly-Diagramm wird angezeigt. • Quelle: die XML-Quelle wird angezeigt. • Message Builder: eine grafische Ansicht, die die Definition des Inhalts von Nachrichten für Assemblys, wie etwa den Schritt Schreiben, vereinfacht.
Assembly-Editor	Eine schematische Darstellung der Assembly. Um neue Komponenten und Schritte hinzuzufügen, ziehen Sie diese per Drag-and-Drop aus der Palette. Um Komponenten zu verbinden, bewegen Sie den Mauszeiger über eine Quellkomponente, um die Ziehpunkte anzuzeigen, und ziehen Sie den Ziehpunkt dann per Drag-and-Drop auf die Zielkomponente.
Eigenschaften	Konfigurationsdetails für das aktuell im Assembly-Diagramm ausgewählte Assembly-Element werden angezeigt. Bearbeiten Sie die Eigenschaften und Parameter mithilfe der Register, die auf der linken Seite der Ansicht angezeigt werden.
Prozessmonitor	Informationen über den Assembly-Prozess werden angezeigt, wenn Sie eine Integration starten. Wählen Sie einen Prozess aus, um seine Details in der Ansicht Prozessdetails anzuzeigen.
Schema Explorer	Ermöglicht das Erkunden des Inhalts von WSDL- und XML-Schema-Dateien und bietet verschiedene Dienstprogrammfeatures für die Entwicklung von Assemblys, die mit Webservices interagieren.
Betrachter für konsolidierte Berichte	Ermöglicht die Analyse von Verarbeitungsschritten für Assemblys in einer Integration.

Integrationsdesign

Assemblys

Assembly-Projekt erstellen

Warum und wann dieser Vorgang ausgeführt wird

Zusammenstellungen definieren nachrichtenbasierte Integrationsprozesse, die aus Anmerkungen, Transporten, Vermittlungskomponenten, Vermittlungsschritten und Fehlerhandlern bestehen.

Prozedur

1. Wählen Sie Datei > Neues > Workday-Zusammenfassungsprojekt.
2. Geben Sie im Feld Projektname einen Namen für Ihr Projekt ein.
Projektname dürfen keine Leerzeichen enthalten.
3. Wählen Sie in der Liste Ziel-Laufzeit die Option *Workday Laufzeit*.
4. Wählen Sie in der Liste Konfigurationen eine der folgenden Konfigurationsoptionen:

Option	Bezeichnung
Assembly	Erstellt ein einfaches Assembly-Projekt.
Assembly (mit Java-Unterstützung)	Erstellt ein Projekt, das eine Java-Facette enthält.
Benutzerdefiniert	Ermöglicht die Festlegung der Projektfacetten für Ihr Assembly-Projekt.

5. Klicken Sie auf Weiter. Geben Sie einen Sammlungsname ein.
Das Feld Sammlungsname wird automatisch mit Ihrem Projektname ausgefüllt.
6. Klicken Sie auf Fertig stellen , um Ihr Projekt zu erstellen.

Anmerkung: Wenn Sie ein Baugruppenprojekt außerhalb Ihres Arbeitsbereichs erstellen, können Sie ihm in Studio einen neuen Namen geben, ohne den Ordernamen am ursprünglichen Speicherort zu ändern. Dies gilt auch für Projekte, die Sie von außerhalb Ihres Arbeitsbereichs importieren. Wählen Sie dazu Datei > umbenennen.

Ergebnisse

Ihr neues Baugruppenprojekt wird in der Ansicht „Projekt-Explorer“ angezeigt. Sie enthält eine Datei „assembly.xml“, die Sie nun mit dem Assembly-Editor öffnen können.

Nächste Maßnahme

Mit dem Assembly-Editor können Sie Ihre Integration definieren, indem Sie Transporte, Mediationskomponenten und Mediationsschritte hinzufügen und verknüpfen.

Schritte zu Komponenten hinzufügen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Schritte führen einzelne Vorgänge für eine Nachricht aus, während sie eine Integration durchläuft. Sie können nur den Komponenten async-mediation und sync-mediation Schritte hinzufügen.

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Fügen Sie der Assembly eine async-mediation oder sync-mediation Komponente hinzu.
3. Ziehen Sie per Drag-and-Drop einen Schritt aus der Palette in Ihre async-mediation oder sync-mediation Komponente. Ordnen Sie die Schritte in der Reihenfolge ihrer Ausführung an. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Palettenkategorie	Beschreibung
Transformieren	Zeigt Mediationsschritte an, mit denen Sie Nachrichten transformieren können, während sie Ihre Integration durchlaufen.
Protokollierung und Ereignisverarbeitung	Zeigt Mediationsschritte an, mit denen Sie Nachrichten protokollieren können, während sie Ihre Integration durchlaufen.
Nachrichtенbearbeitung	Zeigt Mediationsschritte an, mit denen Sie Nachrichten bearbeiten können, während sie Ihre Integration durchlaufen.
Validierung	Zeigt Mediationsschritte an, mit denen Sie Nachrichten validieren können, während sie Ihre Integration durchlaufen.
Codierung	Zeigt Mediationsschritte an, mit denen Sie Nachrichten codieren können, während sie Ihre Integration durchlaufen.
Sonstige	Zeigt verschiedene Mediationsschritte an, mit denen Sie einzelne Vorgänge für Nachrichten durchführen können, während diese Ihre Integration durchlaufen.

4. Klicken Sie auf Speichern.

Ergebnisse

Studio zeigt den neuen Schritt innerhalb der Komponente im Assembly-Diagramm an.

Nächste Maßnahme

Konfigurieren Sie die Eigenschaften des neuen Schritts.

Untergeordnete Elemente hinzufügen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Einige Assembly-Elemente unterstützen untergeordnete Elemente. Untergeordnete Elemente sind optional.

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Fügen Sie ein Assembly-Element zum Assembly-Diagramm hinzu, das untergeordnete Elemente unterstützt.
3. Bewegen Sie den Mauszeiger über den Titel des Assembly-Elements, um die Optionen für das untergeordnete Element anzuzeigen.
4. Klicken Sie auf eine Option für ein untergeordnetes Element, um es zu Ihrem Assembly-Element hinzuzufügen.

Ergebnisse

Studio zeigt das untergeordnete Element auf Ihrem Assembly-Element an.

Nächste Maßnahme

Konfigurieren Sie die Eigenschaften des untergeordneten Elements.

Assembly-Elemente verbinden**Vorbereitungen**

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Verbindungen zwischen Assembly-Elementen definieren den Pfad einer Nachricht durch Ihre Assembly. Sie können Assembly-Elemente mit dem Ziehpunkt der ausgehenden Verbindung (blaues Quadrat) oder dem Ziehpunkt der eingehenden Verbindung (orangefarbenes Quadrat) im Assembly-Editor verbinden.

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Fügen Sie zwei Assembly-Elemente zum Assembly-Diagramm hinzu.
3. Platzieren Sie den Mauszeiger über dem Titel des ersten Assembly-Elements, um dessen Verbindungsziehpunkte anzuzeigen. Ziehen Sie den Ziehpunkt der ausgehenden Verbindung (blaues Quadrat) vom ersten Assembly-Element zum zweiten Assembly-Element.
4. Alternativ können Sie den Mauszeiger über dem Titel des zweiten Assembly-Elements platzieren und den Ziehpunkt der eingehenden Verbindung (orangefarbenes Quadrat) zum ersten Assembly-Element ziehen.

Ergebnisse

Die Nachrichtenausgabe des ersten Assembly-Elements wird zur Eingabe für das zweite Assembly-Element.

Eigenschaften der Assembly-Elemente konfigurieren**Vorbereitungen**

Fügen Sie einem Assembly-Diagramm Assembly-Elemente hinzu.

Warum und wann dieser Vorgang ausgeführt wird

Jedes Element in der Workday Studio-Palette verfügt über eine Ansicht Eigenschaften, in der Sie seine Funktionen konfigurieren können.

Prozedur

1. Klicken Sie mit der rechten Maustaste auf ein Assembly-Element und wählen Sie Ansicht „Eigenschaften“ anzeigen.
2. Erforderliche Eigenschaften sind mit einem Sternchen oben links gekennzeichnet. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Eigenschaftssymbol	Eigenschaftsbeschreibung
	Werte, die innerhalb der Assembly eindeutig sein müssen.
	Eingabe in das Assembly-Element.
	Ausgabe aus dem Assembly-Element.
	MVEL-Ausdrücke.
	MVEL-Vorlagen.
	Boolesche Werte.
	Optionen.
	Numerische Werte.
	Attributwerte.
	Implementierungen von Java-Klassen.
	Spring-Beans, die Implementierungen von Java-Klassen referenzieren.

Anmerkung: Wenn Eigenschaften MVEL-fähig sind, wird das relevante MVEL-Symbol zum eigenen

Symbol der Eigenschaft hinzugefügt. Beispiel: 

3. So vergleichen Sie Eigenschaftsansichten:
 - a) Wählen Sie eine Komponente im Assembly-Diagramm aus.
 - b) Halten Sie die Strg-Taste gedrückt und wählen Sie eine zweite Komponente aus.
 - c) Klicken Sie mit der rechten Maustaste auf eine der beiden Komponenten und wählen Sie Eigenschaften vergleichen. Die Eigenschaftenansicht der ersten Komponente wird links angezeigt. Die Eigenschaftenansicht der zweiten Komponente wird rechts angezeigt und ist an diese Auswahl angeheftet. Die Ansicht auf der linken Seite ändert sich, wenn Sie eine andere Komponente auswählen.
4. Drücken Sie in MVEL-fähigen Eigenschaftsfeldern Strg+Leertaste, um die Inhaltshilfe zu öffnen, eine Reihe von Optionen für die Vervollständigung von kontextabhängigem Code.

Anmerkung: Die Inhaltshilfe ist nur in Assembly-Projekten verfügbar, die die Java-Facette enthalten.

Sammlungen und Projekte verwalten

Vorbereitungen

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

In Workday Studio können Sie mit Sammlungen mehrere Projekte paketieren und bereitstellen. Ein Projekt gehört immer zu mindestens einer Sammlung und kann gleichzeitig Mitglied mehrerer Sammlungen sein.

Prozedur

1. Klicken Sie auf Workday > Einzug verwalten.

Wenn Sie eine Sammlung im linken Fensterbereich einblenden, zeigt Studio die darin enthaltenen Projekte an. Projekte, die nicht in dieser Sammlung enthalten sind, werden im rechten Fensterbereich angezeigt.

2. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Schaltfläche	Beschreibung
Neue Sammlung	Fügt eine neue Sammlung zu Ihrem Arbeitsbereich hinzu.
Sammlung umbenennen	Benennt die ausgewählte Sammlung um.
Sammlung entfernen	Die ausgewählte Sammlung wird aus Ihrem Arbeitsbereich entfernt.
Zu Sammlung hinzufügen	Fügt das ausgewählte Projekt zu einer Sammlung hinzu.
Alle Integrationen zur Sammlung hinzufügen	Fügt alle Projekte zur ausgewählten Sammlung hinzu.
Die Integration aus der Sammlung entfernen	Entfernt das ausgewählte Projekt aus einer Sammlung.

Anmerkung: Stellen Sie sicher, dass alle Projekte in einer Sammlung dieselbe Assembly-Version haben. Andernfalls zeigt Studio eine Fehlermarkierung an. Sie können den Wert der Eigenschaft Version für die Assembly in der Ansicht Eigenschaften im Assembly-Editor konfigurieren.

3. Klicken Sie auf Beenden.

Namespace-Präfixe deklarieren

Vorbereitungen

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

In Workday Studio sind Namespace-Präfixe Abkürzungen mit vordefinierten URI-Werten. Sie können Namespace-Präfixe deklarieren, um auf bestimmte URI-Werte in Ihren Assemblys zu verweisen.

Prozedur

1. Öffnen Sie eine Assembly im Assembly-Editor.
2. Klicken Sie mit der rechten Maustaste auf den Hintergrund des Assembly-Diagramms und wählen Sie Show Properties View.

3. Klicken Sie auf das Register Namespace Prefixes.
4. Klicken Sie auf Hinzufügen, um ein neues Namespace-Präfix hinzuzufügen. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Präfix	Geben Sie hier einen Präfixnamen ein.
Namespace	Geben Sie hier die Namespace-URL ein.

5. Um Ihr neues Namespace-Präfix zu bearbeiten, klicken Sie auf Bearbeiten.
6. Um Ihr neues Namespace-Präfix zu entfernen, klicken Sie auf Entfernen.

Ergebnisse

Das Namespace-Präfix für die Assembly wird im XML-Schema angezeigt.

Pfadvariablen erstellen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Pfadvariablen geben den Speicherort der Ressourcendateien in Ihrem Arbeitsbereichsordner an. Sie können Arbeitsbereichspfadvariablen verwenden, wenn Sie Plattformressourcen in Ihrer Projekt-Build verknüpfen.

Prozedur

1. Wählen Sie Windows > Einstellungen > Allgemeiner > Arbeitsbereich > Verknüpfte Ressourcen.
2. Klicken Sie auf "Neu".

Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Name	Geben Sie den Namen Ihrer Pfadvariablen ein.
Standort	Geben Sie den Dateispeicherort Ihrer Pfadvariablen ein.

3. Klicken Sie auf OK.

Ergebnisse

Ihre neue Pfadvariable gibt jetzt den Speicherort der Ressourcendateien in Ihrem Arbeitsbereichsordner an.

Konzept: Assembly-Palette

Die Palette zeigt alle in Studio verfügbaren Assembly-Elemente an. Um ein Assembly-Element zu einem Assembly-Diagramm hinzuzufügen, ziehen Sie es aus der Palette in den Assembly-Editor. Die Palette listet Assembly-Elemente in den folgenden Kategorien auf:

Palettenkategorie	Beschreibung
Favoriten	Zeigt Ihre bevorzugten Assembly-Elemente an. Sie können dieser Kategorie Assembly-Elemente hinzufügen, indem Sie auf das leere Sternsymbol neben ihnen klicken. Sie können Elemente aus

Palettenkategorie	Beschreibung
	dieser Kategorie entfernen, indem Sie auf das hervorgehobene Sternsymbol neben ihnen klicken.
Vermerk	Zeigt Anmerkungen an, mit denen Sie Ihr Assembly-Diagramm strukturieren, organisieren und mit Labels versehen können.
Modulbibliothek	Zeigt Assembly-Elemente in Modulform an, mit denen Sie Workday-Webservice-Aufrufe und globale Fehlerhandler für Ihre Integration konfigurieren können.
Eingehende Transporte	Zeigt Assembly-Elemente für eingehende Transporte an, die Nachrichten von Workday oder einer lokalen Ressource erhalten.
Ausgehende Transporte	Zeigt Assembly-Elemente für ausgehende Transporte an, die Nachrichten an eine Ressource senden.
Komponenten	Zeigt Mediationskomponenten an, mit denen Sie Nachrichten bearbeiten und umwandeln können, während sie Ihre Integration durchlaufen.
Allgemeine Komponenten	Zeigt vorkonfigurierte Subassemblies an, mit denen Sie Ihre Integrationen erstellen können.
Komponenten des Arbeitsbereichs	Zeigt benutzerdefinierte Subassemblies an, die Sie erstellt und in Ihrem Arbeitsbereich gespeichert haben.
Fehlerhandler	Zeigt Fehlerhandler an, mit denen Sie Fehler in Ihren Integrationen abfangen können.
Transformieren	Zeigt Mediationsschritte an, mit denen Sie Nachrichten transformieren können, während sie Ihre Integration durchlaufen.
Protokollierung und Ereignisverarbeitung	Zeigt Mediationsschritte an, mit denen Sie Nachrichten protokollieren können, während sie Ihre Integration durchlaufen.
Nachrichtенbearbeitung	Zeigt Mediationsschritte an, mit denen Sie Nachrichten bearbeiten können, während sie Ihre Integration durchlaufen.
Validierung	Zeigt Mediationsschritte an, mit denen Sie Nachrichten validieren können, während sie Ihre Integration durchlaufen.
Codierung	Zeigt Mediationsschritte an, mit denen Sie Nachrichten codieren können, während sie Ihre Integration durchlaufen.
Sonstige	Zeigt Mediationsschritte an, mit denen Sie einzelne Vorgänge für Nachrichten durchführen können, während diese Ihre Integration durchlaufen.

Palettenkategorie	Beschreibung
Kontakte	Zeigt die Verbindungen Weiterleiten an und Antwort weiterleiten an an, mit denen Sie den Pfad einer Nachricht durch Ihre Integration definieren können.

Um die in der Palette angezeigten Assembly-Elemente manuell zu filtern, verwenden Sie das Feld Typenfiltertext. Studio stellt außerdem die folgenden vorkonfigurierten Kategoriefilter direkt unter dem Feld Typenfiltertext bereit:

Kategoriefilter	Beschreibung
Wiederherstellen	Löscht die vorherige Filterauswahl.
Transporte	Zeigt nur Paletten-Kategorien für Transporte an.
Komponenten	Zeigt nur Paletten-Kategorien für Mediationskomponenten an.
Schritte	Zeigt nur Paletten-Kategorien für Mediationsschritte an.
Handler und Verbindungen	Zeigt nur Paletten-Kategorien für Fehlerhandler und Verbindungen an.
Kategorien filtern	Zeigt das Fenster Filter für Palettenkategorie an.
Alle einblenden	Blendet alle Paletten-Kategorien ein.
Alle ausblenden	Blendet alle Paletten-Kategorien aus.

Studio zeigt die Palette automatisch neben dem Assembly-Editor an. Um die Palette in einer separaten Ansicht anzuzeigen, wählen Sie Fenster > Ansicht anzeigen > Andere > Allgemeine > Palette und klicken Sie auf OK. Sie können das Layout der Palette auch ändern, indem Sie mit der rechten Maustaste darauf klicken und Layoutwählen.

relatedlinks

Referenz

[Workday 31 What's New Post: Workday Studio](#)

Konzept: Komponenten und Schritte

Sie können nur den Komponenten async-mediation und sync-mediation Schritte hinzufügen.

Workday Studio bietet Schritte zur Unterstützung von Vorgängen wie Transformation, Validierung und Protokollierung an. Sie können auch benutzerdefinierte Schritte mit Spring-Beans schreiben.

Die Komponente async-mediation unterstützt eine Verarbeitungskette, die nur Anforderungsnachrichten verarbeitet. Die Komponente sync-mediation unterstützt zwei Verarbeitungsketten, die Anforderungsnachrichten und Antwortnachrichten auf Anforderungen verarbeiten.

Schritte haben die Eigenschaften Eingabe und Ausgabe, die angeben, woher sie Daten als Eingabe erhalten und wohin sie die Ausgabe leiten. Es gibt fünf mögliche Werte:

Option	Beschreibung
<i>message</i>	Die gesamte Nachricht, einschließlich Root-Teil und Anhängen. Wenn Sie diese Option angeben, erstellt Studio eine neue Nachricht und entfernt effektiv alle Anhänge.

Option	Beschreibung
<i>soapbody</i>	Der Inhalt des SOAP-Texts aus dem Root-Teil der Nachricht.
<i>attachment</i>	Ein MIME-Anhang an die Nachricht. Geben Sie den Indexwert des Anhangs an, auf den Sie zugreifen möchten. Sie können der Nachricht auch einen neuen Anhang hinzufügen, indem Sie einen Indexwert von -1 angeben.
<i>rootpart</i>	Der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn die Nachricht keine MIME-Nachricht ist, die Nachricht selbst. Wenn dieser Wert für das Ausgabeattribut angegeben wird, werden alle Anhänge in der ursprünglichen Nachricht beibehalten.
<i>Variable</i>	Eine Variable aus <code>MediationContext</code> . Geben Sie den Namen der zu verwendenden Variable an.

Konzept: Anmerkungen zur Assembly

Mit Anmerkungen können Sie Ihre Assembly-Diagramme strukturieren, organisieren und mit Labels versehen. Studio bietet im Abschnitt Anmerkungen der Palette die folgenden Optionen:

- Rechteck
- Ellipse
- Anmerkung
- Text
- Swimlane

Sie können:

- Verwenden Sie die Anmerkungen Rechteck, Ellipse, Anmerkung und Text, um Ihrer Assembly allgemeinen erklärenden Text hinzuzufügen.
- Verwenden Sie die Schaltfläche Anmerkungen ein-/ausblenden, um die Anmerkungen Rechteck, Ellipse, Anmerkung und Text ein- oder auszublenden. Sie können neue Anmerkungen zu Assembly-Diagrammen nur hinzufügen, wenn Anmerkungen anzeigen aktiv ist.
- Verwenden Sie das Anstecksymbol für die Anmerkungen Rechteck, Ellipse, Anmerkung und Text, um sie an Ihre Assembly zu heften. Angepinnte Anmerkungen werden immer in Ihren Diagrammen angezeigt, auch wenn Anmerkungen ausblenden aktiviert ist.

Swimlanes

Mit Swimlanes können Sie Komponenten der Assemblys in Gruppen organisieren.

Sie können:

- Blenden Sie die Beschreibungen der Swimlane ein oder aus, indem Sie auf die Schaltfläche Swimlane-Beschreibungen ein-/ausblenden klicken.
- Blenden Sie alle Swimlanes ein und aus, indem Sie die Schaltflächen Alle Swimlanes einblenden und Alle Swimlanes ausblenden anklicken.
- Blenden Sie einzelne Swimlanes ein und aus, indem Sie auf die Plus- und Minus-Symbole in deren Headern klicken. Der Swimlane-Status ist dauerhaft. Wenn Sie eine Assembly öffnen, werden die Swimlanes in dem Status angezeigt, in dem sie beim Speichern der Assembly ein- bzw. ausgeblendet waren.

Swimlanes sind verschachtelbar.

relatedlinks

Referenz

[Workday 31 What's New Post: Workday Studio](#)

Konzept: Verknüpfte Ressourcendateien für Assembly-Projekte

Verknüpfte Ressourcen sind Dateien, die in Ihrem Arbeitsbereich gespeichert sind und mit denen Sie in mehreren Assembly-Projekten verknüpft werden können.

Um eine Liste der verknüpften Ressourcen für ein Projekt aufzurufen, klicken Sie mit der rechten Maustaste auf das Projekt in der Ansicht Project Explorer. Wählen Sie Eigenschaften > Workday > Verknüpfte Ressourcen, um das Fenster Verknüpfte Ressourcen anzuzeigen.

Um eine verknüpfte Ressourcendatei aus einem Projekt zu entfernen, klicken Sie mit der rechten Maustaste auf die Ressourcendatei in der Ansicht Project Explorer und wählen Sie Verknüpfte Ressource entfernen aus.

Um eine verknüpfte Ressource in das Projekt zu kopieren, klicken Sie mit der rechten Maustaste auf die Ressourcendatei in der Ansicht Project Explorer und wählen Sie Convert linked resource to copy aus.

Konzept: Schnellansicht der Gliederung

Die Schnellgliederung bietet eine einfache Möglichkeit, verschiedene Elemente Ihrer Assembly zu suchen. Um die Schnellgliederung zu öffnen, drücken Sie Strg+O, wenn der Assembly-Editor den Fokus darauf gerichtet hat.

Anmerkung: Sie können die Schnellgliederung nicht über das Menü Fenster > - Ansicht anzeigen öffnen.

Mit der Schnellgliederung können Sie Ihre Assembly nach folgenden Kriterien durchsuchen:

Suchart	Beschreibung
MVEL-Ausdruck	Eigenschaften von MVEL-Ausdrücken und -Vorlagen durchsuchen. Verwenden Sie diese Ansicht, um nach Vorkommen von Textwerten zu suchen.
ID	Suchen Sie nach IDs von Assembly-Komponenten.
Variabel	Suchen Sie nach Variablennamen. Studio kategorisiert Variablen in zwei Gruppen: diejenigen, die für den Zugriff auf Schritte definiert sind, und diejenigen, die für das Schreiben von Schritten definiert sind.

Um eine Suchart auszuwählen, klicken Sie auf die entsprechende Schaltfläche unter dem Suchfeld oben in der Ansicht.

Geben Sie zur Suche Text in das Suchfeld ein. Wenn Sie einen MVEL-Ausdruck, eine Komponenten-ID oder eine Variable in der Liste der Suchergebnisse auswählen, springt Studio zur entsprechenden Komponente im Assembly-Diagramm.

Referenz: Assembly-Eigenschaften

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der Baugruppe.

Eigenschaft	XML-Attributname	Beschreibung
Version	version	Versionsnummer der Baugruppe.
Auditprotokoll	audit-log	Einstellungen des Nachrichtenprotokolls für die Baugruppe. Es gibt folgende Optionen: • Wahr: Aktiviert und speichert alle Protokolle. • Falsch: Deaktiviert alle Protokolle. • Bei Fehler: Aktiviert alle Protokolle und speichert nur Fehlerprotokolle.
Callback-Handler-Bean	callback-handler-bean	Die Quell-Bean, die die asynchrone Kommunikation für die Build verarbeitet.
Beanland	beans	Die Quellbean-Pläne in der Baugruppe.
Namespace-Präfixe	xmlns	Die Namespace-Präfixe in der Build.

Referenz: Swimlane-Eigenschaften

Register „Allgemein“

Eigenschaft	Beschreibung
Name	Der Name der Swimlane. Wird im Header der Swimlane angezeigt.
Beschreibung	Eine Beschreibung der Swimlane. Wird angezeigt, wenn Sie Swimlane-Beschreibungen anzeigen aktivieren.
Orientierung	Die Ausrichtung der Swimlane. Folgende Optionen stehen zur Verfügung: • HORIZONTAL • VERTICAL
Ausrichtung	Die Ausrichtung von Komponenten in der Swimlane. Folgende Optionen stehen zur Verfügung: • BEGINNING • MIDDLE • END
Labelausrichtung	Die Ausrichtung des Swimlane-Labels. Folgende Optionen stehen zur Verfügung: • LEFT • CENTER

Eigenschaft	Beschreibung
	• RIGHT
Rahmenfarbe oben	Die Farbe des oberen Rahmens in der Swimlane.
Linienfarbe	Die Farbe des seitlichen und unteren Rahmens der Swimlane.
Füllfarbe	Die Hintergrundfarbe der Swimlane.
Schrift	Die Schriftart des Swimlane-Labels.
Linienstil	Der Stil der seitlichen und unteren Rahmenlinien der Swimlane. Folgende Optionen stehen zur Verfügung: • SOLID • DASH • DOT • DASHDOT • DASHDOTDOT

Referenz: Eigenschaften der Verbindungslinie

Register „Allgemein“

Eigenschaft	Beschreibung
Positionsart	Die Art der Linie, die zur Anzeige von Weiterleiten an und Antwort weiterleiten an in einem Assembly-Diagramm verwendet wird. Folgende Optionen stehen zur Verfügung: • gerade • gebogen

Assembly-Transporte

Enumerationen als Startparameter in Studio verwenden

Vorbereitungen

Erstellen Sie ein Assembly-Projekt, das einen Transport workday-in enthält.

Warum und wann dieser Vorgang ausgeführt wird

In Workday Studio können Sie Enumerationen als Parameter für den Integrationsstart verwenden. Dadurch werden die Werte eingeschränkt, die Sie auf dem Register Integrationskriterien auswählen können, wenn Sie die Integration starten. Sie können die Enumerationen in Studio definieren oder auf vorhandene Enumerationsdefinitionen in Ihrem Mandanten verweisen.

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Klicken Sie auf dem Register Startparameter der Ansicht Eigenschaften für den Transport workday-in auf Parameter für Enumerationsart hinzufügen.

3. Klicken Sie auf Art bearbeiten.
4. Bei der Konfiguration der Enumeration sind folgende Informationen zu beachten:

Option	Bezeichnung
Enumerationsreferenz	Geben Sie den Namen der Art einer vorhandenen Enumerationsdefinition in Ihrem Mandanten ein.
Enumerationsdefinition	Geben Sie einen eindeutigen Name der Art für die Enumerationsdefinition ein. Anmerkung: Wenn Sie den Namen der Art einer vorhandenen Enumerationsdefinition eingeben, überschreibt Workday die Definition. Fügen Sie die Werte hinzu, die die Enumeration zulässt.

Nächste Maßnahme

Stellen Sie Ihre Integration bereit.
relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

Konzept: Assembly-Transporte

Studio bietet 2 Kategorien von Assembly-Transporten in der Palette:

- Eingehende Transporte
- Ausgehende Transporte

Eingangstransporte empfangen Nachrichten von Workday oder einer lokalen Ressource und leiten sie zur Verarbeitung an andere Elemente weiter, die in der Assembly definiert sind. Eingangstransporte geben immer den Beginn einer Verarbeitungskette an.

Ausgangstransporte tauschen Nachrichten mit einer Ressource wie einem HTTP-Endpunkt, einer Workday-Anwendung oder einer FTP-Site aus. Assemblys, die einen lokalen ausgehenden Transport enthalten, können mit Subassemblys kommunizieren, die einen lokalen eingehenden Transport enthalten.

Studio unterstützt diese im Eingangstransports befindlichen Assembly-Elemente auf der Palette:

Name	Funktion	Anmerkungen
workday-in	Definiert einen Startpunkt für die Ausführung einer Assembly.	• Wartet auf Anforderungsnachrichten von Workday. • Unterstützt Startparameter und Servicedefinitionen für Ihre Integration. • Synchron (Anforderung–Antwort)
local-in	Definiert einen Eingabepunkt für eine Assembly oder eine Subassembly.	• Tauscht Nachrichten mit Assemblys aus, die einen lokalen ausgehenden Transport enthalten. • Unterstützt Parameter und Ausgangsparameter.

Name	Funktion	Anmerkungen
		• Synchron (Anforderung–Antwort)

Studio unterstützt die folgenden Elemente von Austransport-Assemblys in der Palette:

Name	Funktion	Anmerkungen
workday-out-rest	Sendet REST-Anforderungsnachrichten für die Workday-Anwendung an eine HTTP-URL.	• Synchron (Anforderung–Antwort)
workday-out-soap	Sendet SOAP-Anforderungsnachrichten für die Workday-Anwendung an eine HTTP-URL.	• Synchron (Anforderung–Antwort)
local-out	Tauscht Nachrichten mit Assemblys und Subassemblys aus, die einen lokalen eingehenden Transport enthalten.	• Synchron (Anforderung–Antwort)
http-out	Sendet HTTP-Anforderungsnachrichten an eine HTTP-URL.	• Unterstützt folgende untergeordnete HTTP-Elemente: • http-basic-auth • http-custom-auth • https-properties • Synchron (Anforderung–Antwort)
ftp-out	Sendet Nachrichten an einen FTP-Server.	• Asynchron (nur Anforderung).
ftps-out	Sendet Nachrichten mithilfe des FTP over SSL (FTPS)-Protokolls sicher an einen FTP-Server.	• Asynchron (nur Anforderung).
sftp-out	Sendet Nachrichten mithilfe des FTP over SSH (SFTP)-Protokolls sicher an einen FTP-Server.	• Asynchron (nur Anforderung).
xmpp-out	Schreibt Textnachrichten an einen Jabber-Server. Beispiel: Ein Google Gmail-Server.	• Asynchron (nur Anforderung).
custom-out	Erzeugt einen individuell anpassbaren Transport.	• Synchronität, die von der Spring-Bean-Implementierung definiert wird.
email-out	Schreibt E-Mails an einen SMTP (Mail)-Server.	• Studio prüft folgende Eigenschaften auf vorangehende

Name	Funktion	Anmerkungen
		und nachfolgende Leerzeichen: • Bis • Thema • Von • Bcc • Cc • Host • Port • Benutzer • Kennwort • Antworten an • Asynchron (nur Anforderung).

Konzept: Enumerationen

Enumerationen beschränken einen Attributwert auf eine Reihe von Literalwerten, z. B. Kostenstellencodes. Sie können Enumerationen verwenden, um einen Satz von Konstanten zu definieren, die beim Design von Integrationen als Literalwerte verwendet werden. Auf diese Weise können Sie den Bereich der von Ihrer Integration verwendeten Daten einschränken.

Sie können Enumerationsdefinitionen in Workday mithilfe der folgenden Aufgaben erstellen und verwalten:

- Definition für Integrationsenumeration erstellen
- Definition für Integrationsenumeration bearbeiten
- Definition für Integrationsenumeration löschen
- Definition für Integrationsenumeration anzeigen

Sie können Enumerationen in den folgenden Elementen verwenden:

- Startparameter.
- Integrationsattribute.
- Integrations-Maps.

Sie können Enumerationen wie folgt zuweisen:

- In Studio.
- Verwenden Sie die Aufgabe Generischen Integrationsservice erstellen in Workday.

Sie können Enumerationen in Studio-Integrationen mit einer der folgenden Methoden zuweisen:

- Erstellen Sie eine Enumerationsdefinition in Studio und stellen Sie sie für Ihren Mandanten bereit, wenn Sie die Studio-Integration starten.
- Erstellen Sie eine Enumerationsreferenz zu einer vorhandenen Enumerationsdefinition in Ihrem Mandanten.

Sie können dieselbe Enumerationsdefinition für mehrere Integrationen verwenden, indem Sie eine Integration auswählen, um die Definition bereitzustellen, und in den anderen darauf verweisen.

relatedlinks

Aufgaben

[Attribut- und Map-Services erstellen](#) auf Seite 315

Referenz: Eigenschaften des Transports „Workday-In“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID des Workday-In- Transports innerhalb der Zusammensetzung.
Weiterleitung an	routes-to	Das Build-Element, an das der Workday-In- Transport die Nachricht weiterleitet.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie " <i>Kein(e)</i> " an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt eine alternative Implementierungsklasse pro Build an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/assembly-bean.xml“ bearbeiten.
Name des Integrationssystems	integration-system-name	Der Name Ihres Integrationssystems.

Register „Startparameter“

Startparameter	Beschreibung
Simple Type Parameter	Startparameter, die Zeichenfolgen-Literalwerte verwenden.
Reference Type Parameter	Startparameter, die Referenzen auf Workday-CRFs und -Webservices verwenden.
Enumeration Type Parameter	Startparameter, die Workday-CRFs und -Enumerationsdefinitionen verwenden.

Register „Services“

Integrationsservice	Beschreibung
Attribute Map Service	Die Integrationsattribute und Integrations-Maps, die in der Integration verwendet werden.

Integrationsservice	Beschreibung
Übermittlungsservice	Übermittelt Dokumente von Ihrem Workday-Mandanten an einen Zielendpunkt.
Listener-Service	Ermöglicht der Integration den Empfang von Startmeldungen von Drittanbieter-Clients über Ihren Workday-Mandanten.
Abrufservice	Ruft Dokumente von externen Clients ab und speichert sie in Ihrem Workday-Mandanten.
Sequenzgeneratorservice	Generiert bei jeder Ausführung der Integration eindeutige, sequenzierte Dateinamen für Dokumente.
Transaktionsprotokollservice	Zeigt die Protokolle für Transaktionsereignisse an, die das Integrationssystem in Ihrem Workday-Mandanten abonniert hat.
Berichtsservice	Gibt Aliase für benutzerdefinierte Berichte an, auf die Sie in der Integration verweisen.
Custom Object Service	Gibt Aliase für benutzerdefinierte Objekte an, auf die Sie in der Integration verweisen.
Service Reference	Referenziert Integrationsservices, die für einen anderen workday-in-Transport innerhalb derselben Sammlung konfiguriert sind.

Referenz: Eigenschaften des Transports „Local-In“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID des lokal eingehenden Transports innerhalb der Baugruppe.
Weiterleitung an	routes-to	Das Build-Element, an das der lokal eingehende Transport die Nachricht weiterleitet.
Zugriff	access	Gibt an, ob Ihre Untergruppe öffentlich oder privat ist. Die Palette zeigt nur öffentliche Subassemblies in der Kategorie Arbeitsbereichskomponenten an.
Globale Fehlerhandler verwenden	use-global-error-handlers	Bestimmt, ob Fehler, die in der Untergruppe auftreten, auf der Ebene der Untergruppe behandelt werden. Der Standardwert ist false, was bedeutet, dass Fehler von der Baugruppe verarbeitet werden,

Eigenschaft	XML-Attributname	Beschreibung
		die die Untergruppe aufgerufen hat.
Symbol	icon	Gibt ein benutzerdefiniertes Symbol für die Untergruppe an. Unterstützt PNG- oder GIF-Symbole im Format 24 x 24 Pixel.
Tooltip	tooltip	Eine QuickInfo für Ihre Untergruppe.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Store Message gilt nicht für lokale Transporte. Studio ignoriert diese Einstellung für local-in-Transporte.
Transportklasse	transport-class	Gibt eine alternative Implementierungsklasse pro Build an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/assembly-bean.xml“ bearbeiten.

Register „Parameter“

Die Parameter des Local-In- Transports geben beliebige an MediationContext Eigenschaften, die Sie festlegen können.

Eigenschaft	XML-Attributname	Beschreibung
Name	name	Der Name Ihres lokalen Eingabeparameters.
Dokumentation	documentation	Eine Textbeschreibung Ihres lokalen Eingabeparameters.
Erforderlich	required	Bestimmt, ob Ihr lokaler In-Transport Ihren Parameter erfordert.
Art	type	Gibt Ihre Parameterart an. Es gibt folgende Optionen: • <i>Zeichenfolge</i> • <i>Ganzzahl</i> • <i>Lang</i> • <i>Doppelt</i> • <i>Float</i> • <i>Boolesch</i>

Eigenschaft	XML-Attributname	Beschreibung
Standard	default	Der Standardwert Ihres Parameters.
Validierung	validation	Ein boolescher MVE-Ausdruck, der als evaluiert werden muss true, andernfalls gibt Studio einen Fehler aus. Ermöglicht es Entwicklern von Untergruppen, Beschränkungen für Parameterwerte festzulegen.

Register „Ausgangsparameter“.

Die Ausgangsparameter des Transports "Lokal-Eingang" geben beliebige an MediationContext - Eigenschaften, die von der Untergruppe abgerufen werden.

Eigenschaft	XML-Attributname	Beschreibung
Name	name	Der Name Ihres Out-Parameters.
Dokumentation	documentation	Eine Textbeschreibung Ihres Out-Parameters.

Referenz: Eigenschaften des Transports „Workday-Out-Rest “

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID des Workday-Out-Rest -Transports innerhalb der Composite.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortnachricht für den Transport von Workday-Out-Rest .
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Workday-Out-Rest -Transport ausgeführt wird.
Zusätzlicher Pfad	extra-path	Gibt den Pfad in Ihrem Workday-Mandanten an, über den Studio die REST-Anforderung sendet. Studio richtet automatisch den Workday-Serviceendpunkt für diesen Pfad ein und hängt ihn an. Beispiel: Der Endpunkt im folgenden Pfad ruft einen Bericht der Art Workday - <i>Niederlassungsverzeichnis</i> von einem Mandanten namens ACME auf, der im JSON-Format angefordert wird:

Eigenschaft	XML-Attributname	Beschreibung
		systemreport2/ACME/ Business_Site_Directory? format=json

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie " <i>Kein(e)</i> " an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt eine alternative Implementierungsklasse pro Build an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/assembly-bean.xml“ bearbeiten.
Methode	method	Ermöglicht das Hinzufügen einer REST-Methode zu Ihrem Workday-Out-Rest -Transport. Beispiel: put, get, post.
Fehlermeldung	failure-message	Eine benutzerdefinierte Meldung, die zum Ersetzen von Ausnahmenmeldungen verwendet wird, die zurückgegeben werden, wenn Sie einen Workday-Out-Rest -Endpunkt aufrufen.

Referenz: Eigenschaften des Transports „Workday-Out-Soap“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID des Workday-Out-Soap- Transports innerhalb der Build-Asset-Mandanten.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortmeldung für den Transport von "workday-out-soap" .
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Workday-Out-Soap -Transport ausgeführt wird.

Eigenschaft	XML-Attributname	Beschreibung
Anwendung	application	Der Name der Workday-Anwendung, an die der Transport Inhalt sendet. Studio verwendet den Namen, den Sie in dieses Feld eingeben, um eine URI zu erstellen, die den Speicherort der Workday-Anwendung angibt.
Version	version	Die Webserviceversion, die von der SOAP-Anforderung aufgerufen wird. Anmerkung: Wenn Sie eine Webservices-Version in einer XSLT-Anforderung angeben, hat diese Version Vorrang.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie " <i>Kein(e)</i> " an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/assembly-bean.xml“ bearbeiten.
Fehlermeldung	failure-message	Eine benutzerdefinierte Meldung, die zum Ersetzen von Ausnahmenmeldungen verwendet wird, die beim Aufrufen eines Workday-out-Soap- Endpunkts zurückgegeben werden.
Ersetzen durch SOIF-Fehler	replace-with-soap-fault	Ersetzt eine fehlgeschlagene Workday-Out-Soap- Meldung durch einen SOAP-Fehler. Der SOAP-Fehler enthält extrahiert faultcode und faultstring Eigenschaftswerten

Referenz: Eigenschaften des Transports „Local-Out“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID des ausgehenden lokalen Transports innerhalb der Baugruppe.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortnachricht für den ausgehenden lokalen Transport.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der lokale ausgehende Transport ausgeführt wird.
Endpunkt	endpoint	Die Adresse des Zielendpunkts, an den der lokal ausgehende Transport die Nachricht weiterleitet. Endpunktadressen folgen dem Format <code>application-name/component-ID</code>, wobei: <code>application-name</code> ist der Name des Projekts, das die Zielgruppe enthält. <code>component-ID</code> ist die ID der Zielkomponente. Beispiel: <code><cc:local-out id="local" endpoint="myassembly/mediation1"/></code>

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie "<i>Kein(e)</i>" an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei

Eigenschaft	XML-Attributname	Beschreibung
		„WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um die WS-Adressierungsdetails anzugeben.
Klonanforderung	clone-request	Gibt an, ob eine Kopie der vom ausgehenden lokalen Transport verarbeiteten Anforderungsnachricht erstellt werden soll, sodass Untergruppen, die die Nachricht ändern, keine Auswirkungen auf die anderen haben.
Abbruch weitergeben	propagate-abort	Gibt an, ob ein Abbruchkennzeichen von einer Untergruppe an die aufrufende Baugruppe weitergegeben wird. Eine Untergruppe stoppt die Verarbeitung, wenn sie als markiert MediationContext als abgebrochen.
Nicht konfigurierte Eigenschaften	unset-properties	Gibt an, ob alle lokalen Ende - Eigenschaften zurückgesetzt werden, nachdem der lokale Ende- Transport eine Untergruppe aufgerufen hat.

Register „Parameter“

Eigenschaft	XML-Attributname	Beschreibung
Name	name	Der Name des untergeordneten Parameterelements.
Wert	value	Der Wert des untergeordneten Parameterelements.

Referenz: Eigenschaften des Transports „HTTP-Out“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID des HTTP-Ausgangstransports innerhalb der Build-gruppe.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortmeldung für den HTTP-Out -Transport. Anmerkung: Der Transport "http-out" füllt die Stelle "" aus MediationContext Eigenschaft <code>http.response.status</code> mit den vom Remote-Server zurückgegebenen Standard-HTTP-Antwortcodes. Beispiele: 202 (Akzeptiert), 403 (Nicht zulässig). Die Baugruppe kann den Wert in dieser Eigenschaft verwenden, um die Behandlung für verschiedene Antworten zu bestimmen.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der HTTP-Out -Transport ausgeführt wird.
Endpunkt	endpoint	Die HTTP-URL-Zieladresse, an die der HTTP-Out- Transport die Anforderungsnachricht weiterleitet. Beispiel: <code>http://myhost.example.org/MyService</code>
Http-Methode	http-method	Die vom http-out- Transport unterstützte HTTP-Anforderungsmethode. Beispiel: DELETE, GET, POST, PUT und PATCH. Anmerkung: Zur Verwendung von PATCH Methode, müssen Sie die festlegen <code>wd.http.client</code> Eigenschaft zu <code>apache</code>. Wenn Sie nicht möchten, dass nachgelagerte Transporte ebenfalls den HTTP-Client verwenden, setzen Sie zurück <code>wd.http.client</code> Eigenschaft zu <code>null</code>. Wenn Sie keine Methode für diese Eigenschaft angeben, verwendet Studio die POST Methode

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie " <i>Kein(e)</i> " an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um die WS-Adressierungsdetails anzugeben.
Verbindungs-Timeout	connect-timeout	Gibt in Millisekunden eine Timeout-Periode für den Verbindungsaufbau mit einem HTTP-Server an. Wenn Sie keinen Wert angeben, wird die Sitzung nach 300.000 ms (5 Minuten) beendet.
Antwort-Timeout	response-timeout	Gibt in Millisekunden eine Timeout-Periode für das Warten auf eine Antwort von einem HTTP-Server an. Wenn Sie keinen Wert angeben, wird die Sitzung nach 2.1600.000 ms (6 Stunden) beendet. Wenn die HTTP-Verbindung für einen längeren Zeitraum nicht reagiert, schlägt der http-out-Transport fehl.
Fehler als Antwort	error-as-response	Damit können Sie eine Fehlermeldung von einem Server in die Meldung für den

Eigenschaft	XML-Attributname	Beschreibung
		HTTP- Ausgangstransport kopieren.
Verbindung schließen	close-connection	Fügt eine hinzu Connection: close -Header mit einer ausgehenden HTTP-Out-Transportnachricht.
Streamen	streaming	Aktiviert HTTP-Streaming mit PUT und POST HTTP-Methoden
Automatische In Bearbeitung	auto-inproc	Gibt an, ob die prozessintegrierte Transportkommunikation für http-out-Anforderungsmeldungen verwendet werden soll. Damit können Sie Nachrichten direkt an Assembly-Endpunkte senden, die sich auf demselben Server befinden, ohne die Netzwerk-E/A-Funktionen des lokalen Hosts zu verwenden.
Wiederholungen	retries	Gibt die Anzahl der Wiederholungsversuche an, die der HTTP-Out -Transport unternimmt, wenn eine Transportausnahme auftritt.
Verzögerung von erneutem Versuch	retry-delay	Gibt eine Verzögerungsperiode in Millisekunden zwischen jedem erneuten Versuch an. Anmerkung: Workday erfüllt die Anforderungen der Retry-After oder RateLimit-Reset -Header-Wert in der Antwort "429 Allow Requests" eines externen Endpunkts, sofern dieser kleiner ist als das Standardmaximum von 5 Minuten oder retry-delay Wert, den Sie hier angeben. Wenn der Antwort-Header-Wert fehlt, verwendet Workday retry-delay -Wert Wenn beide retry-delay und Header-Werte fehlen, verwendet Workday den Standardwert von 5 Minuten. Eine Integration kann keine Verzögerung für Wiederholungsversuche bestimmen und schlägt fehl, wenn der Header-

Eigenschaft	XML-Attributname	Beschreibung
		Wert größer ist als beide <code>retry-delay</code> Wert und das Standardmaximum.
Wiederholungen protokollieren	<code>log-retries</code>	Ermöglicht die Protokollierung von HTTP-Wiederholungsversuchen in der -Serverprotokolldatei.
Gzip akzeptieren	<code>accept-gzip</code>	Gibt an, ob ein HTTP eingeschlossen werden soll <code>accept-encoding=gzip</code> im Anforderungs-Header <code>http-out</code> , das angibt, ob gzip-Inhalte in der Antwort akzeptiert werden.
Gzip-Inhalt	<code>gzip-content</code>	Gibt an, ob ein HTTP eingeschlossen werden soll <code>content-encoding=gzip</code> im Anforderungs-Header <code>http-out</code> , das angibt, ob der Nachrichteninhalte in der ausgehenden Anforderung gzippt werden soll.
Akzeptieren	<code>accept</code>	Gibt an <code>accept</code> -Header-Wert, der mit einer HTTP-Anforderung gesendet werden soll, die eine oder mehrere Inhaltsarten definiert, die in der Antwort von einem HTTP-Server zulässig sind.

Anmerkung: Fügen Sie für Amazon S3-Transporte ein Tag mit dem Schlüssel zu Ihrem IAM-Benutzer hinzu *Workday-Art* und der Wert *Integration*. Bei diesem Tag wird zwischen Groß- und Kleinschreibung unterschieden und Leerzeichen werden nicht ignoriert.

Referenz: Eigenschaften des Transports „FTP-Out“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	<code>id</code>	Die eindeutige ID der FTP-Ausgangskomponente innerhalb der Build.
Antwort weiterleiten an	<code>routes-response-to</code>	Das Ziel der Antwortmeldung für den FTP-Out-Transport.
Ausführung bei	<code>execute-when</code>	Eine Bedingung, die bestimmt, ob der Transport ausgeführt wird.
Endpunkt	<code>endpoint</code>	Gibt das Host- (und optional den Port) und das FTP-Verzeichnis

Eigenschaft	XML-Attributname	Beschreibung
		an, in das Dateien geschrieben werden sollen. Befolgt das Format ftp://FTP-directory, wobei FTP-directory ist der Zielstandort. Beispiel: ftp://ftpserver.example.org/outputDir Anmerkung: Der von Ihnen angegebene Verzeichnisspeicherort muss bereits auf dem FTP-Server vorhanden sein.
Binärdatei	binary-file	Gibt an, ob die Ausgabedateien binär übertragen werden.
Eingabedateimuster	input-file-pattern	Ein Dateinamensmuster für den Abgleich von Dateien eines bestimmten Typs. Beispiel: Angeben *.xls, um nur Excel-Dateien abzurufen.
Eingabeinhaltsart	input-content-type	Die Medienart, die zur Laufzeit für enthalten ist content-type-Wert im Nachrichten-Header. Ein Medientyp umfasst mindestens zwei Teile: einen Typ, einen Untertyp und einen oder mehrere optionale Parameter. Die text Für Unterart ist Folgendes angegeben: charset Parameter, der die Zeichencodierung angibt. Beispiel: text/html; charset=UTF-8 Diese Eigenschaft ist besonders nützlich bei Nicht-XML-Textdateiformaten, bei denen der Medientyp und der Zeichensatz nicht explizit in der Datei angegeben sind.
Ausgabedateimuster	output-file-pattern	Gibt an, wie die Namen der Ausgabedateien generiert werden. Das Muster kann Literalzeichenfolgen sowie eines oder mehrere der folgenden Token enthalten: • \${INFILE}, der durch den Namen der Eingabedatei ersetzt wird.

Eigenschaft	XML-Attributname	Beschreibung
		• <code>\${UUID}</code>, die durch eine allgemein eindeutige Kennung ersetzt wird. • <code>\${EXT}</code>, am Ende des Musters platziert und durch eine Dateinamenserweiterung ersetzt, die auf dem Inhaltstyp der Daten basiert, die in die Ausgabe der Build-Komponente geschrieben werden.
Methode	method	Gibt die Übertragungsmethode an: • <code>put</code>: Schreibt Dateien in den FTP-Endpunkt, der das Remote-Verzeichnis enthält. • <code>list</code>: listet alle Dateien und Verzeichnisse am FTP-Endpunkt auf. Verwenden Sie die Eigenschaft "Pfad für Eingabedatei" für <code>list</code>, um ein Muster auf das zurückgegebene Angebot anzuwenden. Sie können auf die Ergebnisse von zugreifen <code>list</code>-Methode unter Verwendung des <code>ftp</code>-Objekts MVE. Beispiel: <code>ftp.list()</code> gibt eine Liste mit Zeichenfolgen zurück, die die zuletzt abgerufene Liste enthält. • <code>get</code>: Ruft die erste gefundene Datei ab, die mit dem Eigenschaftswert Input File Template übereinstimmt. • <code>delete</code>: Löscht die erste gefundene Datei, die dem Eigenschaftswert von "Muster für Eingabedatei" entspricht. Anmerkung: Eine Datei wird nicht automatisch von der FTP-Site nach einer gelöscht get wurde durchgeführt. Wenn

Eigenschaft	XML-Attributname	Beschreibung
		Sie eine Datei löschen möchten, müssen Sie eine separate Methode aufrufen. Beispiel: Sie können eine separate einschließen ftp-out Transport speziell, um die Datei zu löschen.
Kennwort	password	Das Kennwort, das dem Benutzernamen für den Zugriff auf den FTP-Server zugeordnet ist.
Benutzername	username	Der Benutzername, der für den Zugriff auf den FTP-Server verwendet wird.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie "Kein(e)" an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um die WS-Adressierungsdetails anzugeben.
Passiver Modus	passive-mode	Gibt an, ob der Transport FTP im passiven Modus unterstützt.

Eigenschaft	XML-Attributname	Beschreibung
		Im passiven Modus initiiert der FTP-Client beide für den FTP-Transfer erforderlichen Verbindungen. Der Client kontaktiert den Server über den Befehlsport und gibt den Befehl PASV aus. Der Server antwortet dann und gibt an, an welchem Port er die Datenverbindung empfängt. Der Client initiiert dann die Datenverbindung von seinem Datenport zum angegebenen Server-Datenport. Schließlich sendet der Server eine Bestätigung an den Datenport des Clients zurück. Im aktiven Modus initiiert der Client die Befehlsverbindung und der Server initiiert die Datenverbindung.
Timeout	timeout	Gibt in Millisekunden eine Timeout-Periode für FTP-Sitzungen zwischen Clients und dem FTP-Server an, auf dem der Transport empfangen wird. Wenn Sie keinen Wert angeben, wird die Sitzung nach 300.000 ms (5 Minuten) beendet.
MIME-Typen mappen	mime-types-map-file	Den Speicherort einer Kartendatei für das Mapping zwischen MIME-Typen und Dateierweiterungen. Verwenden Sie diese Eigenschaft für Folgendes: • Ordnen Sie MIME-Typen Dateierweiterungen zu, wenn die Eigenschaft „Methode“ auf festgelegt ist put. • Ordnen Sie Dateierweiterungen den MIME-Typen zu, wenn die Eigenschaft „Methode“ auf festgelegt ist get. Die Datei für dieses Mapping heißt ccx/config/mIME.types. Sie können Ihre eigene Version dieser Datei bereitstellen.

Eigenschaft	XML-Attributname	Beschreibung
		Dieser Transport kann Dateinamen für seine Ausgabedateien mithilfe von generieren \${EXT} Token am Ende von platziert output - file - pattern -Wert
Name der temporären Datei	temp - file - name	Gibt einen temporären Dateinamen an. Wenn die Eigenschaft „Methode“ auf eingestellt ist put: Die Datei wird unter diesem temporären Dateinamen hochgeladen. Nach Abschluss des FTP wird die Datei in den Namen umbenannt, der in der Eigenschaft „Muster für Ausgabedatei“ angegeben ist.
Verbindung schließen	close - connection	Gibt an, ob die Verbindung getrennt wird, wenn die FTP-Aktion abgeschlossen ist.
Poolgröße	pool - size	Die maximale Anzahl gleichzeitiger Verbindungen und Sitzungen, die pro Host für diese Komponente unterstützt werden können. Wenn mehr als die angegebene Anzahl von Threads versuchen, eine Verbindung zu demselben Host herzustellen, müssen einige Threads auf Zugriff auf den Pool warten. Wenn die Poolgröße unbegrenzt sein soll, geben Sie eine negative Zahl an.
Debuggen	debug	Aktiviert den JSCAPI-Debug-Modus für den Transport.

Referenz: Eigenschaften für Transport „FTPS-Out“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der ftps-out-Komponente innerhalb der Build.
Antwort weiterleiten an	routes - response - to	Das Ziel der Antwortmeldung für den ausgehenden FPS-Transport.

Eigenschaft	XML-Attributname	Beschreibung
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Transport ausgeführt wird.
Endpunkt	endpoint	Gibt das Host- (und optional den Port) und das FTP-Verzeichnis an, in das Dateien geschrieben werden sollen. Befolgt das Format <code>ftps://FTPS-directory</code>, wobei FTPS-directory ist der Zielstandort. Beispiel: <code>ftps://ftpserver.example.org/outputDir</code> Anmerkung: Der von Ihnen angegebene Verzeichnisspeicherort muss bereits auf dem Server vorhanden sein.
Binärdatei	binary-file	Gibt an, ob die Ausgabedateien als Binärobjekte übertragen werden.
Explizite SSL	explicit-ssl	Gibt die Verbindungsart an: explizites SSL (AUTH TLS) oder implizites SSL (AUTH SSL). Auf <code>true</code>, um explizite SLL zu verwenden. FTP-Clients können wählen, ob sie die Kommunikation unverschlüsselt als normale FTP-Sitzung fortsetzen oder über den Befehl AUTH in den FTPS-Modus wechseln möchten.
Eingabedateimuster	input-file-pattern	Ein Dateinamensmuster für den Abgleich von Dateien eines bestimmten Typs. Beispiel: Angeben <code>*.xls</code>, um nur Excel-Dateien abzurufen.
Eingabeinhaltsart	input-content-type	Die Medienart, die zur Laufzeit für enthalten ist <code>content-type</code>-Wert im Nachrichten-Header. Ein Medientyp umfasst mindestens zwei Teile: einen Typ, einen Untertyp und einen oder mehrere optionale Parameter. Die text Für Unterart ist Folgendes angegeben:

Eigenschaft	XML-Attributname	Beschreibung
		charset Parameter, der die Zeichencodierung angibt. Beispiel: text/html; charset=UTF-8 Diese Eigenschaft ist besonders nützlich bei Nicht-XML-Textdateiformaten, bei denen der Medientyp und der Zeichensatz nicht explizit in der Datei angegeben sind.
Ausgabedateimuster	output-file-pattern	Gibt an, wie die Namen der Ausgabedateien generiert werden. Das Muster kann Literalzeichenfolgen sowie eines oder mehrere der folgenden Token enthalten: • <code>\${INFILE}</code>, der durch den Namen der Eingabedatei ersetzt wird. • <code>\${UUID}</code>, die durch eine allgemein eindeutige Kennung ersetzt wird. • <code>\${EXT}</code>, am Ende des Musters platziert und durch eine Dateinamenserweiterung ersetzt, die auf dem Inhaltstyp der Daten basiert, die in die Ausgabe der Build-Komponente geschrieben werden.
Methode	method	Gibt die Übertragungsmethode an: • <code>put</code>: Schreibt Dateien in den FTP-Endpunkt, der das Remote-Verzeichnis enthält. • <code>list</code>: listet alle Dateien und Verzeichnisse am FTP-Endpunkt auf. Verwenden Sie die Eigenschaft "Pfad für Eingabedatei" für <code>list</code>, um ein Muster auf das zurückgegebene Angebot anzuwenden. Sie können auf die Ergebnisse von zugreifen <code>list</code>-Methode unter Verwendung des <code>ftp</code>-

Eigenschaft	XML-Attributname	Beschreibung
		Objekts MVE. Beispiel: <code>ftp.list()</code> gibt eine Liste mit Zeichenfolgen zurück, die die zuletzt abgerufene Liste enthält. • <code>get</code>: Ruft die erste gefundene Datei ab, die mit dem Eigenschaftswert <code>Input File Template</code> übereinstimmt. • <code>delete</code>: Löscht die erste gefundene Datei, die dem Eigenschaftswert von "Muster für Eingabedatei" entspricht. Anmerkung: Eine Datei wird nicht automatisch von der FTP-Site nach einer gelöscht <code>get</code> wurde durchgeführt. Wenn Sie eine Datei löschen möchten, müssen Sie eine separate Methode aufrufen. Beispiel: Sie können eine separate einschließen <code>ftp-out</code> Transport speziell, um die Datei zu löschen.
Kennwort	password	Das Kennwort, das dem Benutzernamen für den Zugriff auf den FTP-Server zugeordnet ist.
Benutzername	username	Der Benutzername, der für den Zugriff auf den FTP-Server verwendet wird.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie "Kein(e)" an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt <code>store</code> in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global

Eigenschaft	XML-Attributname	Beschreibung
		ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um die WS-Adressierungsdetails anzugeben.
Passiver Modus	passive-mode	Gibt an, ob der Transport FTP im passiven Modus unterstützt. Im passiven Modus initiiert der FTP-Client beide für den FTP-Transfer erforderlichen Verbindungen. Der Client kontaktiert den Server über den Befehlsport und gibt den Befehl PASV aus. Der Server antwortet dann und gibt an, an welchem Port er die Datenverbindung empfängt. Der Client initiiert dann die Datenverbindung von seinem Datenport zum angegebenen Server-Datenport. Schließlich sendet der Server eine Bestätigung an den Datenport des Clients zurück. Im aktiven Modus initiiert der Client die Befehlsverbindung und der Server initiiert die Datenverbindung.
Timeout	timeout	Gibt in Millisekunden eine Timeout-Periode für FTP-Sitzungen zwischen Clients und dem FTP-Server an, auf dem der Transport empfangen wird. Wenn Sie keinen Wert angeben, wird die Sitzung nach 300.000 ms (5 Minuten) beendet.
MIME-Typen mappen	mime-types-map-file	Gibt den Speicherort einer Kartendatei für das Mapping zwischen MIME-Typen und Dateierweiterungen

Eigenschaft	XML-Attributname	Beschreibung
		an. Verwenden Sie diese Eigenschaft für Folgendes: • Ordnen Sie MIME-Typen Dateierweiterungen zu, wenn die Eigenschaft „Methode“ auf festgelegt ist put. • Ordnen Sie Dateierweiterungen den MIME-Typen zu, wenn die Eigenschaft „Methode“ auf festgelegt ist get. Die Datei für dieses Mapping heißt <code>ccx/config/mIME.types</code>. Sie können Ihre eigene Version dieser Datei bereitstellen. Dieser Transport kann Dateinamen für seine Ausgabedateien mithilfe von generieren <code>\${EXT}</code> Token am Ende von platziert <code>output - file-pattern</code> -Wert
Name der temporären Datei	<code>temp-file-name</code>	Gibt einen temporären Dateinamen an. Wenn die Eigenschaft „Methode“ auf eingestellt ist put: Die Datei wird unter diesem temporären Dateinamen hochgeladen. Nach Abschluss des FTP wird die Datei in den Namen umbenannt, der in der Eigenschaft „Muster für Ausgabedatei“ angegeben ist.
Verbindung schließen	<code>close-connection</code>	Gibt an, ob die Verbindung getrennt wird, wenn die FTP-Aktion abgeschlossen ist.
Poolgröße	<code>pool-size</code>	Die maximale Anzahl gleichzeitiger Verbindungen und Sitzungen, die pro Host für diese Komponente unterstützt werden können. Wenn mehr als die angegebene Anzahl von Threads versuchen, eine Verbindung zu demselben Host herzustellen, müssen einige Threads auf Zugriff auf den Pool warten.

Eigenschaft	XML-Attributname	Beschreibung
		Wenn die Poolgröße unbegrenzt sein soll, geben Sie eine negative Zahl an.
Debuggen	debug	Aktiviert den JSCAPI-Debug-Modus für den Transport.

Referenz: Eigenschaften des Transports „SFTP-Out“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der SFTP-Ausgangskomponente innerhalb der Build.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortmeldung für den SFTP-Ausgangstransport .
Ausführung bei	execute-when	Gibt eine Bedingung an, die bestimmt, ob der Transport ausgeführt wird.
Endpunkt	endpoint	Gibt das Host- (und optional den Port) und das FTP-Verzeichnis an, in das Dateien geschrieben werden sollen. Befolgt das Format <code>sftp://SFTP-directory</code>, wobei SFTP-directory ist der Zielstandort. Beispiel: <code>sftp://sftpserver.example.org/outputDir</code> Anmerkung: Der von Ihnen angegebene Verzeichnisspeicherort muss bereits auf dem Server vorhanden sein.
Doppelte Authentifizierung	dual-authentication	Gibt an, ob eine doppelte Authentifizierung sowohl mit einer Benutzernamen/Kennwort-Kombination als auch mit einem privaten Schlüssel erzwungen werden soll. Der Standardwert ist <code>false</code> : Dies bedeutet, dass sich ein Benutzer entweder mit einem privaten Schlüssel oder mit einer Benutzernamen/Kennwort-Kombination authentifizieren kann, aber nicht mit beiden.

Eigenschaft	XML-Attributname	Beschreibung
		Wenn die Eigenschaft auf eingestellt ist <code>false</code> und ein Benutzer gibt einen privaten Schlüssel ein (indem er die Eigenschaften des privaten Schlüssels für die sftp-out-Komponente konfiguriert), dann verwendet die Authentifizierung automatisch diesen Schlüsselwert und den Benutzernamen und ignoriert das Kennwortfeld. Wenn die Eigenschaft auf eingestellt ist <code>false</code> und kein privater Schlüssel konfiguriert ist, verwendet die Authentifizierung die Kombination aus Benutzername/Kennwort. Wenn die Eigenschaft auf eingestellt ist <code>true</code> dann verwendet die Authentifizierung sowohl den Wert des privaten Schlüssels als auch die Kombination aus Benutzername und Kennwort.
Fingerabdruck für Host-Schlüssel	host-key-fingerprint	Der Verschlüsselungsschlüssel, den der FTP-Server für die SSH-Kommunikation verwendet.
Benutzername	username	Der Benutzername, der für den Zugriff auf den FTP-Server verwendet wird.
Kennwort	password	Das Kennwort, das dem Benutzernamen für den Zugriff auf den FTP-Server zugeordnet ist.
Binärdatei	binary-file	Gibt an, ob die Ausgabedateien binär übertragen werden.
Explizite SSL	explicit-ssl	Verbindungsart: explizites SSL (AUTH TLS) oder implizites SSL (AUTH SSL). Auf <code>true</code> , um explizites SSL zu verwenden.
Eingabedateimuster	input-file-pattern	Ein Dateinamensmuster für den Abgleich von Dateien eines bestimmten Typs. Beispiel: Angeben <code>*.xls</code>, um nur Excel-Dateien abzurufen.
Eingabeinhaltsart	input-content-type	Die Medienart, die zur Laufzeit für enthalten ist <code>content-type</code> -Wert im Nachrichten-

Eigenschaft	XML-Attributname	Beschreibung
		Header. Ein Medientyp umfasst mindestens zwei Teile: einen Typ, einen Untertyp und einen oder mehrere optionale Parameter. Die text Für Unterart ist Folgendes angegeben: charset Parameter, der die Zeichencodierung angibt. Beispiel: text/html; charset=UTF-8 Diese Eigenschaft ist besonders nützlich bei Nicht-XML-Textdateiformaten, bei denen der Medientyp und der Zeichensatz nicht explizit in der Datei angegeben sind.
Ausgabedateimuster	output-file-pattern	Gibt an, wie die Namen der Ausgabedateien generiert werden. Das Muster kann Literalzeichenfolgen sowie eines oder mehrere der folgenden Token enthalten: • <code>\${INFILE}</code>, der durch den Namen der Eingabedatei ersetzt wird. • <code>\${UUID}</code>, die durch eine allgemein eindeutige Kennung ersetzt wird. • <code>\${EXT}</code>, am Ende des Musters platziert und durch eine Dateinamenserweiterung ersetzt, die auf dem Inhaltstyp der Daten basiert, die in die Ausgabe der Build-Komponente geschrieben werden.
Methode	method	Gibt die Übertragungsmethode an: • <code>put</code>: Schreibt Dateien in den FTP-Endpunkt, der das Remote-Verzeichnis enthält. • <code>list</code>: listet alle Dateien am FTP-Endpunkt auf. Verwenden Sie die Eigenschaft "Pfad für Eingabedatei" für <code>list</code>, um ein Muster auf

Eigenschaft	XML-Attributname	Beschreibung
		das zurückgegebene Angebot anzuwenden. Sie können auf die Ergebnisse von zugreifen <code>list</code> -Methode unter Verwendung des <code>ftp</code>-Objekts MVE. Beispiel: <code>ftp.list()</code> gibt eine Liste mit Zeichenfolgen zurück, die die zuletzt abgerufene Liste enthält. • <code>get</code>: Ruft die erste gefundene Datei ab, die mit dem Eigenschaftswert <code>Input File Template</code> übereinstimmt. • <code>delete</code>: Löscht die erste gefundene Datei, die dem Eigenschaftswert von <code>"Muster für Eingabedatei"</code> entspricht. Anmerkung: Eine Datei wird nicht automatisch von der FTP-Site nach einer gelöscht <code>get</code> wurde durchgeführt. Wenn Sie eine Datei löschen möchten, müssen Sie eine separate Methode aufrufen. Beispiel: Sie können eine separate einschließen <code>ftp-out</code> Transport speziell, um die Datei zu löschen.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	<code>store-message</code>	Geben Sie <i>"Kein(e)"</i> an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt <code>store</code> in Studio.
Transportklasse	<code>transport-class</code>	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei

Eigenschaft	XML-Attributname	Beschreibung
		„WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt <code>set-dynamic-endpoint</code>, um die WS-Adressierungsdetails anzugeben.
Timeout	timeout	Gibt in Millisekunden eine Timeout-Periode für FTP-Sitzungen zwischen Clients und dem FTP-Server an, auf dem der Transport empfangen wird. Wenn Sie keinen Wert angeben, wird die Sitzung nach 300.000 ms (5 Minuten) beendet.
MIME-Typen mappen	mime-types-map-file	Gibt den Speicherort einer Kartendatei für das Mapping zwischen MIME-Typen und Dateierweiterungen an. Verwenden Sie diese Eigenschaft für Folgendes: • Ordnen Sie MIME-Typen Dateierweiterungen zu, wenn die Eigenschaft „Methode“ auf festgelegt ist <code>put</code>. • Ordnen Sie Dateierweiterungen den MIME-Typen zu, wenn die Eigenschaft „Methode“ auf festgelegt ist <code>get</code>. Die Datei für dieses Mapping heißt <code>ccx/config/mIME.types</code>. Sie können Ihre eigene Version dieser Datei bereitstellen. Dieser Transport kann Dateinamen für seine Ausgabedateien mithilfe von generieren <code>\${EXT}</code> Token am Ende von platziert <code>output-file-pattern</code> -Wert

Eigenschaft	XML-Attributname	Beschreibung
Name der temporären Datei	temp-file-name	Gibt einen temporären Dateinamen an. Wenn die Eigenschaft „Methode“ auf eingestellt ist put: Die Datei wird unter diesem temporären Dateinamen hochgeladen. Nach Abschluss des FTP wird die Datei in den Namen umbenannt, der in der Eigenschaft „Muster für Ausgabedatei“ angegeben ist.
Verbindung schließen	close-connection	Gibt an, ob die Verbindung getrennt wird, wenn die FTP-Aktion abgeschlossen ist.
Poolgröße	pool-size	Die maximale Anzahl gleichzeitiger Verbindungen und Sitzungen, die pro Host für diese Komponente unterstützt werden können. Wenn mehr als die angegebene Anzahl von Threads versuchen, eine Verbindung zu demselben Host herzustellen, müssen einige Threads auf Zugriff auf den Pool warten. Wenn die Poolgröße unbegrenzt sein soll, geben Sie eine negative Zahl an.
Debuggen	debug	Aktiviert den JSCAPI-Debug-Modus für den Transport.

Referenz: Eigenschaften des Transports „XMPP-Out“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der xmpp-out- Komponente innerhalb der Build.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortmeldung für den FTP-Out- Transport.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Transport ausgeführt wird.
Endpunkt	endpoint	Gibt den Empfänger der Instant Message an. Beispiel: xmpp:logan.mcneil@gmail.com

Eigenschaft	XML-Attributname	Beschreibung
Domäne	domain	Die Domäne, in der der Benutzername registriert ist. Beispiel: gmail.com
Server	server	Der Host oder die IP-Adresse des XMPP-/Jabber-Servers.
Port	port	Die Portnummer, an der der XMPP-Transport auf Nachrichten wartet. Der von den meisten XMPP-Servern verwendete Port ist 5222.
Benutzername	username	Der Benutzername, der für den Zugriff auf den FTP-Server verwendet wird.
Kennwort	password	Das Kennwort, das dem Benutzernamen für den Zugriff auf den FTP-Server zugeordnet ist.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie " <i>Kein(e)</i> " an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um die WS-Adressierungsdetails anzugeben.

Referenz: Eigenschaften des Transports „Custom-Out“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der benutzerdefinierten Ausgangskomponente in der Baugruppe.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortnachricht für den benutzerdefinierten ausgehenden Transport.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Transport ausgeführt wird.
Endpunkt	endpoint	Der Endpunkt, an den der benutzerdefinierte ausgehende Transport Nachrichten sendet. Wird von der Spring-Bean-Implementierung custom-out verwendet.
Eingabe	input	Gibt an, wo die Build-Komponente Daten als Eingabe bezieht. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin die Baugruppenkomponente

Eigenschaft	XML-Attributname	Beschreibung
		die Ausgabe leitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Quell Bean	spring-bean	Gibt die Quell-Bean an, die die Schnittstellenklasse des Transports implementiert.
Methodenname	method-name	Der Name der Java-Methode in der benutzerdefinierten Java-Klasse.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie "Kein(e)" an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an.

Eigenschaft	XML-Attributname	Beschreibung
		Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um die WS-Adressierungsdetails anzugeben.

Referenz: Eigenschaften für Transport „Email-Out“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der E-Mail-Ausgangskomponente innerhalb der Build.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortnachricht für den ausgehenden E-Mail-Transport.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Transport ausgeführt wird.
Bis	to	Die bevorzugte E-Mail-Adresse des vorgesehenen Empfängers. Befolgt das Format mailto: [emailaddress]. Wenn Sie nur BCC-Adressen verwenden möchten, geben Sie einen leeren Wert an mailto: - Wert
Thema	subject	Die Betreffzeile der E-Mail.
Von	from	Die E-Mail-Adresse, die im Feld " Von" der übermittelten E-Mail angezeigt werden soll.
Ssl	ssl	Gibt an, ob die Verbindung per SSL erfolgt.

Eigenschaft	XML-Attributname	Beschreibung
Starttls	starttls	Gibt an, ob eine sofortige SSL-Aushandlung mit dem Remote-SMTP-Server aktiviert werden soll, indem vor dem Start der SMTP-Sitzung der Befehl STARTTLS ausgegeben wird. Der SMTP-Server antwortet mit einem der folgenden Antwortcodes: • 220 Bereit zum Starten • TLS 501 Syntaxfehler (keine Parameter zulässig) • 454 TLS aus einem vorübergehenden Grund nicht verfügbar Anmerkung: Die Eigenschaften ssl und starttls schließen sich gegenseitig aus.
Host	host	Der Name oder die IP-Adresse des SMTP-Mailservers, den der Transport zur Übermittlung der Nachricht verwendet. Wenn Sie OAuth verwenden, legen Sie diesen Wert auf fest localhost.
Port	port	Die Portnummer, unter der der E-Mail-Server ausgeführt wird. Der Standardwert ist 25, der Standard-SMTP-Port. Ausnahmen: • Wenn für die ssl - Eigenschaft festgelegt ist true und kein Port angegeben ist, ist der Standardwert 465. • Wenn für die Eigenschaft Starttls Folgendes festgelegt ist: true und kein Port angegeben ist, ist der Standardwert 587. Anmerkung:
Benutzer		Der Benutzername für den Zugriff auf den E-Mail-Server.
Kennwort	password	Das Kennwort für den - Benutzernamen, der in der Eigenschaft "User" angegeben ist.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie " <i>Kein(e)</i> " an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um die WS-Adressierungsdetails anzugeben.
Bcc	bcc	Gibt durch ein Semikolon getrennte E-Mail-Empfänger für anonyme Kopien an. Beispiel: logan.mcneil@workday.com; jake.lee@workday.com.
Cc	cc	Gibt E-Mail-Empfänger für Durchschlagskopien mithilfe eines Semikolon-Trennzeichens an. Beispiel: logan.mcneil@workday.com; jake.lee@workday.com.
Antworten an	reply-to	Die E-Mail-Adresse, die automatisch in das Feld An eingefügt wird, wenn ein Benutzer auf eine E-Mail-Nachricht antwortet. Für die meisten E-Mail-Nachrichten reicht die Adresse im Feld Von aus, sodass die Eigenschaft Antworten an nicht definiert werden muss. Geben Sie mehrere E-Mail-Adressen

Eigenschaft	XML-Attributname	Beschreibung
		mithilfe eines Semikolon-Trennzeichens an. Beispiel: <code>logan.mcneil@workday.com;jake.lee@workday.com</code> .
Timeout	<code>timeout</code>	Gibt in Millisekunden eine Timeout-Periode für FTP-Sitzungen zwischen Clients und dem FTP-Server an, auf dem der Transport empfangen wird. Wenn Sie keinen Wert angeben, erfolgt das Sitzungstimeout nach 300000 ms (5 Minuten).
Vorübergehende Ausnahmen	<code>transient-exceptions</code>	Vorübergehende Ausnahmen sind Ausnahmen, die ohne Änderungen bei einem erneuten Versuch erfolgreich sein könnten. Nicht vorübergehende Ausnahmen sind solche, die bei einem erneuten Versuch fehlschlagen, bis die zugrunde liegende Ursache des Problems behoben ist. Workday behandelt standardmäßig nur <code>java.net.ConnectException</code> und <code>java.net.SocketException</code> Fehler als vorübergehende Ausnahmen. Geben Sie zusätzliche vorübergehende Ausnahmen als Komma, Leerzeichen oder durch Kommas und Leerzeichen getrennte Enumerationen von Klassennamen an. Beispiele: <code>java.net.UnknownHostException</code> <code>java.net.ConnectException</code> <code>java.net.SocketException</code> <code>java.net.UnknownHostException,j</code> <code>java.net.UnknownHostException, java.net.ConnectException, java.net.SocketException</code> <code>@{new java.net.UnknownHostException(</code> Anmerkung: Wenn Sie zusätzliche vorübergehende Ausnahmen angeben, müssen Sie auch die Standardausnahmen einschließen <code>java.net.ConnectException</code>

Eigenschaft	XML-Attributname	Beschreibung
		und <code>java.net.SocketException</code> in der Liste. Wenn Sie dies nicht tun, überschreibt Workday diese.

Register „SMTP Headers“

Eigenschaft	XML-Attributname	Beschreibung
Name	name	Der Name eines benutzerdefinierten SMTP-Headers. Um einen neuen benutzerdefinierten Header unter vorhandenen Zeilen hinzuzufügen, klicken Sie auf Hinzufügen und geben dann Name und Wert an. Um einen neuen Header über einer vorhandenen Zeile hinzuzufügen, platzieren Sie den Cursor in dieser Zeile und klicken Sie dann im Dropdown-Menü Hinzufügen auf Einfügen. Verwenden Sie die Schaltflächen rechts neben dem Register SMTP Headers, um Zeilen zu löschen oder neu anzuordnen.
Wert	value	Der Wert eines benutzerdefinierten SMTP-Headers.

OAuth Tab

Um die Microsoft Outlook OAuth-Authentifizierung mit der Komponente „E-Mail-Ausgang“ von Studio zu verwenden, registrieren Sie Studio als App bei Microsoft Azure, bieten domänenweite Delegation und erteilen die Administratorzustimmung für Mail.Send -Berechtigungen und rufen verschiedene Werte ab. Weitere Informationen finden Sie unter <https://learn.microsoft.com/en-gb/azure/active-directory/develop/v2-oauth2-client-creds-grant-flow>.

Um die Google Gmail OAuth-Authentifizierung mit der Komponente „Email-Out“ von Studio zu verwenden, registrieren Sie ein Servicekonto bei Google Cloud, stellen Sie eine domänenweite Delegation bereit und rufen Sie eine JSON-Beschreibung für verschiedene Werte ab. Weitere Informationen finden Sie unter <https://support.google.com/cloud/answer/6158849?hl=en>.

Eigenschaft	XML-Attributname	Beschreibung
Anbieter	provider	Den Name des E-Mail-Anbieters MICROSOFT_OUTLOOK oder GOOGLE_GMAIL.

Eigenschaft	XML-Attributname	Beschreibung								
Name	name	Die von Workday vorgegebenen Namen, die Sie mit verschiedenen Werten aus dem Microsoft Azure-Portal oder der Google Cloud Console verknüpfen. Geben Sie für Microsoft Outlook die folgenden Namen ein: - outlook.client.id - outlook.client.secret - outlook.tenant.id - outlook.user.id Geben Sie für Google Gmail Folgendes ein: - gmail.service.account.key.json - gmail.delegated.user								
Wert	value	Die von Microsoft oder Google erhaltenen Werte, die Sie oben mit den Namen verknüpfen. Erstellen Sie für Microsoft Outlook die folgenden Paare aus Name und Wert: <table><tr><th>Name</th><th>Wert</th></tr><tr><td>outlook.client.id</td><td>Auf der Seite Übersicht einer registrierten App im Microsoft Azure-Portal: Anwendungs-ID (Client).</td></tr><tr><td>outlook.client.secret</td><td>Auf der Seite Certificates & secrets einer registrierten App im Microsoft Azure-Portal: ID des geheimen Client-Schlüssels.</td></tr><tr><td>outlook.tenant.id</td><td>Auf der Seite Übersicht einer registrierten</td></tr></table>	Name	Wert	outlook.client.id	Auf der Seite Übersicht einer registrierten App im Microsoft Azure-Portal: Anwendungs-ID (Client).	outlook.client.secret	Auf der Seite Certificates & secrets einer registrierten App im Microsoft Azure-Portal: ID des geheimen Client-Schlüssels.	outlook.tenant.id	Auf der Seite Übersicht einer registrierten
Name	Wert									
outlook.client.id	Auf der Seite Übersicht einer registrierten App im Microsoft Azure-Portal: Anwendungs-ID (Client).									
outlook.client.secret	Auf der Seite Certificates & secrets einer registrierten App im Microsoft Azure-Portal: ID des geheimen Client-Schlüssels.									
outlook.tenant.id	Auf der Seite Übersicht einer registrierten									

Eigenschaft	XML-Attributname	Beschreibung	
		Name	Wert
			App im Microsoft Azure-Portal: Verzeichnis-ID (Mandant).
		outlook.user.id	Die Outlook-Adresse des Kontos, von dem die E-Mail gesendet wird. Wird auch in das Feld Von auf dem Register Allgemein eingegeben.
		Erstellen Sie für Google Gmail die folgenden Paare aus Name und Wert:	
		Name	Wert
		gmail.service.accounts	Dem JSON-Text aus der Datei, die aus dem Abschnitt KEYS der Seite Service accounts in der Google Cloud-Konsole heruntergeladen wurde.
		gmail.delegated	Die Gmail-Adresse, von der die E-Mail gesendet wird. Muss ein Hauptelement sein, das mit dem Google-Servicekonto verknüpft ist.

Eigenschaft	XML-Attributname	Beschreibung	
		Name	Wert
			Wird auch in das Feld Von auf dem Register Allgemein eingegeben.

Gmail über SSL oder mit STARTTLS

Stellen Sie bei der Konfiguration des Transports email-out für die Verwendung von Gmail über SSL oder STARTTLS Folgendes sicher:

- Benutzer und Kennwort enthalten gültige Gmail-Zugangsdaten.
- Endpunkt ist `mailto:[name]@gmail.com`.
- Host ist `smtp.gmail.com`.
- `ssl` oder `starttls` ist auf `gesetzt true`.
- Port ist entweder:
 - 465, wenn für `ssl` eingestellt ist `true`.
 - 587, wenn `starttls` auf `gesetzt true` ist.
- Timeout ist 30000 (30 Sekunden).

Microsoft Office 365 über SSL oder mit STARTTLS

Stellen Sie bei der Konfiguration des Transports email-out für die Verwendung von Microsoft Office 365 über SSL oder STARTTLS Folgendes sicher:

- Benutzer und Kennwort enthalten gültige Office 365-Zugangsdaten.
- Endpunkt ist `mailto:[name]@[domain-name.top-level-domain]`.
- Host ist `smtp.office365.com`.
- `ssl` oder `starttls` ist auf `gesetzt true`.
- Port :
 - 465, wenn für `ssl` eingestellt ist `true`.
 - 587, wenn `starttls` auf „true“gesetzt ist.
- Timeout ist 30000 (30 Sekunden).

Nachrichtentext und Anhänge für nicht-binäre Formate konfigurieren

Wenn sie eingehende Nachrichten empfangen, gleichen viele E-Mail-Anwendungen den ersten Textteil der MIME-Nachricht ab und verwenden diesen als Nachrichtentext. Um den MIME-Typ des Root-Nachrichtenteils auf festzulegen `text/plain`, setzen Sie die Eingabe- und Ausgabe- Eigenschaften der Nachricht auf `message` und dann den Output- MIME-Typ auf `text/plain`.

Verwenden Sie den Nachrichten-Builder , um den Nachrichteninhalt für den E-Mail-Nachrichtentext mithilfe eines Textelements zu definieren.

Setzen Sie für den Anhang die Eigenschaft Ausgabe auf *den Anhang*, den Anhangindex für Ausgabe auf 0 und der Ausgabe -MIME-Typ für `text/eindeutig`.

Verwenden Sie erneut den Nachrichten-Builder , um den Nachrichteninhalt für den Anhang mithilfe eines Textelements zu definieren.

Um zu verhindern, dass einige E-Mail-Anwendungen Textanhänge im Nachrichtentext anzeigen, legen Sie den MIME-Typ des Anhangs auf `application/octet-stream` fest, auch wenn es sich tatsächlich um

Text handelt. Verwenden Sie dazu einen MVAL-Ausdruck in einem Evaluierungsschritt . Beispiel:
`parts[1].setMimeType('application/octet-stream')`

Verwenden Sie zum Festlegen des Dateinamens des Anhangs einen MVE-Ausdruck Content-
Disposition-Header. Beispiel: `parts[1].setHeader('Content-Disposition',
'attachment; filename=payload.txt')`.

Konfigurieren Sie abschließend den Schritt für den E-Mail-Ausgang .

Nachrichtentext und Anhang für binäre Formate konfigurieren

Um eine binäre Datei im MediationContext zu injizieren, definieren Sie einen Abrufservice für den workday-
in-Transport, der die Datei von einem FTP- oder SFTP-Transport abrufen.

Erstellen Sie den Nachrichtentext und verwenden Sie dann einen Schritt eval, um die abgerufene Datei
in eine Variable zu kopieren. Sie können dazu die Dokumentzugriffsvariable da und die Methode toVar
verwenden. Beispiel: `da.toVar(Rising_2011_Logo.jpg', 'wd.retrieve.variable')`.

Verwenden Sie einen Kopierschritt , um die Datei aus zu kopieren `wd.retrieve.variable` an einen
Anhang Setzen Sie dazu die Eigenschaft „Input“ auf „Variable“ und geben Sie dann „Variable“ an
`wd.retrieve.variable` als Eingabevariable. Legen Sie die Eigenschaft Ausgabe auf *den Anhang* fest
und geben Sie für den Anhang für die Ausgabe einen Index von an 0 und einen Ausgabe -MIME-Typ *Bild/*
jpeg.

Konfigurieren Sie abschließend den Schritt für den E-Mail-Ausgang .

Untergeordnete Elemente des „HTTP-Out“-Transports

Konzept: Untergeordnete Elemente des „HTTP-Out“-Transports

Der Transport http-out unterstützt folgende untergeordnete Elemente:

Untergeordnetes Element	Beschreibung
http-basic-auth	Unterstützt die Standardauthentifizierung für Ihren Transport http-out.
http-custom-auth	Unterstützt die benutzerdefinierte Authentifizierung für Ihren Transport http-out.
https-properties	Unterstützt HTTPS-Eigenschaften für Ihren Transport http-out.

Referenz: Eigenschaften von „HTTP-Basic-Auth“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Kennwort	password	Ein Kennwort für die Authentifizierung.
Benutzername	username	Einen Benutzernamen für die Authentifizierung.
Realm	realm	Der von einer Callback-Handler-Bean verwendete Bereichsname.

Eigenschaft	XML-Attributname	Beschreibung
Callback-Handler Bean	callback-handler-bean	Eine Bean, die implementiert <code>javax.security.auth.callback.Callback</code> und verarbeitet die Callback-Art <code>com.capeclear.transport.impl.http.a</code>, um Benutzernamen und Kennwortdetails für die Authentifizierung anzugeben. Wenn Sie keine Kennwortdetails für ein untergeordnetes Element der Art "http-basic-auth" angeben, müssen Sie stattdessen eine Callback-Handler-Bean verwenden. Beispiel: <code><http-basic-auth username="me" callback-handler-bean="myCallBackHandlerBean"/></code>. Wenn Sie keine Details zu "Benutzername" und "Kennwort" für ein untergeordnetes Element "http-basic-auth" angeben, müssen Sie stattdessen eine Callback-Handler-Bean verwenden. Beispiel: <code><http-basic-auth callback-handler-bean="myCallBackHandlerBean"/></code>. Wenn Sie den Benutzernamen und das Kennwort für ein untergeordnetes Element der Art http-basic-auth angeben, benötigen Sie kein Callback-Handler-Bean. Beispiel: <code><http-basic-auth username="me" password="mypassword"/></code>.

Referenz: Eigenschaften von „HTTP-Custom-Auth“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Kennwort	password	Ein Kennwort für die Authentifizierung.
Benutzername	username	Einen Benutzernamen für die Authentifizierung.
Realm	realm	Der von einer Callback-Handler-Bean verwendete Bereichsname.

Eigenschaft	XML-Attributname	Beschreibung
Callback-Handler Bean	callback-handler-bean	Eine Bean, die implementiert <code>javax.security.auth.callback.Callback</code> und verarbeitet die Callback-Art <code>com.capeclear.transport.impl.http.a</code>, um Benutzernamen und Kennwortdetails für die Authentifizierung anzugeben. Wenn Sie keine Kennwortdetails für ein untergeordnetes Element der Art <code>http-custom-auth</code> angeben, müssen Sie stattdessen eine Callback-Handler-Bean verwenden. Beispiel: <code><http-custom-auth username="me" callback-handler-bean="myCallBackHandlerBean"/></code>. Wenn Sie keine Details zu Benutzername und Kennwort für ein untergeordnetes Element von <code>http-custom-auth</code> angeben, müssen Sie stattdessen eine Callback-Handler-Bean verwenden. Beispiel: <code><http-custom-auth callback-handler-bean="myCallBackHandlerBean"/></code>. Wenn Sie den Benutzernamen und das Kennwort für ein untergeordnetes Element der Art <code>http-custom-auth</code> angeben, benötigen Sie kein Callback-Handler-Bean. Beispiel: <code><http-custom-auth username="me" password="mypassword"/></code>.
Authenticator Bean	authenticator-bean	Die Quell-Bean, die eine benutzerdefinierte Authentifizierung durchführt.

Referenz: HTTPS-Eigenschaften

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Hosts akzeptieren	accept-hosts	Die Zeichenfolge des regulären Java-Ausdrucks, die für die Verifizierung von in SSL-Zertifikaten angegebenen

Eigenschaft	XML-Attributname	Beschreibung
		Hostnamen verwendet werden soll.
Kennwortschlüssel	key-password	Das Kennwort, das für den Zugriff auf den entsprechenden Schlüssel im Schlüsselspeicher verwendet werden soll.
Keystore-Datei	key-file	Der Speicherort der Schlüsselspeicher-Datei, die für die Verbindung mit einem HTTPS-Endpunkt verwendet werden soll.
Kennwort für Schlüsselspeicher	keystore-password	Das Kennwort für den Zugriff auf den Schlüsselspeicher.
Proxyhost	proxy-host	Der Hostname eines Proxyservers, den Sie als Proxy für HTTPS-Anforderungen für diesen Transport verwenden.
Proxy-Port	proxy-port	Der Port eines Proxyservers, den Sie als Proxy für HTTPS-Anforderungen für diesen Transport verwenden.
Truststore-Datei	truststore-file	Der Speicherort der Truststore-Datei, der beim Herstellen einer Verbindung mit einem HTTPS-Endpunkt verwendet werden soll.
Truststore-Kennwort	truststore-password	Das Kennwort für den Zugriff auf den Truststore.

Untergeordnete Elemente des „FTPS-Out“-Transports

Konzept: Untergeordnete Elemente des „FTPS-Out“-Transports

Der Transport ftps-out unterstützt folgende untergeordnete Elemente:

Untergeordnetes Element	Beschreibung	Anmerkungen
proxy-properties	Proxy-Authentifizierungseigenschaften.	
firewall-properties	Eigenschaften der Firewall-Verbindung	Der Transport ftps-out ist standardmäßig für Firewalls geeignet, da die Verbindung passiv ist. Einige Firewalls fordern jedoch möglicherweise dazu auf, alle Verbindungen durch sie zu leiten.

Untergeordnetes Element	Beschreibung	Anmerkungen
client-key-properties	Authentifizierungseigenschaften für privaten Schlüssel.	
server-key-properties	Authentifizierungseigenschaften für Serverschlüssel.	

Referenz: Eigenschaften der Proxy-Eigenschaften

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Proxyhost	proxy-host	Der Hostname des Proxyservers, der beim Herstellen einer Verbindung zur sicheren FTP-Site verwendet werden soll.
Proxykennwort	proxy-password	Das Kennwort des Proxyservers, das beim Herstellen einer Verbindung zur sicheren FTP-Site verwendet werden soll.
Proxy-Port	proxy-port	Die Portnummer des Proxyservers, die beim Herstellen einer Verbindung zur sicheren FTP-Site verwendet werden soll.
Proxyart	proxy-type	Die Proxyart des Proxyservers, die verwendet werden soll, wenn eine Verbindung zur sicheren FTP-Site hergestellt wird. Die Standardart ist <code>http</code> .
Proxybenutzername	proxy-username	Der Benutzername für den Proxyserver, der beim Herstellen einer Verbindung zur sicheren FTP-Site verwendet werden soll.

Referenz: Eigenschaften der Firewall-Eigenschaften

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Firewall-Host	firewall-host	Der Hostname oder die IP-Adresse der Firewall, die beim Herstellen einer Verbindung zur sicheren FTP-Site verwendet werden soll.
Firewall-Port	firewall-port	Die Portnummer der Firewall.

Referenz: Eigenschaften von „Client-Key“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Passwörter	passphrase	Die Passwörter des privaten Schlüssels, die bei der Authentifizierung mit der sicheren FTP-Site verwendet werden sollen.
Privater Schlüssel	private-key	Der private Schlüssel, der dem Benutzernamen für die SFTP-Authentifizierung zugeordnet ist. Sie können ihn auf zwei Arten angeben: - Als URI unter Verwendung des Anhangprotokolls und der Workday-ID. Beispiel: <code>attachment:privatekey/{WID}</code>, wobei WID ist eine 32-stellige Workday-ID. - Als relativen Pfadort. Beispiel: <code>private_key.txt</code>, wobei <code>private_key.txt</code> eine Datei ist, die sich im Ordner <code>ws/WSAR-INF</code> befindet.

Referenz: Eigenschaften von „Server-Key“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Passwörter	passphrase	Die Passwörter des privaten Schlüssels, die bei der Authentifizierung mit der sicheren FTP-Site verwendet werden sollen.
Privater Schlüssel	private-key	Der private Schlüssel, der dem Benutzernamen für die SFTP-Authentifizierung zugeordnet ist. Sie können ihn auf zwei Arten angeben: Als URI unter Verwendung des Anhangprotokolls und der Workday-ID. Beispiel: <code>attachment:privatekey/</code>

Eigenschaft	XML-Attributname	Beschreibung
		@{WID}, wobei WID ist eine 32-stellige Workday-ID. Als relativen Pfadort. Beispiel: <code>private_key.txt</code>, wobei <code>private_key.txt</code> eine Datei ist, die sich im Ordner <code>ws/WSAR-INF</code> befindet.

Untergeordnete Elemente von SFTP-Out

Konzept: Untergeordnete Elemente von SFTP-Out

Der Transport `sftp-out` unterstützt folgende untergeordnete Elemente:

Untergeordnetes Element	Beschreibung	Anmerkungen
<code>proxy-properties</code>	Proxy-Authentifizierungseigenschaften.	
<code>private-key-properties</code>	Authentifizierungseigenschaften mit privatem Schlüssel.	

Referenz: Eigenschaften der Proxy-Eigenschaften

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Proxyhost	<code>proxy-host</code>	Der Hostname des Proxyservers, der beim Herstellen einer Verbindung zur sicheren FTP-Site verwendet werden soll.
Proxykennwort	<code>proxy-password</code>	Das Kennwort des Proxyservers, das beim Herstellen einer Verbindung zur sicheren FTP-Site verwendet werden soll.
Proxy-Port	<code>proxy-port</code>	Die Portnummer des Proxyservers, die beim Herstellen einer Verbindung zur sicheren FTP-Site verwendet werden soll.
Proxyart	<code>proxy-type</code>	Die Proxyart des Proxyservers, die verwendet werden soll, wenn eine Verbindung zur sicheren FTP-Site hergestellt wird. Die Standardart ist <code>http</code> .

Eigenschaft	XML-Attributname	Beschreibung
Proxybenutzername	proxy-username	Der Benutzername für den Proxyserver, der beim Herstellen einer Verbindung zur sicheren FTP-Site verwendet werden soll.

Referenz: Eigenschaften des privaten Schlüssels

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Passwörter	passphrase	Die Passwörter des privaten Schlüssels, die bei der Authentifizierung mit der sicheren FTP-Site verwendet werden sollen.
Privater Schlüssel	private-key	Der private Schlüssel, der dem Benutzernamen für die SFTP-Authentifizierung zugeordnet ist. Sie können es als relativen Pfadort angeben. Beispiel: private_key.txt, wobei sich die Datei des privaten Schlüssels in befindet ws/WSAR-INF Ordner.

Komponenten

Konzept: Komponenten

Mit Komponenten können Sie Nachrichten bearbeiten und umwandeln, während sie Ihre Integration durchlaufen. Workday Studio bietet Unterstützung für diese Komponenten im Abschnitt Komponenten der Assembly Palette:

Mediationskomponente	Beschreibung	Anmerkungen
async-mediation	Unterstützt eine Verarbeitungskette, die Sie mit Schritten füllen können, um eine Anforderungsnachricht in Ihrer Assembly zu definieren.	
sync-mediation	Unterstützt zwei Verarbeitungsketten, die Sie mit Schritten füllen können, um Anforderungsnachrichten und Anforderungs-Antwort-Nachrichten in Ihrer Assembly zu definieren.	
custom-mediation	Erstellt eine individuell anpassbare	

Mediationskomponente	Beschreibung	Anmerkungen
	Komponente, die sowohl Anforderungsnachrichten als auch Anforderungs-Antwort-Nachrichten unterstützt.	
route	Implementiert Routing-Strategien und Unterrouen, um den Nachrichtenfluss durch Ihre Assembly zu leiten.	
splitter	Implementiert Splitter-Strategien und Unterrouen, um den Nachrichtenfluss durch Ihre Assembly zu leiten.	Gibt Signale an nachgelagerte aggregator-Komponenten aus.
aggregator	Verkettet Nachrichten zu Batches, die Sie an ein konfiguriertes Ziel weiterleiten können.	Erkennt vorgelagerte Splitter-Komponenten.

Referenz: Eigenschaften der Komponente „Async-Mediation“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der asynchronen Vermittlungskomponente innerhalb der Baugruppe.
Weiterleitung an	routes-to	Das Build-Element, an das die asynchrone Vermittlung die Nachricht weiterleitet.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob die Komponente für asynchrone Vermittlung ausgeführt wird.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Fortfahren nach Fehler	continue-after-error	Gibt an, wie sich Workday verhält, wenn ein Fehlerhandler einen Fehler löscht. Der Standardwert ist Rücklauf, d. h. die Verarbeitung wird ab der vorherigen Komponente in der Assembly fortgesetzt. Die Einstellung Wiederherstellen legt fest, dass Workday im Falle eines Fehlers zum nächsten

Eigenschaft	XML-Attributname	Beschreibung
		Schritt in der Mediation überspringt.
Nachgelagerte Fehler behandeln	handle-downstream-errors	Gibt an, ob nachgelagerte Fehler behandelt werden sollen. Wenn Sie diese Option auf Falsch festlegen, werden nur Fehler behandelt, die innerhalb der Komponente async-mediation auftreten. Globale Fehlerbehandlungen werden stattdessen nachgelagerte Fehler behandeln. Wenn diese Option auf Wahr festgelegt ist, können lokale Fehlerbehandlungen für die Komponente async-mediation nachgelagerte Fehler behandeln.

Referenz: Eigenschaften der Komponente „Sync-Mediation“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der Synchronisierung/Medien-Komponente innerhalb der Build.
Weiterleitung an	routes-to	Das Build-Element, an das die Nachricht bei der Synchronisierung/Mediation weitergeleitet wird.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortmeldung für die Komponente „Synchronisierung/Mediation“ .
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob die Komponente „Sync/Mediation“ ausgeführt wird.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Fortfahren nach Fehler	continue-after-error	Gibt an, wie sich Workday verhält, wenn ein Fehlerhandler einen Fehler löscht. Der Standardwert ist Rücklauf, d. h. die Verarbeitung wird ab

Eigenschaft	XML-Attributname	Beschreibung
		der vorherigen Komponente in der Assembly fortgesetzt. Die Einstellung Wiederherstellen legt fest, dass Workday im Falle eines Fehlers zum nächsten Schritt in der Mediation überspringt.
Nachgelagerte Fehler behandeln	handle-downstream-errors	Gibt an, ob nachgelagerte Fehler behandelt werden sollen. Wenn Sie diese Option auf Falsch festlegen, werden nur Fehler behandelt, die innerhalb der Komponente sync-mediation auftreten. Globale Fehlerbehandlungen werden stattdessen nachgelagerte Fehler behandeln. Wenn diese Option auf Wahr festgelegt ist, können lokale Fehlerbehandlungen für die Komponente sync-mediation nachgelagerte Fehler behandeln.

Referenz: Eigenschaften der Komponente „Custom-Mediation“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id		Die eindeutige ID der Komponente custom-mediation in der Assembly.
Weiterleitung an	routes-to	Das Build-Element, an das die benutzerdefinierte Vermittlung die Nachricht weiterleitet.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortnachricht für die benutzerdefinierte Vermittlungskomponente .
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob die benutzerdefinierte Vermittlungskomponente ausgeführt wird.
Quell Bean	ref	Die benutzerdefinierte springbean-Java-Klasse, die durch custom-mediationimplementiert wird.

Eigenschaft	XML-Attributname	Beschreibung
Anforderungsmethode	request-method	Der Name der Methode, die für die Anforderungsverarbeitung verwendet wird.
Antwortmethode	response-method	Der Name der Methode, die für die Antwortverarbeitung verwendet wird.

Referenz: Eigenschaften der Komponente „Route“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der Routing-Komponente innerhalb der Baugruppe.

Referenz: Eigenschaften der Komponente „Splitter“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der Split-Komponente innerhalb der Baugruppe.
Fehler: Keine Split-Nachricht	no-split-error-message	Gibt an, ob Workday einen Fehler auslöst, wenn die Split-Komponente keine Split-Nachricht für eine Masseneingabemeldung generiert.
Splitten bis	split-until	Gibt an, wann die Split-Komponente die Verarbeitung einer Masseneingabemeldung einstellt.

Referenz: Eigenschaften der Komponente „Aggregator“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der Aggregatorkomponente innerhalb der Baugruppe.
Weiterleitung an	routes-to	Das Build-Element, an das der Aggregator die Meldung weiterleitet.

Eigenschaft	XML-Attributname	Beschreibung
Sortieren bei	collate-when	Ein MVE-Ausdruck, der bestimmt, ob eingehende Nachrichten sortiert werden.
Batch erzwingen bei letzter Nachricht	force-batch-on-last-message	Erzwingt die Batch-Verarbeitung, sobald die Split-Komponente die letzte Nachricht erhält.
Batch erzwingen bei	force-batch-when	Ein MVE-Ausdruck, der bestimmt, ob eingehende Nachrichten in einem Batch erfasst werden.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Externe Kontrolle zulassen	allow-external-control	Ermöglicht es vorgelagerten Komponenten, zu steuern, wann die Aggregatorkomponente sortiert, in Batches verarbeitet oder beides.

Untergeordnete Elemente der Komponente „Route“

Konzept: Untergeordnete Elemente der Komponente „Route“

Die Komponente route unterstützt diese untergeordneten Elemente.

Untergeordnetes Element	Beschreibung	Anmerkungen
custom-strategy	Eine benutzerdefinierte Routingstrategie, die Sie als Spring-Bean konfigurieren.	
regex-strategy	Implementiert reguläre Ausdrücke, die Nachrichten an Unterrouen weiterleiten.	Implementiert untergeordnete choose-route-Elemente, um Ihre Ziel-Unterroute anzugeben.
mvel-strategy	Implementiert MVEL-Ausdrücke, die Nachrichten an Unterrouen weiterleiten.	Implementiert untergeordnete choose-route-Elemente, um Ihre Ziel-Unterroute anzugeben.
round-robin-strategy	Leitet Nachrichten in zyklischer Reihenfolge an Unterrouen weiter.	
xpath-strategy	Implementiert XPath-Ausdrücke, die Nachrichten an Unterrouen weiterleiten.	Implementiert untergeordnete choose-route-Elemente, um Ihre Ziel-Unterroute anzugeben.
failover-strategy	Implementiert einen Failover-Mechanismus, der Nachrichten an Unterrouen weiterleitet. Wenn ein untergeordnetes	

Untergeordnetes Element	Beschreibung	Anmerkungen
	Element einer sub-route fehlschlägt, wird es durch die nächste sub-route ersetzt.	
all-strategy	Leitet Nachrichten nacheinander an jede Unterroute weiter.	
doc-iterator	Iteriert über Dokumente, die Ihre Assembly von Workday abrufen.	
loop-strategy	Implementiert einen Schleifenmechanismus, der Nachrichten an Unterrouen weiterleitet.	
choose-route	Gibt das untergeordnete Element des sub-route-Ziels an, an das Ihre Strategie die Nachricht leitet.	Unterstützt XPath, MVEL oder reguläre Ausdrücke.
sub-route	Empfängt Nachrichten von Ihrer Routingstrategie und leitet sie an andere Komponenten und Transporte in Ihrer Assembly weiter.	

Referenz: Eigenschaften „Custom-Strategy“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Quell Bean	ref	Die spring bean, die Ihre benutzerdefinierte Strategieschnittstellenklasse implementiert.
Wiederholbar	repeatable	Ermöglicht Looping. Wenn diese Option auf Wahr festgelegt ist, implementiert Ihre Assembly eine Schleife, die wiederholt Ihre custom-strategy aufruft. Wenn diese Option auf Falsch gesetzt ist, implementiert Ihre Assembly keine Schleife für Ihre custom-strategy.
Wiederholungsgrenze	repeat-limit	Begrenzt, wie oft eine Schleife Ihre benutzerdefinierte Strategieaufruft.
Fehlerbehandlung	handle-error	Gibt an, ob Ihre übergeordnete Routing- Komponente Ausnahmen für Ihre Unter-

Eigenschaft	XML-Attributname	Beschreibung
		Routing-Komponente ""behandelt.

Referenz: Eigenschaften der Regex-Strategie

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Filter	filter	Gibt an, ob in Studio eine Fehlermeldung angezeigt wird, wenn Sie ungültige reguläre Ausdrücke in einem untergeordneten Element der Art "Choice-Routing" implementieren.

Referenz: Eigenschaften der MVEL-Strategie

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Filter	filter	Gibt an, ob in Studio eine Fehlermeldung angezeigt wird, wenn Sie ungültige MVE-Ausdrücke in einem untergeordneten Element der Art "Choice-Routing" implementieren.

Referenz: Eigenschaften der Round-Robin-Strategie

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Klonanforderung	clone-request	Gibt an, ob die round-Robin-Strategie eine Kopie der ursprünglichen Vermittlungsnachricht an jedes Ihrer untergeordneten Routing-Elemente sendet.
Failover	failover	Gibt an, ob die round-Robin-Strategie eine Schleife durch Ihre untergeordneten Elemente des Unter-Routings durchläuft, wenn sie ein fehlgeschlagenes Unter-Routing ""findet.
Timeout	timeout	Ein Millisekundenwert, der angibt, wie lange ein fehlgeschlagener Unterrouuten

Eigenschaft	XML-Attributname	Beschreibung
		vom Routing ausgeschlossen wird.

Referenz: Eigenschaften der XPath-Strategie

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Filter	filter	Gibt an, ob in Studio eine Fehlermeldung angezeigt wird, wenn Sie ungültige XPath-Ausdrücke in einem untergeordneten Element der Art "Choice-Routing" implementieren.
Namespaces	namespaces	Die von Ihrem XPath-Ausdruck verwendeten Namespaces im Format <code>prefix namespace</code> <code>prefix2 namespace2</code> . Überschreibt Namespace-Mappings in der Datei <code>Asset.xml</code> .
XPath-Version	xpath-version	Gibt an, ob XPath Version 1.0, 2.0 oder 3.0 verwendet werden soll. Geben Sie bis Workday 11 den Wert 1 für Assembly-Versionen ein. Geben Sie die Werte 2 oder 3 für Assembly-Versionen ab Workday 12 ein. Sie können in Workday die Version dynamisch festlegen, indem Sie eine MVAL-Vorlage verwenden, die einen Systemeigenschaftswert einfügt. Beispiel: <code>@{props.get('the_version')}</code>).

Referenz: Eigenschaften der Failover-Strategie

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Fehler löschen	clear-fault	Gibt an, ob die Failover-Strategie Ausnahmen löscht, die von Ihren untergeordneten Elementen der Weiterleitung generiert wurden.
Klonanforderung	clone-request	Gibt an, ob die Failover-Strategie eine Kopie der ursprünglichen

Eigenschaft	XML-Attributname	Beschreibung
		Vermittlungsmeldung an jedes Ihrer untergeordneten Routing-Elemente sendet.

Referenz: Eigenschaften für alle Strategien

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Klonanforderung	clone-request	Gibt an, ob die Strategie „Alle“ eine Kopie der ursprünglichen Vermittlungsnachricht an jedes Ihrer untergeordneten Routing-Elemente sendet.
Fehler ignorieren	ignore-errors	Gibt an, ob die Strategie „Alle“ Fehler ignoriert, die von Ihren untergeordneten Elementen der Weiterleitung generiert werden.

Referenz: Eigenschaften des „Doc-Iterator“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Variablenname	variable-name	Der Name der Variable, in der der Dokument-Iterator Ihr abgerufenes Dokument speichert. Der Standardwert ist <code>wd.retrieve.variable</code> .
Labels	labels	Gibt ein oder mehrere Labels an, die an Ihr abgerufenes Dokument angehängt werden.
Sortieren nach	sort-by	Die Sortierreihenfolge für das Abrufen und Präsentieren von Dokumenten. Folgende Optionen sind verfügbar: • FILENAME_ASCENDING • FILENAME_DESCENDING • NONE

Referenz: Eigenschaften der Schleifenstrategie

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Bedingung	condition	Ein MVE-Ausdruck, der einen booleschen Wert generiert. Die Schleifenstrategie wird

Eigenschaft	XML-Attributname	Beschreibung
		ausgeführt, bis der Ausdruck als falsch evaluiert wird.
Schritt	increment	Ein Inkrementwert für jede Schleifesiteration.
Init	init	Der Startwert Ihrer Schleife.
Wiederholungsgrenze	repeat-limit	Die maximale Anzahl von Schleifen, die durch die Loop-Strategyimplementiert werden. Eine Routing-Strategie kann insgesamt bis zu 200.000-mal wiederholt werden. Wenn dieselbe Strategie mehrmals aufgerufen wird, wird der Schleifenzähler nicht zurückgesetzt. Sie wird ab dem vorherigen Wert fortgesetzt. Sie können eine benutzerdefinierte Java-Bean oder einen benutzerdefinierten Splitter verwenden, um diesen Grenzwert zu vermeiden.

Referenz: Eigenschaften „Choose-Route“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Ausdruck	expression	Ein Ausdruck, mit dem Werte eingehender Nachrichten abgeglichen werden.
Route	route	Ein untergeordnetes Ziel -Unter-Routing -Element für übereinstimmende Ausdruckswerte.

Referenz: Eigenschaften „Sub-Route“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Name	name	Der eindeutige Name der Unterroute innerhalb der Gruppe.
Weiterleitung an	routes-to	Das Build-Element, an das die Nachricht von der Unter-Weiterleitung weitergeleitet wird.

Untergeordnete Elemente der Komponente „Splitter“

Konzept: Untergeordnete Elemente der Komponente „Splitter“

Die Komponente splitter unterstützt diese untergeordneten Elemente.

Untergeordnetes Element	Beschreibung	Anmerkungen
custom-splitter	Eine benutzerdefinierte Splitterstrategie, die Sie als Spring Bean konfigurieren.	
standard-splitter	Extrahiert Daten aus Massennachrichten, splittet die Daten in Tokens auf und leitet die tokenisierten Daten an Ihre Ziel-Unterrouen weiter.	Es werden folgende untergeordnete Elemente unterstützt: • header-ends-with • header-fix-lines • content-fixed-lines • content-starts-ends • content-starts-with • footer-fixed-lines • footer-starts-with
xpath-splitter	Implementiert einen XPath-Ausdruck, um Daten aus eingehenden Massennachrichten zu extrahieren.	
xml-stream-splitter	Implementiert einen XPath-Ausdruck, um Daten aus eingehenden Massennachrichten zu extrahieren. Unterstützt XML-Streaming.	
json-splitter	Extrahiert JSON aus Masseneingabenachrichten.	
unzip-splitter	Extrahiert Daten aus ZIP- und TAR-Dateien, teilt die Daten in einzelne Nachrichten auf und leitet die Nachrichten an ihre Ziel-Unterrouen weiter.	
Tabelle-Splitter	Extrahiert Daten aus einer <code>mtable</code> Objekt und teilt die Daten in einzelne Zeilen auf.	
sub-route	Empfängt Nachrichten von Ihrer Splitterstrategie und leitet sie an andere Komponenten und Transporte in Ihrer Assembly weiter.	

Referenz: Eigenschaften des Custom-Collater

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Quell Bean	ref	Die spring bean, die Ihre benutzerdefinierte Split- Schnittstellenklasse implementiert.

Referenz: Eigenschaften des Standard-Splitter

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Ausgabe-MIME-Typ	output-mimetype	MIME-Typ Ihrer tokenisierten Ausgabedaten.
Regexp für Zeilen-Token-Trennzeichen	line-token-delimiter-regexp	Der begrenzende reguläre Ausdruck, mit dem tokenisierte Daten aus Massennachrichten extrahiert werden.
Trennzeichen für mehrere Zeilen	multiple-lines-separator	Eine Zeichenfolge, die Token über mehrere Zeilen hinweg verkettet.

Referenz: Eigenschaften des XPath-Splitters

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Namespaces	namespaces	Die von Ihrem XPath-Ausdruck verwendeten Namespaces.
XPath	xpath	Der XPath-Ausdruck, der bestimmt, wo die Eingabemeldungen gesplittet werden.
XPath Version	xpath-version	Gibt an, ob XPath Version 1.0, 2.0 oder 3.0 verwendet werden soll. Geben Sie bis Workday 11 den Wert 1 für Assembly-Versionen ein. Geben Sie die Werte 2 oder 3 für Assembly-Versionen ab Workday 12 ein. Sie können festlegen, dass Workday die Version dynamisch festlegen kann, indem Sie eine MVAL-Vorlage verwenden, die einen Eigenschaftswert

Eigenschaft	XML-Attributname	Beschreibung
		einfügt. Beispiel: <code>@{props.get('the_version')}</code> }).

Referenz: Eigenschaften des XML-Stream-Splitter

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Namespaces	namespaces	Die von Ihrem XPath-Ausdruck verwendeten Namespaces.
XPath	xpath	Der XPath-Ausdruck, der XML aus Masseneingabemeldungen extrahiert.
XPath-Version	xpath-version	Gibt an, ob XPath Version 1.0, 2.0 oder 3.0 verwendet werden soll. Geben Sie bis Workday 11 den Wert 1 für Assembly-Versionen ein. Geben Sie die Werte 2 oder 3 für Assembly-Versionen ab Workday 12 ein. Sie können in Workday die Version dynamisch festlegen, indem Sie eine MVAL-Vorlage verwenden, die einen Systemeigenschaftswert einfügt. Beispiel: <code>@{props.get('the_version')}</code> }).

Referenz: Eigenschaften des JSON-Splitters

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Json-Pfad	json-path	Der Ausdruck, mit dem JSON aus Masseneingabemeldungen extrahiert wird.

Referenz: Eigenschaften des Unzip-Splitters

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Dateimuster	file-pattern	Der reguläre Ausdruck, der Daten aus ZIP- und TAR-Eingabedateien extrahiert.
Format	format	Das Format Ihrer Eingabedateien. Folgende Optionen sind verfügbar:

Eigenschaft	XML-Attributname	Beschreibung
		• zip • tar

Referenz: Eigenschaften des Mtable-Splitter

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Mz-Tabellenname	mtable-name	Der Name der MediationContext Eigenschaft, die den enthält mtable.
Zeilenname	row-name	Der Name der MediationContext - Eigenschaft, in der die iterierte Zeile gespeichert wird.
Zeilenformat	row-format	Bestimmt die Klasse der iterierten Zeile. Unterstützte Werte sind: • array • map

Untergeordnete Elemente des Standard-Splitters

Konzept: Untergeordnete Elemente des Standard-Splitters

Die Komponente standard-splitter unterstützt diese untergeordneten Elemente.

Untergeordnetes Element	Beschreibung
header-ends-with	Gibt das Ende des Header-Abschnitts in Ihrem Dokument an.
header-fix-lines	Gibt die Anzahl der Zeilen an, die der Header-Abschnitt in Ihrem Dokument einnimmt.
content-fixed-lines	Gibt die Anzahl der Zeilen an, die der Abschnitt mit dem Inhalt der Datensätze in Ihrem Dokument einnimmt.
content-starts-ends	Gibt den Start- und Endpunkt des Abschnitts mit dem Inhalt der Datensätze in Ihrem Dokument an.
content-starts-with	Gibt den Startpunkt des Abschnitts mit dem Inhalt der Datensätze in Ihrem Dokument an.
footer-fixed-lines	Gibt die Anzahl der Zeilen an, die der Fußzeilenabschnitt in Ihrem Dokument einnimmt.
footer-starts-with	Gibt den Startpunkt des Fußzeilenabschnitts in Ihrem Dokument an.

Referenz: Eigenschaften für „Header-Ends-With“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
In Header einschließen	include-in-header	Bestimmt, ob die übereinstimmende Zeile Ihres regulären Ausdrucks Teil des Headers ist.
Regex	regex	Ein regulärer Ausdruck, der das Ende des Headers angibt.

Referenz: Eigenschaften von „Header-Fixed-Lines“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Anzahl Positionen	lines-count	Die Anzahl der vom Header besetzten Zeilen.

Referenz: Eigenschaften von „Content-Fixed-Lines“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Reg Ausdruck für Positionen ignorieren	ignore-lines-regex	Ein regulärer Ausdruck, der übereinstimmende Positionen aus dem Inhaltsabschnitt des Datensatzes ausschließt.
Anzahl Positionen	lines-count	Die Anzahl der Zeilen, die vom Abschnitt für den Inhalt des Datensatzes besetzt werden.

Referenz: Eigenschaften von „Content-Starts-Ends“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Regex beenden	end-regex	Ein regulärer Ausdruck, der das Ende des Abschnitts „Inhalt von Datensätzen“ identifiziert.
Starten Sie den regulären Ausdruck	start-regex	Ein regulärer Ausdruck, der den Beginn des Abschnitts „Inhalt von Datensätzen“ angibt.

Referenz: Eigenschaften von „Content-Starts-With“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Regex	regex	Ein regulärer Ausdruck, der den Beginn des Abschnitts „Inhalt von Datensätzen“ angibt.

Referenz: Eigenschaften von „Footer-Fixed-Lines“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Reg Ausdruck für Positionen ignorieren	ignore-lines-regex	Ein regulärer Ausdruck, der übereinstimmende Positionen aus dem Footer-Abschnitt ausschließt.
Anzahl Positionen	lines-count	Die Anzahl der Zeilen, die vom Footer-Abschnitt eingenommen werden.
Wenn Sie Positionen ignorieren, bereinigen Sie sie	trim-eof-ignore-lines	Bestimmt, ob die übereinstimmenden Zeilen Ihres regulären Ausdrucks aus der Fußzeile entfernt werden.

Referenz: Eigenschaften von „Footer-Starts-With“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Regex	regex	Ein regulärer Ausdruck, der den Beginn des Abschnitts „Inhalt von Datensätzen“ angibt.

Untergeordnete Elemente der Komponente „Aggregator“

Konzept: Untergeordnete Elemente der Komponente „Aggregator“

Die Komponente aggregator unterstützt diese untergeordneten Elemente.

Untergeordnetes Element	Beschreibung
custom-batch-strategy	Eine benutzerdefinierte Batchingstrategie, die Sie als Spring-Bean konfigurieren.
size-batch-strategy	Stapelt eingehende Nachrichten basierend auf einem angegebenen Wert für die Batch-Größe als Batch auf.
time-batch-strategy	Stapelt eingehende Nachrichten basierend auf einem angegebenen Zeitperiodenwert als Batch auf.

Untergeordnetes Element	Beschreibung
custom-collater	Eine benutzerdefinierte Sortierstrategie, die Sie als Spring Bean konfigurieren.
message-content-collater	Aggregiert den Nachrichteninhalt von <code>MediationMessage</code> .
xml-message-content-collater	Aggregiert den XML-Dokumentinhalt eines XML-Message-Streams.
json-collater	Aggregiert JSON-Dokumente in eine einzigen Ausgabe.
zip-file-collater	Aggregiert mehrere Dateien in eine einzige ZIP-Datei.
mtable-collater	Aggregate <code>mtable</code> Zeilen in eine neue <code>mtable</code> Instanz.

Referenz: Eigenschaften der „Custom-Batch-Strategy“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Quell Bean	<code>ref</code>	Die spring bean, die Ihre benutzerdefinierte-Batch-Strategy- Schnittstellenklasse implementiert.

Referenz: Eigenschaften der „Size-Batch-Strategy“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Batch-Größe	<code>batch-size</code>	Die maximale Anzahl von Nachrichten, die ein Batch verkettet.

Referenz: Eigenschaften der „Time-Batch-Strategy“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Zeitperiode	<code>time-period</code>	Dauer der Batch-Aggregation.
Zeiteinheit	<code>time-unit</code>	Die durch die Eigenschaft „Zeitperiode“ implementierte Zeiteinheit. Der Standardwert ist Sekunden.

Referenz: Eigenschaften des Custom-Collaters

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Ausgabe	output	Gibt an, wohin das untergeordnete Element der benutzerdefinierten Sortierung die Ausgabe leitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Quell Bean	ref	Die spring bean, die Ihre benutzerdefinierte Schnittstellenklasse für Sortierung implementiert.

Referenz: Eigenschaften des Message-Content-Collater

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Ausgabe	output	Gibt an, wohin das untergeordnete Element message-content-collater

Eigenschaft	XML-Attributname	Beschreibung
		die Ausgabe leitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Geben Sie den Index des Anhangs an. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Header-Text	header - text	Fügt Ihrer Ausgabenachricht einen Header hinzu.
Footer-Text		Fügt eine Fußzeile zu Ihrer Ausgabenachricht hinzu.
Trennzeichen	separator	Eine Zeichenfolge, die beim Verketteten von Nachrichten in einem Batch verwendet wird.

Referenz: Eigenschaften des XML-Message-Content-Collater

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Ausgabe	output	Gibt an, wohin das untergeordnete Element xml-message-content-collater die Ausgabe leitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich

Eigenschaft	XML-Attributname	Beschreibung
		rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Geben Sie den Index des Anhangs an. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Header-Text	header - text	Fügt Ihrer Ausgabenachricht einen Header hinzu.
Footer-Text		Fügt eine Fußzeile zu Ihrer Ausgabenachricht hinzu.
Trennzeichen	separator	Eine Zeichenfolge, die beim Verketteten von Nachrichten in einem Batch verwendet wird.
Namespaces	namespaces	Die von Ihrem XPath-Ausdruck verwendeten Namespaces.
XPath	xpath	Ein XPath-Ausdruck, der übereinstimmende XML-Meldungen aggregiert. Wenn nicht angegeben, aggregiert der xml-message-content-collater den gesamten XML-Nachrichteninhalt in einer Ausgabedatei.

Referenz: Eigenschaften des JSON-Aggregators

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Ausgabe	output	Gibt an, wohin das untergeordnete Element „json-collater“ die Ausgabe leitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Geben Sie den Index des Anhangs an. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.

Referenz: Eigenschaften des Zip-File-Collater

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Ausgabe	output	Gibt an, wohin das untergeordnete Element zip-file-collater die Ausgabe leitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch

Eigenschaft	XML-Attributname	Beschreibung
		alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Format	format	Das Format Ihrer Ausgabedateien. Folgende Optionen sind verfügbar: • zip • tar
Name der Nachrichtenentität	message-entity-name	Der Name jeder Nachricht, die zu Ihrer Ausgabedatei hinzugefügt wurde.

Referenz: Eigenschaften des Mtable-Collater

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Ausgabe	output	Workday ignoriert diese Einstellung für untergeordnete Elemente der Art "mtable-collater" .
Mz-Tabellenname	mtable-name	Der Name der MediationContext - Eigenschaft, in der Ihre Ausgabe gespeichert ist mtable.
Zeilenname	row-name	Der Name der MediationContext - Eigenschaft, in der Ihre

Eigenschaft	XML-Attributname	Beschreibung
		aggregierten Daten gespeichert sind <code>mtable</code> Zeile.
Spaltennamen	<code>column-names</code>	Eine durch Kommas getrennte Liste von Spalten, die Ihre Ausgabe definiert <code>mtable</code> .
Indexspalte	<code>index-column</code>	Der Name der Indexspalte Ihrer Ausgabe <code>mtable</code> .
Nullzeile zulassen	<code>allow-null-row</code>	Bestimmt, ob Ihre Ausgabe <code>mtable</code> unterstützt Zeilenwerte für 0.

Allgemeine Komponenten

Konzept: Allgemeine Komponenten

Workday Studio bietet eine Reihe vorkonfigurierter Subassemblies, mit denen Sie Ihre Integrationen erstellen können. Diese Untergruppen sind auf dem Register Allgemeine Komponenten der Palette verfügbar. Wenn Sie eins zu Ihrem Baugruppendiagramm hinzufügen, fügt Studio eins ein `local-out`-Element in der XML-Quelle mit dem entsprechenden Endpunkt in Workday.

Außerdem werden die Parameter angezeigt, die Sie mit dieser Untergruppe verwenden können, wobei die erforderlichen Parameter bereits ausgewählt sind. Sie können Parameter jederzeit in der Ansicht Eigenschaften der Subassembly anpassen.

Studio enthält folgende Allgemeine Komponenten in der Palette:

Name	Funktion	Anmerkungen
ftp	Leitet zu einem ftp-out-Transport. weiter.	- Bietet einen zentralen Eingabepunkt für alle FTP-Transporte. - Verwendet die <code>wd.ftp.endpoint</code> Parameterwert, um die Art des implementierten FTP-Transports zu bestimmen. Beispiele: <code>ftp://</code>, <code>sftp://</code>, oder <code>ftps://</code>. - Legt die Transporteinstellungen offen, die in Workday häufig verwendet werden, z. B. Authentifizierung mit Benutzername und Kennwort. Workday blendet die erweiterte Authentifizierung mit öffentlichen Schlüsseln aus. Anmerkung: Die FTP-Untergruppe ist in der Studio

Name	Funktion	Anmerkungen
		Palette nicht mehr verfügbar. In neuen Integrationen sollten Sie die Komponenten ftp-out oder sftp-out verwenden, um Dokumente an einen FTP-Server zu senden. Diese Komponenten bieten mehr Sicherheit, Performance und Zuverlässigkeit. Sie sind auch mit späteren Versionen von JSCASE kompatibel. Die FTP-Untergruppe bleibt in vorhandenen Integrationen funktionsfähig, aber Workday empfiehlt, sie sofern praktikabel zu ersetzen.
GetEventConfigurations	Ruft die Servicekonfigurationen ab, die an ein bestimmtes Integrationsereignis angehängt sind.	
GetEventDocuments	Ruft die Dokumente ab, die an ein bestimmtes Integrationsereignis angehängt sind.	
GetIntegrationEvent	Ruft ein Integrationsereignis ab, das auf einer Workday-Referenz-ID basiert.	
GetIntegrationSystems	Ruft die Konfiguration des Integrationssystems aus Workday ab und parst sie.	
PagedGet	Implementiert die für alle erforderliche Paging-Logik paged-get Webservicevorgang, der den seitenspezifischen Datenabruf ermöglicht.	• Verwenden Sie die <code>is.paged.get.process.endpoint</code>-Parameter, um jede Seite mit Antwortdaten zur Verarbeitung an eine andere Untergruppe zu senden. • Verwenden Sie die <code>is.paged.get.aggregate.header</code>, <code>is.paged.get.aggregate.footer</code> und <code>is.paged.get.aggregate.xpath</code>-Parameter, um alle Seiten mit Antwortdaten zu aggregieren. • Aggregierte XML-Daten sind als Meldung in verfügbar MediationContext wenn die Untergruppe zurückgegeben wird.

Name	Funktion	Anmerkungen
		Wenn von einem Webservice erwartet wird, dass er eine große Anzahl von Seiten zurückgibt, verwenden Sie den Ansatz der Verarbeitung pro Seite.
PagedGetLocalPaging	Implementiert die für alle erforderliche Paging-Logik paged-get Lokaler Webservicevorgang, der den seitenspezifischen Datenabruf ermöglicht.	- Ermöglicht die lokale Ausführung des Paging-Prozesses auf dem ESB-Server statt der Remote-Arbeit durch Workday. - Verwendet eine einzelne Abfrage, um alle Workday-IDs zurückzugeben. Bei nachfolgenden Aufrufen werden die Daten für jede Workday-ID abgerufen.
PdfPrintStep	Konvertiert benutzerdefinierte Berichtsdaten aus einer workday-out-rest-Anforderung in PDF-Format mithilfe eines BIRT-Berichtsdesigns	Geben Sie für jeden der Mpunkte-fähigen Parameter einen Literalzeichenfolgenwert an, indem Sie ihn in einfache Anführungszeichen setzen.
PutIntegrationEvent	Aktualisiert den Status des Integrationsereignisses eines Integrationssystems in Workday.	- Verwendet die Put_Integration_Event Webservicevorgang - Fügen Sie die vom Webservicevorgang zurückgegebene Workday-ID für die Ereignisantwort in ein MediationContext Eigenschaft <code>integrations.event.response.wi</code>
PutIntegrationMessage	Sendet Nachrichten zur Aktualisierung der Integration an Workday.	- Sie können auch Dokumente in Form von Referenzen zum Workday-Dokumentspeicher hinzufügen. - Verwendet die Put_Integration_Message Webservicevorgang - Workday beschränkt die Anzahl der Aufrufe, die eine Untergruppe von PutIntegrationMessage in einer einzelnen

Name	Funktion	Anmerkungen
		Integration ermöglichen kann, auf 500. Nachfolgende Integrationsmeldungen werden in einer Protokolldatei protokolliert und können nicht im Integrationereignis angezeigt werden. (Die Umleitung gilt nicht für Nachrichten, die Anhänge haben oder den Status des Integrationereignisses ändern.)
SalesforceConnector	Sendet authentifizierte Nachrichten an Salesforce.com.	
PrismConnector	Sendet CSV-Daten an Prism Analytics.	

Referenz: Eigenschaften der FTP-Subassembly

Register „Allgemein“

Anmerkung: Die FTP-Untergruppe ist in der Studio Palette nicht mehr verfügbar. In neuen Integrationen sollten Sie die Komponenten ftp-out oder sftp-out verwenden, um Dokumente an einen FTP-Server zu senden. Diese Komponenten bieten mehr Sicherheit, Performance und Zuverlässigkeit. Sie sind auch mit späteren Versionen von JSCASE kompatibel. Die FTP-Untergruppe bleibt in vorhandenen Integrationen funktionsfähig, aber Workday empfiehlt, sie sofern praktikabel zu ersetzen.

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der FTP-Untergruppe innerhalb der Gruppe.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortmeldung für die FTP- Untergruppe.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob die FTP- Untergruppe ausgeführt wird.
Endpunkt	endpoint	Die Adresse des Zielendpunkts für die Untergruppe. Die Ftp - Komponente unterstützt: vm: // Endpunktadresse Beispiel: vm: //wcc/Ftp.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie " <i>Kein(e)</i> " an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um die WS-Adressierungsdetails anzugeben.
Klonanforderung	clone-request	Gibt an, ob die Verarbeitung zwischen Untergruppen verkettet werden soll. Setzen Sie die Eigenschaft auf Wahr, um eine Kopie der ursprünglichen Nachricht zu erstellen, sodass Subassemblies, die die Anforderungsnachricht ändern, keine Auswirkungen auf die anderen haben. Setzen Sie die Eigenschaft auf Falsch, um die ursprüngliche Nachricht zu übermitteln.
Abbruch weitergeben	propagate-abort	Gibt an, ob ein Abbruchkennzeichen von einer Untergruppe an die aufrufende Baugruppe weitergegeben wird. Eine Untergruppe stoppt die Verarbeitung, wenn sie als markiert MediationContext als abgebrochen.

Eigenschaft	XML-Attributname	Beschreibung
Nicht konfigurierte Eigenschaften	unset-properties	Gibt an, ob alle lokalen Ende - Eigenschaften zurückgesetzt werden, nachdem der lokale Ende- Transport eine Untergruppe aufgerufen hat.

Register „Parameter“

Parameter	Beschreibung
<code>wd.ftp.endpoint</code>	Das Endpunktschema, mit dem bestimmt wird, welcher FTP-Ausgangstransport verwendet werden soll. Der Endpunkt muss mit beginnen <code>ftp://</code> , <code>sftp://</code> , oder <code>ftps://</code> .
<code>wd.ftp.username</code>	Der Benutzername, der für den Zugriff auf den FTP-Server verwendet wird.
<code>wd.ftp.password</code>	Das Kennwort, das dem Benutzernamen für den Zugriff auf den FTP-Server zugeordnet ist.
<code>wd.ftp.method</code>	Die Methode oder Meldungsaktion, die die FTP-Untergruppe implementiert, wenn eine Verbindung zum Endpunkt hergestellt wird. Schließen Sie jede Aktion in einfache Anführungszeichen ein. Gültige Optionen: • <code>'put '</code>: Die FTP-Komponente sendet die Nachricht an den FTP-Server und die Nachricht bleibt unverändert. • <code>'get '</code>: Die Build ersetzt den Root-Teil der Nachricht durch die Daten, die sie vom FTP-Server erhält. • <code>'list '</code>: Die Meldung bleibt unverändert. Workday fügt eine Kontext-Eigenschaft mit dem Namen hinzu <code>wd.ftp.files.list</code>, die Folgendes enthält: <code>List<String></code> - Objektinstanz Anmerkung: Mit der <code>'list '</code> Methode mit einem Dateimuster von <code>*.txt</code> oder <code>*.xml</code> führt zu einer Ausnahme. • <code>'delete '</code>: Workday fordert das Löschen der Datei vom FTP-Server an. Die Meldung bleibt unverändert. • <code>'mget '</code> (Multiget): Workday lädt alle Dateien herunter, die mit dem Eingabemuster übereinstimmen, und ersetzt den Root-Teil der Nachricht durch eine ZIP-Datei, die diese Dateien enthält. • <code>'mdelete '</code> (Mehrfachlöschung): Workday löscht alle Dateien, die mit dem Eingabemuster übereinstimmen. Die Meldung bleibt unverändert.

Parameter	Beschreibung
<code>wd.ftp.passive.mode</code>	Gibt an, ob während der FTP-Verbindung der passive Modus verwendet wird.
<code>wd.ftp.file.pattern</code>	Gibt das Dateimuster an, das beim Durchführen des FTP-Vorgangs verwendet wird. Beispiele: • <code>*.xml</code> • <code>*.txt</code> Für die meisten anderen Methoden als <code>list</code> ausführen, unterstützt Workday Wildcards und führt die Filterung auf Client-Seite durch. Für den <code>list</code>-Methode verwendet Workday eine serverseitige Filterung, die auf der Syntax regulärer Ausdrücke basiert. Beispiele: • <code>wd.ftp.method</code> ist <code>'get'</code> und <code>wd.ftp.file.pattern</code> ist <code>*.xml</code>: Workday ruft die erste XML-Datei ab, die auf dem FTP-Server im angegebenen Verzeichnis gefunden wird. • <code>wd.ftp.method</code> ist <code>'mget'</code> und <code>wd.ftp.file.pattern</code> ist <code>*.xml</code>: Workday ruft alle auf dem FTP-Server im angegebenen Verzeichnis gefundenen XML-Dateien im ZIP-Format ab. • <code>wd.ftp.method</code> ist <code>'list'</code> und <code>wd.ftp.file.pattern</code> ist <code>^.*\.(xml)\$</code>: Workday listet alle auf dem FTP-Server im angegebenen Verzeichnis gefundenen XML-Dateien auf.
<code>wd.ftp.temp.file.name</code>	Gibt an, ob bei einem FTP eine temporäre Datei verwendet werden soll put Nachrichtenaktion.
<code>wd.ftp.timeout</code>	Ein Timeout für die Nachrichtenaktion in Millisekunden.
<code>wd.ftp.directory</code>	Der relative oder vollständig qualifizierte Verzeichnispfad.
<code>wd.ftp.private.key</code>	Der private Schlüssel, der dem Benutzernamen für die SFTP-Authentifizierung zugeordnet ist. Sie wurde mithilfe des Anhangprotokolls und einer Workday-ID als URI formatiert. Beispiel: <code>attachment:privatekey/{WID}</code> , wobei die WID einer 32-stelligen Workday-ID entspricht.
<code>wd.ftp.max.attempts</code>	Die maximale Anzahl der Versuche, auf den FTP-Server zuzugreifen.
<code>wd.ftp.debug</code>	Gibt an, ob Workday die Debug-Protokolle zu den Protokollen hinzufügt.
<code>wd.ftp.proxy.host</code>	Der Hostname des FTP-Proxy-Servers.
<code>wd.ftp.proxy.port</code>	Der Port des FTP-Proxy-Servers.

Parameter	Beschreibung
<code>wd.ftp.proxy.username</code>	Der Benutzername für den Zugriff auf den FTP-Proxy-Server.
<code>wd.ftp.proxy.password</code>	Das Kennwort des FTP-Proxyservers.
<code>wd.ftp.proxy.type</code>	Die Art des Proxyservers.
<code>wd.ftp.input.content.type</code>	Gibt den Inhaltstyp von Dateien an, die über FTP abgerufen werden get. Beispiel: <code>text/plain; charset=Big5</code> .

Referenz: Eigenschaften der Subassembly „GetEventConfigurations“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	<code>id</code>	Die eindeutige ID der Untergruppe „GetEventConfigurations“ innerhalb der Baugruppe.
Antwort weiterleiten an	<code>routes-response-to</code>	Das Ziel der Antwortmeldung für die Untergruppe „GetEventConfigurations“.
Ausführung bei	<code>execute-when</code>	Eine Bedingung, die bestimmt, ob die Untergruppe „GetEventConfigurations“ ausgeführt wird.
Endpunkt	<code>endpoint</code>	Gibt die Zieledpunktadresse für die Untergruppe an. Die Komponente "GetEventConfigurations" unterstützt Folgendes: <code>vm:// Endpunktadresse</code> Beispiel: <code>vm://wcc/GetEventConfigurations</code> .

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	<code>store-message</code>	Geben Sie " <i>Kein(e)</i> " an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	<code>transport-class</code>	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global

Eigenschaft	XML-Attributname	Beschreibung
		ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um die WS-Adressierungsdetails anzugeben.
Klonanforderung	clone-request	Gibt an, ob die Verarbeitung zwischen Untergruppen verkettet werden soll. Setzen Sie die Eigenschaft auf Wahr, um eine Kopie der ursprünglichen Nachricht zu erstellen, sodass Subassemblies, die die Anforderungsnachricht ändern, keine Auswirkungen auf die anderen haben. Setzen Sie die Eigenschaft auf Falsch, um die ursprüngliche Nachricht zu übermitteln.
Abbruch weitergeben	propagate-abort	Gibt an, ob ein Abbruchkennzeichen von einer Untergruppe an die aufrufende Baugruppe weitergegeben wird. Eine Untergruppe stoppt die Verarbeitung, wenn sie als markiert MediationContext als abgebrochen.
Nicht konfigurierte Eigenschaften	unset-properties	Gibt an, ob alle lokalen Ende - Eigenschaften zurückgesetzt werden, nachdem der lokale Ende- Transport eine Untergruppe aufgerufen hat.

Register „Parameter“

Parameter	Beschreibung
ie.event.wid	Die Workday-ID für das Integrationsereignis, dessen Servicekonfigurationen Sie abrufen möchten. Wird automatisch

Parameter	Beschreibung
	mit ausgefüllt <code>lp.isSet()</code> ? <code>lp.getIntegrationEventWID()</code> : null.
<code>is.event.disable.msg.storage</code>	Schaltet den Nachrichtenspeicher aus, wenn diese Option aktiviert ist <code>true</code> .
<code>ie.event.wws.version</code>	Die Version des Webservice für das Integrationsereignis.

Referenz: Eigenschaften der Subassembly „GetEventDocuments“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der Untergruppe „GetEventDocuments“ innerhalb der Baugruppe.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortmeldung für die Untergruppe „GetEventDocuments“ .
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob die Untergruppe „GetEventDocuments“ ausgeführt wird.
Endpunkt	endpoint	Die Adresse des Zielendpunkts für die Untergruppe. Die Komponente "GetEventDocuments" unterstützt Folgendes: vm : // Endpunktadresse Beispiel: vm : //wcc/GetEventDocuments.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie " <i>Kein(e)</i> " an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei

Eigenschaft	XML-Attributname	Beschreibung
		„WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um die WS-Adressierungsdetails anzugeben.
Klonanforderung	clone-request	Gibt an, ob die Verarbeitung zwischen Untergruppen verkettet werden soll. Setzen Sie die Eigenschaft auf Wahr, um eine Kopie der ursprünglichen Nachricht zu erstellen, sodass Subassemblies, die die Anforderungsnachricht ändern, keine Auswirkungen auf die anderen haben. Setzen Sie die Eigenschaft auf Falsch, um die ursprüngliche Nachricht zu übermitteln.
Abbruch weitergeben	propagate-abort	Gibt an, ob ein Abbruchkennzeichen von einer Untergruppe an die aufrufende Baugruppe weitergegeben wird. Eine Untergruppe stoppt die Verarbeitung, wenn sie als markiert MediationContext als abgebrochen.
Nicht konfigurierte Eigenschaften	unset-properties	Gibt an, ob alle lokalen Ende - Eigenschaften zurückgesetzt werden, nachdem der lokale Ende- Transport eine Untergruppe aufgerufen hat.

Register „Parameter“

Parameter	Beschreibung
ie.event.wid	Die Workday-ID für das Integrationsereignis, dessen Servicekonfigurationen Sie abrufen möchten. Wird automatisch mit ausgefüllt lp.isSet() ? lp.getIntegrationEventWID() : null.

Parameter	Beschreibung
	Dies bedeutet, dass der Wert des Parameters standardmäßig die Integrationsereignis-WID vorgibt, wenn das Startdokument im Vermittlungskontext angegeben wurde, und null, wenn dies nicht der Fall ist.

Referenz: Eigenschaften der Subassembly „GetIntegrationEvent“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der Untergruppe „GetIntegrationEvent“ innerhalb der Baugruppe.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortmeldung für die Untergruppe „GetIntegrationEvent“ .
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob die Untergruppe „GetIntegrationEvent“ ausgeführt wird.
Endpunkt	endpoint	Die Adresse des Zielpunkts für die Untergruppe. Die Komponente GetIntegrationEvent unterstützt vm:// Endpunktadresse Beispiel: vm://wcc/GetIntegrationEvent.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie "Kein(e)" an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.

Eigenschaft	XML-Attributname	Beschreibung
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um die WS-Adressierungsdetails anzugeben.
Klonanforderung	clone-request	Gibt an, ob die Verarbeitung zwischen Untergruppen verkettet werden soll. Setzen Sie die Eigenschaft auf Wahr, um eine Kopie der ursprünglichen Nachricht zu erstellen, sodass Subassemblys, die die Anforderungsnachricht ändern, keine Auswirkungen auf die anderen haben. Setzen Sie die Eigenschaft auf Falsch, um die ursprüngliche Nachricht zu übermitteln.
Abbruch weitergeben	propagate-abort	Gibt an, ob ein Abbruchkennzeichen von einer Untergruppe an die aufrufende Baugruppe weitergegeben wird. Eine Untergruppe stoppt die Verarbeitung, wenn sie als markiert MediationContext als abgebrochen.
Nicht konfigurierte Eigenschaften	unset-properties	Gibt an, ob alle lokalen Ende - Eigenschaften zurückgesetzt werden, nachdem der lokale Ende- Transport eine Untergruppe aufgerufen hat.

Register „Parameter“

Parameter	Beschreibung
ie.event.wid	Die Workday-ID für das Integrationsereignis, dessen Servicekonfigurationen Sie abrufen möchten.

Referenz: Eigenschaften der Subassembly „GetIntegrationSystems“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der Untergruppe "GetIntegrationSystems" innerhalb der Baugruppe.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortmeldung für die Untergruppe "GetIntegrationSystems" .
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob die Untergruppe „GetIntegrationSystems“ ausgeführt wird.
Endpunkt	endpoint	Die Adresse des Zielpunkts für die Untergruppe. Die Komponente "GetIntegrationSystems" unterstützt: vm: // Endpunktadresse Beispiel: vm: //wcc/ GetIntegrationSystems.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie "Kein(e)" an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/ Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll.

Eigenschaft	XML-Attributname	Beschreibung
		Verwenden Sie den Schritt <code>set-dynamic-endpoint</code> , um die WS-Adressierungsdetails anzugeben.
Klonanforderung	<code>clone-request</code>	Gibt an, ob die Verarbeitung zwischen Untergruppen verkettet werden soll. Setzen Sie die Eigenschaft auf Wahr, um eine Kopie der ursprünglichen Nachricht zu erstellen, sodass Subassemblies, die die Anforderungsnachricht ändern, keine Auswirkungen auf die anderen haben. Setzen Sie die Eigenschaft auf Falsch, um die ursprüngliche Nachricht zu übermitteln.
Abbruch weitergeben	<code>propagate-abort</code>	Gibt an, ob ein Abbruchkennzeichen von einer Untergruppe an die aufrufende Baugruppe weitergegeben wird. Eine Untergruppe stoppt die Verarbeitung, wenn sie als markiert <code>MediationContext</code> als abgebrochen.
Nicht konfigurierte Eigenschaften	<code>unset-properties</code>	Gibt an, ob alle lokalen Ende - Eigenschaften zurückgesetzt werden, nachdem der lokale Ende- Transport eine Untergruppe aufgerufen hat.

Register „Parameter“

Parameter	Beschreibung
<code>is.system.wid</code>	Die Workday-ID für das Integrationssystem, dessen Servicekonfigurationen Sie abrufen und parsen möchten. Wird automatisch mit ausgefüllt <code>lp.getIntegrationSystemRefWID()</code> .

Referenz: Eigenschaften für Subassembly „PagedGet“**Register „Allgemein“**

Eigenschaft	XML-Attributname	Beschreibung
Id	<code>id</code>	Die eindeutige ID der PagedGet- Untergruppe innerhalb der Gruppe.

Eigenschaft	XML-Attributname	Beschreibung
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortmeldung für die PagedGet- Untergruppe.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob die PagedGet- Untergruppe ausgeführt wird.
Endpunkt	endpoint	Die Adresse des Zielendpunkts für die Untergruppe. Die PagedGet -Komponente unterstützt vm: // Endpunktadresse Beispiel: vm: //wcc/PagedGet.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie "Kein(e)" an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um die WS-Adressierungsdetails anzugeben.
Klonanforderung	clone-request	Gibt an, ob die Verarbeitung zwischen Untergruppen verkettet werden soll. Setzen Sie die Eigenschaft auf Wahr, um eine Kopie der ursprünglichen Nachricht zu erstellen, sodass Subassemblies, die die

Eigenschaft	XML-Attributname	Beschreibung
		Anforderungsnachricht ändern, keine Auswirkungen auf die anderen haben. Setzen Sie die Eigenschaft auf Falsch, um die ursprüngliche Nachricht zu übermitteln.
Abbruch weitergeben	propagate-abort	Gibt an, ob ein Abbruchkennzeichen von einer Untergruppe an die aufrufende Baugruppe weitergegeben wird. Eine Untergruppe stoppt die Verarbeitung, wenn sie als markiert MediationContext als abgebrochen.
Nicht konfigurierte Eigenschaften	unset-properties	Gibt an, ob alle lokalen Ende - Eigenschaften zurückgesetzt werden, nachdem der lokale Ende- Transport eine Untergruppe aufgerufen hat.

Register „Parameter“

Parameter	Beschreibung
<code>is.paged.get.request.current.page.xpath</code>	Der XPath-Speicherort im Anforderungsdokument, an dem PagedGet Die Komponente "" legt die Anzahl der angeforderten Seiten fest.
<code>is.paged.get.response.current.page.xpath</code>	Der XPath-Ausdruck, um den aktuellen Seitenwert aus der Webservice-Antwortmeldung zu extrahieren.
<code>is.paged.get.response.total.pages.xpath</code>	Der XPath-Ausdruck, um den Wert der Seitensumme aus der Webservice-Antwortmeldung zu extrahieren.
<code>is.paged.get.response.total.results.xpath</code>	Der XPath-Ausdruck, um den Gesamtwert der Ergebnisseite aus der Webservice-Antwortmeldung zu extrahieren.
<code>is.paged.get.process.endpoint</code>	Der lokal eingehende Endpunkt der Untergruppe, die jede Ergebnisseite verarbeitet. Beispiel: <code>vm://payrollInterface/ProcessPayeesLocalIn</code>. Anmerkung: Workday ignoriert Aggregatparameter, wenn Sie diesen Parameter festlegen. Diese MediationContext -Eigenschaften sind in der Verarbeitungsuntergruppe verfügbar: <code>is.paged.get.request.current.page.xpath</code> <code>is.paged.get.response.current.page.xpath</code> <code>is.paged.get.response.total.pages.xpath</code>

Parameter	Beschreibung
	<code>is.paged.get.response.total.results.xpath</code>
<code>is.paged.get.aggregate.header</code>	Der Header-Wert für aggregierte Antworten in Form eines XML-Fragments. Beispiel: <code><AllData></code> .
<code>is.paged.get.aggregate.footer</code>	Der Header-Wert für aggregierte Antworten in Form eines XML-Fragments. Beispiel: <code></AllData></code> .
<code>is.paged.get.aggregate.xpath</code>	Der XPath-Ausdruck, der beim Aggregieren auf die Antwortmeldung angewendet werden soll.
<code>is.paged.get.namespaces</code>	Gibt Namespaces an, die im Dokument nicht verfügbar sind.
<code>is.paged.get.application</code>	Die Anwendungsgruppe des Webservice, den Sie aufrufen. Beispiel: <code>Human_Resources</code> .
<code>is.paged.get.version</code>	Die Version des öffentlichen Webservice, den Sie aufrufen. Als Best Practice sollten Sie die zum Zeitpunkt der Erstellung der Integration aktuelle WWS-Version verwenden. Verwenden Sie keine dynamische neueste Version, die sich während der Ausführung der Integration ändert. Bei Baugruppen mit einer Version vor 2016.45 wird dieser Parameter automatisch mit ausgefüllt <code>util.assemblyVersionAsWWSVersion</code>. Bei Builds mit Version 2016.45 oder höher müssen Sie explizit die Webservices-Version angeben. Beispiel: <code>v27.2</code>.
<code>is.paged.get.page.zero</code>	Gibt an, ob eine Antwortseite für Fälle generiert werden soll, in denen die Gesamtanzahl der Seiten 0 ist.
<code>is.paged.get.parallel</code>	Gibt an, ob Webservice-Antworten parallel abgerufen werden sollen, sodass die in benannte Untergruppe aktiviert wird <code>is.paged.get.process.endpoint</code> - Parameter zur Verarbeitung jeder Paging-Antwort. Wird automatisch mit ausgefüllt <code>false</code>. Dies bedeutet, dass Webservice-Antworten aggregiert und nicht parallel abgerufen werden. Workday verarbeitet Parallelantworten mithilfe eindeutiger Kontexte, die einen schreibgeschützten Zugriff auf die Eigenschaftswerte und Variablen ermöglichen, die vor der Eingabe von <code>PagedGet</code> festgelegt wurden. Wenn die Untergruppe mit jeder Seite abgeschlossen wird, gehen alle Eigenschaften oder Variablen verloren, die während dieser Verarbeitung festgelegt wurden. Das <code>PagedGet</code> legt weiterhin fest <code>is.paged.get.total.pages</code> und <code>is.paged.get.total.results</code>

Parameter	Beschreibung
	-Eigenschaften für den Aufruf MediationContext. Die <code>is.paged.get.current.page</code> Die Eigenschaft "" wird in den Kontexten für Parallelseiten festgelegt. Workday verarbeitet nicht parallele Antworten im selben Kontext und die Seiten in der Untergruppe. Das bedeutet, dass Sie Eigenschaften und Variablen in der Untergruppe festlegen und den Status vorheriger Seiten prüfen können.
<code>is.paged.get.store.requests</code>	Gibt an, ob die Webservice-Anforderungsmeldungen für jede Datenseite im Nachrichtenspeicher gespeichert werden sollen.
<code>is.paged.get.overall.timeout.seconds</code>	Der Gesamt-Timeout in Sekunden. Wird kein Wert festgelegt, liegt der Standardwert bei 6 Stunden. Sie können ein maximales Timeout von 30 Stunden festlegen.
<code>is.paged.get.timeout.seconds</code>	Das Timeout für die Verarbeitung jeder Seite. Wird kein Wert festgelegt, liegt der Standardwert bei 6 Stunden. Darf den Gesamt-Timeout-Wert nicht überschreiten.

Ausgangsparameter

Bei der Verarbeitung übergibt die Subassembly PagedGet die folgenden Ausgangsparameterwerte zur Verwendung durch nachgelagerte Komponenten:

Ausgangsparameter	Beschreibung
<code>is.paged.get.last.page</code>	Die letzte Seite, die vom Webservicevorgang zurückgegeben wurde.
<code>is.paged.get.current.page</code>	Die Nummer der Seite, die aktuell verarbeitet wird.
<code>is.paged.get.total.pages</code>	Die Gesamtanzahl der zu verarbeitenden Seiten.
<code>is.paged.get.total.results</code>	Die Gesamtanzahl der vom Webservicevorgang zurückgegebenen Ergebnisse.

Referenz: Eigenschaften der Subassembly „PagedGetLocalPaging“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der Untergruppe PagedGetLocalPaging innerhalb der Baugruppe.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortmeldung für die Untergruppe PagedGetLocalPaging .

Eigenschaft	XML-Attributname	Beschreibung
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob die Untergruppe PagedGetLocalPaging ausgeführt wird.
Endpunkt	endpoint	Die Adresse des Zielendpunkts für die Untergruppe. Die Komponente PagedGetLocalPaging unterstützt: <code>vm://</code> Endpunktadresse Beispiel: <code>vm://wcc/PagedGet</code> .

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie " <i>Kein(e)</i> " an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um die WS-Adressierungsdetails anzugeben.
Klonanforderung	clone-request	Gibt an, ob die Verarbeitung zwischen Untergruppen verkettet werden soll. Setzen Sie die Eigenschaft auf Wahr, um eine Kopie der ursprünglichen Nachricht zu erstellen, sodass Subassemblies, die die

Eigenschaft	XML-Attributname	Beschreibung
		Anforderungsnachricht ändern, keine Auswirkungen auf die anderen haben. Setzen Sie die Eigenschaft auf Falsch, um die ursprüngliche Nachricht zu übermitteln.
Abbruch weitergeben	propagate-abort	Gibt an, ob ein Abbruchkennzeichen von einer Untergruppe an die aufrufende Baugruppe weitergegeben wird. Eine Untergruppe stoppt die Verarbeitung, wenn sie als markiert MediationContext als abgebrochen.
Nicht konfigurierte Eigenschaften	unset-properties	Gibt an, ob alle lokalen Ende - Eigenschaften zurückgesetzt werden, nachdem der lokale Ende- Transport eine Untergruppe aufgerufen hat.

Register „Parameter“

Parameter	Beschreibung
is.paged.get.process.endpoint	Gibt den lokal-in- Endpunkt der Untergruppe an, die jede Seite mit Ergebnissen verarbeitet. Beispiel: vm://payrollInterface/ProcessPayeesLocalIn. Anmerkung: Workday ignoriert Aggregatparameter, wenn Sie diesen Parameter festlegen. Diese MediationContext -Eigenschaften sind in der Verarbeitungsuntergruppe verfügbar: - is.paged.get.request.current.page.xpath - is.paged.get.response.current.page.xpath - is.paged.get.response.total.pages.xpath - is.paged.get.response.total.results.xpath
is.paged.get.page.size	Die Anzahl der Datensätze auf jeder Datenseite, die abgerufen werden soll. Wird automatisch mit ausgefüllt 100.
is.paged.get.response.reference.wids.xpath	Der XPath-Ausdruck, mit dem die WIDs in der Meldung zum Abrufen aller WIDs gesucht werden. Beispiel: //wd:Payee_Reference/wd:ID[@wd:type='WID'].
is.paged.get.request.reference.wids.xpath	Der XPath-Ausdruck, mit dem das übergeordnete Element der Anforderungsmeldung gesucht wird, in dem WIDs beim Abrufen jeder Datenseite angegeben werden.

Parameter	Beschreibung
<code>is.paged.get.request.reference.wids.elements</code>	Der Elementname für die WID, die in der Anforderung für jede Datenseite verwendet werden soll.
<code>is.paged.get.application</code>	Die Anwendungsgruppe des Webservice, den Sie aufrufen. Beispiel: Human_Resources.
<code>is.paged.get.version</code>	Die Version des öffentlichen Webservice, den Sie aufrufen. Als Best Practice sollten Sie die zum Zeitpunkt der Erstellung der Integration aktuelle WWS-Version verwenden. Verwenden Sie keine dynamische neueste Version, die sich während der Ausführung der Integration ändert. Bei Baugruppen mit einer Version vor 2016.45 wird dieser Parameter automatisch mit <code>util.assemblyVersionAsWWSVersion</code> ausgefüllt. Bei Builds mit Version 2016.45 oder höher müssen Sie explizit die Webservices-Version angeben. Beispiel: <code>v27.2</code>.
<code>is.paged.get.xpath.ns.prefix</code>	Das Präfix des Workday-XML-Namespace, der in allen XPath-Ausdrücken verwendet wird. Wird automatisch mit <code>wd</code> ausgefüllt.

Referenz: Eigenschaften der Subassembly „PdfPrintStep“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	<code>id</code>	Die eindeutige ID der PdfDruckSchritt- Untergruppe innerhalb der Gruppe.
Antwort weiterleiten an	<code>routes-response-to</code>	Das Ziel der Antwortmeldung für die Untergruppe von PdfDruckSchritt .
Ausführung bei	<code>execute-when</code>	Eine Bedingung, die bestimmt, ob die Untergruppe PdfDruckSchritt ausgeführt wird.
Endpunkt	<code>endpoint</code>	Die Adresse des Zielpunkts für die Untergruppe. Die Komponente "PdfDruckSchritt" unterstützt Folgendes: <code>vm://</code> Endpunktadresse Beispiel: <code>vm://wcc/PdfPrintStep</code> .

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie "<i>Kein(e)</i>" an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um die WS-Adressierungsdetails anzugeben.
Klonanforderung	clone-request	Gibt an, ob die Verarbeitung zwischen Untergruppen verkettet werden soll. Setzen Sie die Eigenschaft auf Wahr, um eine Kopie der ursprünglichen Nachricht zu erstellen, sodass Subassemblies, die die Anforderungsnachricht ändern, keine Auswirkungen auf die anderen haben. Setzen Sie die Eigenschaft auf Falsch, um die ursprüngliche Nachricht zu übermitteln.
Abbruch weitergeben	propagate-abort	Gibt an, ob ein Abbruchkennzeichen von einer Untergruppe an die aufrufende Baugruppe weitergegeben wird. Eine Untergruppe stoppt die Verarbeitung, wenn sie als markiert MediationContext als abgebrochen.

Eigenschaft	XML-Attributname	Beschreibung
Nicht konfigurierte Eigenschaften	unset-properties	Gibt an, ob alle lokalen Ende - Eigenschaften zurückgesetzt werden, nachdem der lokale Ende- Transport eine Untergruppe aufgerufen hat.

Register „Parameter“

Parameter	Beschreibung
<code>pdf.report.design.variable</code>	Der Name der Integrationsvariable, die das Berichtsdesign enthält, das das Layout des gedruckten benutzerdefinierten Berichts definiert. Beispiel: <code>'test.report.design'</code> . Die Definition des Berichtsdesigns ist in einer Datei mit der Erweiterung <code>.rptdesign</code> enthalten.
<code>pdf.workday.report.variable</code>	Der Name der Integrationsvariable, die die Rohdaten des Workday-Berichts enthält. Beispiel: <code>'test.workday.report'</code> .
<code>pdf.timezone</code>	Die Java-Zeitzone für Formatierungszwecke. Beispiel: • <code>'US/Eastern'</code> • <code>'GMT'</code> • <code>'CET'</code>
<code>pdf.language</code>	Die Sprache des Berichts für Formatierungszwecke. Beispiel: • <code>'en_US'</code> • <code>'fr_FR'</code> • <code>'de_DE'</code>
<code>pdf.locale</code>	Das Gebietsschema des Berichts für Formatierungszwecke. Beispiel: • <code>'en_US'</code> • <code>'fr_FR'</code> • <code>'de_DE'</code>
<code>pdf.business.form.layout.wid</code>	Die Workday-ID des verwendeten Geschäftsformularlayouts. Ermöglicht das Abrufen von Bildern, die im Layout als Token definiert sind.
<code>pdf.report.design.wid</code>	Die Workday-ID des verwendeten Berichtsdesigns. Aktiviert die Übersetzung.
<code>pdf.apply.report.design.per.row</code>	Gibt an, ob Workday das Berichtsdesign auf einzelne Berichtszeilen und nicht auf den Bericht als Ganzer anwendet.

relatedlinks

Referenz

[The Next Level: Using BIRT in Studio Integrations](#)

Referenz: Eigenschaften der Subassembly „PutIntegrationEvent“

Allgemeine Tabelle

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der Untergruppe PutIntegrationEvent innerhalb der Build.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortmeldung für die Untergruppe PutIntegrationEvent .
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob die Unterdomäne PutIntegrationEvent ausgeführt wird.
Endpunkt	endpoint	Die Adresse des Zielpunkts für die Untergruppe. Die Komponente PutIntegrationEvent unterstützt: vm:// Endpunktadresse Beispiel: vm://wcc/ PutIntegrationEvent.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie "Kein(e)" an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll.

Eigenschaft	XML-Attributname	Beschreibung
		Verwenden Sie den Schritt <code>set-dynamic-endpoint</code> , um die WS-Adressierungsdetails anzugeben.
Klonanforderung	<code>clone-request</code>	Gibt an, ob die Verarbeitung zwischen Untergruppen verkettet werden soll. Setzen Sie die Eigenschaft auf Wahr, um eine Kopie der ursprünglichen Nachricht zu erstellen, sodass Subassemblies, die die Anforderungsnachricht ändern, keine Auswirkungen auf die anderen haben. Setzen Sie die Eigenschaft auf Falsch, um die ursprüngliche Nachricht zu übermitteln.
Abbruch weitergeben	<code>propagate-abort</code>	Gibt an, ob ein Abbruchkennzeichen von einer Untergruppe an die aufrufende Baugruppe weitergegeben wird. Eine Untergruppe stoppt die Verarbeitung, wenn sie als markiert <code>MediationContext</code> als abgebrochen.
Nicht konfigurierte Eigenschaften	<code>unset-properties</code>	Gibt an, ob alle lokalen Ende - Eigenschaften zurückgesetzt werden, nachdem der lokale Ende- Transport eine Untergruppe aufgerufen hat.

Register „Parameter“

Parameter	Beschreibung
<code>is.event.wid</code>	Die Workday-ID des Integrationsereignisses, mit dem Workday die Meldung verknüpft. Wenn Sie diesen Wert nicht angeben und er nicht in vorhanden ist <code>lp</code> Beim Transfer an die MVSEL-Variable verwendet Workday die <code>is.system.wid</code> -Parameter, der die Meldung mit dem Integrationssystem verknüpft. Wird automatisch mit ausgefüllt <code>lp.isSet() ? lp.getIntegrationEventWID() : null.</code>
<code>is.system.wid</code>	Die Workday-ID des Integrationssystems. Wird automatisch mit ausgefüllt <code>lp.isSet() ? lp.getIntegrationSystemRefWID() : null.</code>

Parameter	Beschreibung
	Workday verwendet diesen Parameter nur, wenn die Integrationsereignis-ID nicht verfügbar ist.
<code>is.process.desc</code>	Die Beschreibung des Integrationsereignisses. Workday zeigt diesen Wert im Prozessmonitor an. Erforderlich beim Erstellen eines neuen Integrationsereignisses.
<code>is.event.initiated</code>	Datum und Uhrzeit der Initiierung des Integrationsereignisses. Wenn Sie den Zeichenfolgenwert eingeben 'current', werden das aktuelle Datum und die aktuelle Uhrzeit an Workday gesendet.
<code>is.event.completed</code>	Das Datum und die Uhrzeit des Abschlusses des Integrationsereignisses. Wenn Sie den Zeichenfolgenwert eingeben 'current', werden das aktuelle Datum und die aktuelle Uhrzeit an Workday gesendet.
<code>is.response.msg</code>	Die letzte Nachricht oder Aktivität für das Integrationsereignis. Ein Freiformtextfeld.
<code>is.event.member.wids</code>	Eine durch Kommas getrennte Liste von Workday-IDs, die Mitglieder darstellen, die mit diesem Integrationsereignis verknüpft sind.
<code>is.percent.complete</code>	Ein Wert, den Workday verwendet, um den Fortschrittsbalken für den Fertigstellungsgrad zu verwalten. Der Wert kann von jeder Art sein, die in ein Java-BigDecimal-Objekt konvertiert werden kann. Der Wert muss zwischen 0 und 100 liegen.

Referenz: Eigenschaften der Subassembly „PutIntegrationMessage“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der Untergruppe PutIntegrationMessage innerhalb der Build.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortmeldung für die Untergruppe PutIntegrationMessage .
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob die Untergruppe PutIntegrationMessage ausgeführt wird.
Endpunkt	endpoint	Die Adresse des Zielendpunkts für die Untergruppe. Die Komponente PutIntegrationMessage unterstützt vm: //

Eigenschaft	XML-Attributname	Beschreibung
		Endpunktadresse Beispiel: <code>vm://wcc/</code> <code>PutIntegrationMessage</code> .

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	<code>store-message</code>	Geben Sie " <i>Kein(e)</i> " an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt <code>store</code> in Studio.
Transportklasse	<code>transport-class</code>	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	<code>ignore-dynamic-endpoints</code>	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt <code>set-dynamic-endpoint</code> , um die WS-Adressierungsdetails anzugeben.
Klonanforderung	<code>clone-request</code>	Gibt an, ob die Verarbeitung zwischen Untergruppen verkettet werden soll. Setzen Sie die Eigenschaft auf Wahr, um eine Kopie der ursprünglichen Nachricht zu erstellen, sodass Subassemblys, die die Anforderungsnachricht ändern, keine Auswirkungen auf die anderen haben. Setzen Sie die Eigenschaft auf Falsch, um die ursprüngliche Nachricht zu übermitteln.
Abbruch weitergeben	<code>propagate-abort</code>	Gibt an, ob ein Abbruchkennzeichen von einer

Eigenschaft	XML-Attributname	Beschreibung
		Untergruppe an die aufrufende Baugruppe weitergegeben wird. Eine Untergruppe stoppt die Verarbeitung, wenn sie als markiert MediationContext als abgebrochen.
Nicht konfigurierte Eigenschaften	unset-properties	Gibt an, ob alle lokalen Ende - Eigenschaften zurückgesetzt werden, nachdem der lokale Ende- Transport eine Untergruppe aufgerufen hat.

Register „Parameter“

Parameter	Beschreibung
<code>is.event.wid</code>	Die Workday-ID des Integrationsereignisses, mit dem Workday die Meldung verknüpft. Wenn Sie diesen Wert nicht angeben und er nicht in vorhanden ist lp Beim Transfer an die MVSEL-Variable verwendet Workday die <code>is.system.wid</code> -Parameter, der die Meldung mit dem Integrationssystem verknüpft. Wird automatisch mit ausgefüllt <code>lp.isSet() ? lp.getIntegrationEventWID() : null</code> .
<code>is.system.wid</code>	Die Workday-ID des Integrationssystems. Wird automatisch mit ausgefüllt <code>lp.isSet() ? lp.getIntegrationSystemRefWID() : null</code> . Workday verwendet diesen Parameter nur, wenn die Integrationsereignis-ID nicht verfügbar ist.
<code>is.message.severity</code>	Gibt den Schweregrad der Meldung an: 'CRITICAL', 'DEBUG', 'ERROR', 'INFO', oder 'WARNING'. Wird automatisch mit ausgefüllt 'INFO'.
<code>is.message.summary</code>	Die an Workday gesendete Nachrichtenzusammenfassung.
<code>is.message.detail</code>	Das Detail der an Workday gesendeten Nachricht.
<code>is.message.detail.richtext</code>	Bestimmt, ob das Nachrichtendetail als Rich Text formatiert wird. Wird automatisch mit ausgefüllt <code>true</code> .
<code>is.document.variable.name</code>	Der Name einer Variable in der MediationContext, die die Ausgabe des Atom-Schemas eines Lagerschritts enthält. Der variable Inhalt wird verwendet, um eine Referenz auf das Integrationsereignis hinzuzufügen. Die store Für den Schritt, mit dem das Dokument erstellt wird, müssen

Parameter	Beschreibung
	folgende Schemaattribute festgelegt sein: schema="http://www.w3.org/2005/Atom".
is.document.file.name	Der Dateiname des Dokuments. Wenn Sie keinen Titel angeben, verwendet Workday den Dokumenttitel aus dem Kerndokument.
is.document.owner.wid	Die Workday-ID des Dokumenteigentümers. Wenn Sie keine ID angeben, verwendet Workday die ID des aktuellen Integrationssystembenutzers.
is.document.labels	Eine durch Kommas getrennte Liste von Labels, die auf das Dokument angewendet werden, wenn es sich im Bereich der Integrationsereignis-ID oder des zugehörigen Geschäftsprozesses befindet.
is.document.deliverable	Gibt an, ob das zum Integrationsereignis hinzugefügte Dokument automatisch an einen Kunden oder einen externen Anbieter übermittelt wird. Wird automatisch mit ausgefüllt 'true'.
is.document.delivery.services	Eine durch Kommas getrennte Liste mit Namen von Integrationsservices, mit denen das Dokument getaggt wird, das an das Integrationsereignis angehängt wird. Standardmäßig verwendet Workday alle mit dem Integrationssystem verknüpften Übermittlungsservices.
is.document.retrieved	Gibt an, ob das Dokument so behandelt wird, als wäre es vom Abrufservice abgerufen worden. Wird automatisch mit ausgefüllt 'false'.
is.message.targets	Eine durch Kommas getrennte Liste mit Referenzen auf Zielobjekte für diese Meldung. Diese Ziele werden zu Objektlinks auf dem Register Integrationsmeldung anzeigen der Workday-Anwendung.
is.message.targets.type	Eine Zielreferenzart Wird automatisch mit ausgefüllt 'WID'.
is.message.secured.instance.refs	Eine durch Kommas getrennte Liste gesicherter Instanzreferenzen für diese Meldung.
is.message.secured.instance.refs.type	Ermöglicht das Überschreiben der Standardart für gesicherte Instanzreferenzen. Wird automatisch mit ausgefüllt 'WID'.
is.message.storage.enabled	Bestimmt, ob die Nachricht gespeichert wird. Wird automatisch mit ausgefüllt 'false'.

Referenz: Eigenschaften der Subassembly „SalesforceConnector“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID der SalesforceConnector-Untergruppe innerhalb der Gruppe.
Antwort weiterleiten an	routes-response-to	Das Ziel der Antwortnachricht für die SalesforceConnector-Untergruppe.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob die SalesforceConnector-Untergruppe ausgeführt wird.
Endpunkt	endpoint	Die Adresse des Zielpunkts für die Untergruppe. Die SalesforceConnector - Komponente unterstützt: vm: // Endpunktadresse Beispiel: vm: //wcc/ SalesforceConnector.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	store-message	Geben Sie "Kein(e)" an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	transport-class	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	ignore-dynamic-endpoints	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt set-dynamic-endpoint, um

Eigenschaft	XML-Attributname	Beschreibung
		die WS-Adressierungsdetails anzugeben.
Klonanforderung	clone-request	Gibt an, ob die Verarbeitung zwischen Untergruppen verkettet werden soll. Setzen Sie die Eigenschaft auf Wahr, um eine Kopie der ursprünglichen Nachricht zu erstellen, sodass Subassemblies, die die Anforderungsnachricht ändern, keine Auswirkungen auf die anderen haben. Setzen Sie die Eigenschaft auf Falsch, um die ursprüngliche Nachricht zu übermitteln.
Abbruch weitergeben	propagate-abort	Gibt an, ob ein Abbruchkennzeichen von einer Untergruppe an die aufrufende Baugruppe weitergegeben wird. Eine Untergruppe stoppt die Verarbeitung, wenn sie als markiert MediationContext als abgebrochen.
Nicht konfigurierte Eigenschaften	unset-properties	Gibt an, ob alle lokalen Ende - Eigenschaften zurückgesetzt werden, nachdem der lokale Ende- Transport eine Untergruppe aufgerufen hat.

Register „Parameter“

Parameter	Beschreibung
wd.sf.auth.username	Benutzername für Ihr Salesforce.com-Entwicklerkonto.
wd.sf.auth.password	Das Kennwort Ihres Salesforce.com-Entwicklerkontos.
wd.sf.msg.varname	Die XML-Nachricht, die Workday an den Salesforce.com-Endpunkt sendet.
wd.sf.auth.token	Ihr Sicherheits-Token für das Salesforce.com-Entwicklerkonto.
wd.sf.auth.endpoint	Optionaler Salesforce.com-Endpunkt. Ein Standardendpunkt ist bereits als Teil der Untergruppe konfiguriert http-out Transport.
wd.sf.api.custom.version	Eine optionale Salesforce.com-API-Version. Wenn Sie keinen Wert angeben, wird die Version der Endpunkt-API verwendet.

Die PrismAnalytics-Untergruppe

Schritte: CSV-Daten an Prism Analytics senden

Vorbereitungen

Sicherheit: Domäne *Security Administration* im funktionalen Bereich „System“.

Warum und wann dieser Vorgang ausgeführt wird

Mit der PrismAnalytics-Komponente von Workday Studio können Sie CSV-Daten an Workday Prism Analytics senden. Bevor Sie eine Assembly für die Verwendung von PrismAnalytics konfigurieren, müssen Sie einen Prism-API-Client in Workday registrieren und einen REST-API-Endpunkt anfordern. Sie müssen auch eine JSON-Beschreibung der Quell-CSV-Daten erstellen.

Prozedur

1. [Einen Workday Prism Analytics-API-Client registrieren](#) auf Seite 138.
Registrieren Sie in Workday einen Integrations-API-Client mit Prism Analytics als Bereich. Rufen Sie die Werte für Client-ID, geheimer Client-Schlüssel und Aktualisierungs-Token ab, die die PrismAnalytics-Komponente als Parameter benötigt.
2. [Workday-REST-API-Endpunkt anfordern](#) auf Seite 139.
Rufen Sie in Workday den Workday-REST-API-Endpunkt ab, den die PrismAnalytics-Komponente als Parameter benötigt.
3. [CSV-Quelldaten beschreiben](#) auf Seite 139.
Beschreiben Sie die CSV-Quelldaten in JSON-Text, die als Variable an die PrismAnalytics-Komponente übergeben werden sollen.
4. [Assembly mit PrismAnalytics-Subassembly konfigurieren](#) auf Seite 146.
Erstellen Sie in Workday Studio eine Build, die CSV-Daten abrufen und an eine ordnungsgemäß konfigurierte PrismAnalytics-Komponente weiterleitet.

relatedlinks

Referenz

[Workday 31 What's New Post: Workday Studio](#)

Einen Workday Prism Analytics-API-Client registrieren

Vorbereitungen

Sicherheit: Domäne *Security Administration* im funktionalen Bereich „System“.

Warum und wann dieser Vorgang ausgeführt wird

Um die PrismAnalytics-Komponente von Workday Studio verwenden zu können, müssen Sie einen API-Client in Workday mit Prism Analytics als Umfang registrieren. Die Komponente erfordert die Werte Client-ID und Geheimer Client-Schlüssel, die Workday während des Registrierungsprozesses generiert. Außerdem ist ein Wert für Aktualisierungs-Token erforderlich.

Prozedur

1. Öffnen Sie die Aufgabe Register API Client for Integrations. Bei der Bearbeitung dieser Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Nicht ablaufende Aktualisierungs-Token	Verhindert, dass für das Aktualisierungs-Token ein Timeout erreicht wird. Wird automatisch von Workday ausgewählt. Nicht löschen.
Umfang (Funktionaler Bereich)	Wählen Sie Prism Analytics aus.
Workday-eigenen Bereich einschließen	Bietet Zugriff auf Workday-Kerndomänen, die sich nicht in funktionalen Bereichen befinden. Aktivieren Sie dieses Kontrollkästchen.

2. Kopieren Sie die Werte für Client-ID und Geheimer Client-Schlüssel . Die PrismAnalytics-Komponente von Studio erfordert sie für `wd.prism.component.client.id` und `wd.prism.component.client.secret` Parameter.
3. In den möglichen Aktionen wählen Sie API-Client > Aktualisierungs-Tokens für Integrationen verwalten.
4. Wählen Sie ein Workday-Konto aus, das mit dem Aktualisierungs-Token verknüpft werden soll. Das Konto muss die Berechtigung zum Erstellen von Prism-Tabellen in Workday haben.
Anmerkung: Sie können diese Schritte nicht als Integrationssystembenutzer ausführen.
5. Wählen Sie Neues Aktualisierungs-Token generieren aus.
6. Kopieren Sie den Wert Aktualisierungs-Token . Die PrismAnalytics- Komponente erfordert sie für `wd.prism.component.refresh.token` Parameter.

Workday-REST-API-Endpunkt anfordern

Vorbereitungen

Sicherheit: Domäne *Security Administration* im funktionalen Bereich „System“.

Warum und wann dieser Vorgang ausgeführt wird

Die Studio-Komponente PrismAnalytics erfordert einen Workday-REST-API-Endpunkt. Sie finden diesen Endpunkt in Workday.

Prozedur

1. Öffnen Sie den Bericht API-Clients anzeigen.
2. Kopieren Sie den Wert Workday REST API Endpoint .
Beispiel: `https://i-07d55d20e81e2d191.workdaysuv.com/ccx/api/v1/super.`
3. Löschen Sie die `https://` am Anfang und die `/ccx/api/v1/<tenant_name>` am Ende.
Beispiel: `i-07d55d20e81e2d191.workdaysuv.com.`
4. Kopieren Sie den Text. Die PrismAnalytics- Komponente von Studio erfordert sie `wd.prism.component.api.hostname` Parameter.

CSV-Quelldaten beschreiben

Warum und wann dieser Vorgang ausgeführt wird

Die PrismAnalytics-Komponente von Workday Studio erfordert eine Beschreibung der CSV-Quelldaten. Sie können den JSON-Text im Tool Ihrer Wahl erstellen und später in Studio verwenden. Beispiel: Fügen Sie ihn auf dem Register Nachrichten-Builder eines write-Schritts in eine Textnachricht ein.

Prozedur

1. Erstellen Sie einen JSON-Text mit zwei Hauptknoten:

- `parseOptions`: Informationen zum Format der CSV-Daten, einschließlich Zeichensatz und Trennzeichen.
- `fields`: Informationen zu jedem Feld in den CSV-Daten, einschließlich Datentyp und Formatierung.

2. In der `parseOptions` -Knoten können Sie folgende Felder definieren:

Feldname	Art	Erforderlich	Beschreibung
<code>charsetName</code>	Objekt	Ja	Der Zeichensatz der CSV-Datendatei. Beispiel: <pre>"charset": { "id": "Encoding=UTF-8" }</pre> Derzeit unterstützt Workday nur das Codierungsformat UTF-8. Workday speichert die Daten letztendlich im Unicode-Format. Zeichen ohne Unicode-Äquivalent werden als Unicode-Ersatzzeichen gespeichert (ein Fragezeichen auf einem Hintergrund mit schwarzer Raute).
<code>fieldsEnclosedBy</code>	String	Nein	Das Zeichen, mit dem Felder in den CSV-Daten eingeschlossen werden. Der Standardwert sind doppelte Anführungszeichen. Wenn ein doppeltes Anführungszeichen innerhalb eines Felds verwendet wird, umgehen Sie dieses, indem Sie ein weiteres doppeltes Anführungszeichen voranstellen. Beispiel: <pre>"fieldsEnclosedBy": "\""</pre>

Feldname	Art	Erforderlich	Beschreibung
fieldsDelimitedBy	String	Ja	Das Zeichen, das zum Trennen von Feldwerten in den CSV-Daten verwendet wird. Beispiel: <pre>"fieldsDelimitedBy": ","</pre>
headerLinesToIgnore	Zahl	Nein	Die Anzahl der Zeilen, die der Parser als Header ignorieren soll. Auf 0 wenn die CSV-Daten keinen Header haben. Beispiel: <pre>"headerLinesToIgnore": 1</pre>
type	Objekt	Ja	Die Art der Eingabedatei. Derzeit unterstützt Workday nur durch Trennzeichen getrennte Dateien. Beispiel: <pre>"type": { "id": "Schema_File_Type=Delimit</pre> Anmerkung: Bei diesem Wert wird zwischen Groß- und Kleinschreibung unterschieden. Beachten Sie das Großbuchstabe „D“ in „Trennzeichen“.

3. In der fields -Knoten können Sie folgende Felder definieren:

Feldname	Art	Erforderlich	Beschreibung
ordinal	Zahl	Ja	Die Reihenfolge des Felds im CSV-Dataset (Index). Die Ordinalwerte: Müssen mit 1 beginnen.

Feldname	Art	Erforderlich	Beschreibung
			• Müssen zusammenhängend sein. • Dürfen keine Nummer überspringen. • Sie dürfen nicht mehr als 1.000 enthalten.
name	String	Ja	Der Name des CSV-Dataset-Felds. Feldnamen: • Müssen im Dataset eindeutig sein. • Können bis zu 255 Zeichen enthalten. • Dürfen nur alphanumerische Zeichen und Unterstriche enthalten. • Müssen mit einem Buchstaben beginnen. • Dürfen nicht mit einem Unterstrich enden. • Darf nicht mit "WPA_" beginnen.
description	String	Nein	Beschreibung des Felds. Maximal 1.000 Zeichen. Beispiel: <pre>"description": "The worker's full name."</pre>
type	Objekt	Ja	Die Art des Felds. Folgende Optionen sind verfügbar: • Text • Numerisch • Boolesch • Datum

Feldname	Art	Erforderlich	Beschreibung
			Beispiel: <pre>"type": { "id": "Schema_Field_Type=Da</pre>
parseFormat	String	Ja (für Datumsarten)	Das Format des Datums. Beispiel: <pre>"parseFormat": "dd-MM-yy HH24:mm:ss"</pre>
precision	Zahl	Ja (für numerische Arten)	Die maximale Anzahl von Stellen in einem numerischen Wert. Schließt alle Zahlen links und rechts eines Dezimalzeichens ein, aber nicht das Dezimalzeichen selbst. Der Maximalwert ist 38. Beispiel: <pre>"precision": 38</pre>
scale	Zahl	Ja (für numerische Arten)	Die Anzahl der Stellen rechts vom Dezimalzeichen in einem numerischen Wert. Muss kleiner sein als precision - Wert Beispiel: <pre>"scale": 2</pre>
defaultValue	Text	Nein	Der Standardwert für ein Feld, wenn in der Upload-Datei kein Wert angegeben ist. Drücken Sie einen Standardtextwert in doppelten Anführungszeichen aus. Beispiel: "N/A". Numerische Standardwerte müssen mit den angegebenen Werten übereinstimmen precision und scale.

Feldname	Art	Erforderlich	Beschreibung
			Standardwerte für das Datum müssen in diesem Format angegeben werden: jjjj-MM-tt'T'HH:mm:ss'Z'.
required	Boolesch	Nein	Bestimmt, ob ein Feld einen Wert enthalten muss. Auf True wenn ein Feld erforderlich ist, andernfalls auf "". False. Wenn ein Feld erforderlich ist, darf sein Wert in Vorgängen zum Einfügen, Aktualisieren oder Hochladen von Daten nicht 0 sein.
externalID	Boolesch	Ja (um Aktualisierungs-, Upsert- und Löschvorgänge zu nutzen).	Gibt an, ob ein Feld eine externe ID darstellt. Auf True , um ein Feld mit einer externen ID festzulegen, andernfalls auf festgelegt False. Da Felder für "Externe ID" nicht "null" sein dürfen, sollten Sie angeben, dass alle Felder für "Externe ID" erforderlich sind. Sie können nur ein Feld als externe ID angeben. Geben Sie eine externe ID in neuen Tabellen an, wenn Sie Aktualisierungs-, Upsert- und Löschfunktionen nutzen möchten. Um die Aktualisierungs-, Upsert- und Löschfunktionen für eine vorhandene Tabelle zu verwenden, müssen

Feldname	Art	Erforderlich	Beschreibung
			Sie sie so ändern, dass sie ein einzelnes Feld enthält, das als externe ID gekennzeichnet ist. Die öffentliche API von Prism unterstützt nicht die Änderung einer Tabelle, die Daten enthält. Nehmen Sie die Änderung in solchen Fällen in Workday vor.

Beispiel

Ihre CSV-Quelldatei beschreibt diese Tabelle:

Mitarbeiter	Personalnummer	Tätigkeitsbezeichnung	Einstellungsdatum	Jahresgehalt
Logan McNeil	21001	Vice President, Human Resources	1/1/2010	213298.00
Oliver Reynolds	21003	Chief Information Officer	1/1/2010	394935.00
Maximilian Schneider	21004	Chief Operating Officer	1/1/2010	255373.00
Teresa Serrano	21005	Controller	1/1/2010	291.785,00

Sie beschreiben es mit JSON wie folgt:

```
{
  "parseOptions": {
    "fieldsDelimitedBy": ",",
    "fieldsEnclosedBy": "\"\"",
    "headerLinesToIgnore": 1,
    "charset": {
      "id": "Encoding=UTF-8"
    }
  },
  "type": {
    "id": "Schema_File_Type=Delimited"
  }
},
"fields": [
  {
    "ordinal": 1,
    "name": "Employee",
    "description": "Employee",
    "precision": 255,
    "scale": 0,
    "parseFormat": null,
    "type": {
      "id": "Schema_Field_Type=Text"
    }
  }
],
```

```

{
  "ordinal": 2,
  "name": "Employee_ID",
  "description": "Employee ID",
  "precision": 38,
  "scale": 0,
  "parseFormat": null,
  "type": {
    "id": "Schema_Field_Type=Numeric"
  }
},
{
  "ordinal": 3,
  "name": "Job_Title",
  "description": "Job Title",
  "precision": 255,
  "scale": 0,
  "parseFormat": null,
  "type": {
    "id": "Schema_Field_Type=Text"
  }
},
{
  "ordinal": 4,
  "name": "Hire_Date",
  "description": "Hire Date",
  "precision": 0,
  "scale": 0,
  "parseFormat": "MM/dd/yy",
  "type": {
    "id": "Schema_Field_Type=Date"
  }
},
{
  "ordinal": 5,
  "name": "Annual_Salary",
  "description": "Annual Salary",
  "precision": 38,
  "scale": 2,
  "type": {
    "id": "Schema_Field_Type=Numeric"
  }
}
],
"schemaVersion": {
  "id": "Schema_Version=1.0"
}
}

```

relatedlinks

Referenz

Referenz: [Unterstützte Datumsformate für externe Daten in Tabellen und Datasets](#)

Assembly mit PrismAnalytics-Subassembly konfigurieren

Warum und wann dieser Vorgang ausgeführt wird

Um CSV-Daten an Prism Analytics zu senden, muss Ihre Assembly eine PrismAnalytics-Komponente enthalten, die Folgendes ausführt:

- Empfängt CSV-Daten als Eingabe, entweder im Root-Teil der Nachricht oder als Variable.
- Empfängt eine Variable, die eine JSON-Beschreibung der CSV-Daten enthält.

- Hat eine Reihe von Parametern, die korrekt konfiguriert sind.

Prozedur

1. Konfigurieren Sie Ihre Assembly, um eine Quell-CSV-Datei oder Quell-CSV-Dateien abzurufen. Je nach der Methode, mit der Sie die CSV-Daten erfassen, müssen Sie möglicherweise einen Variablennamen angeben oder einen MIME-Typ festlegen.

Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

CSV-Anschaffungsmethode	Anmerkungen
Komponente sftp-out	• Geben Sie den Namen der CSV-Datei in der Eigenschaft Muster für Eingabedatei der Komponente an. Sie können Wildcards verwenden. • Verwenden Sie die Komponente get -Methode zum Abrufen einer einzelnen Datei oder ihrer mget -Methode zum Abrufen mehrerer Dateien. • Studio fügt die CSV-Daten zum Root-Teil der Nachricht hinzu. • Es ist kein Variablenname erforderlich. • Studio legt den MIME-Typ automatisch korrekt fest.
Abruf- oder Listener-Service	• Studio fügt die CSV-Daten zu einer Variable hinzu. • Benennen Sie die Variable mit der <code>wd.prism.component.csv.input.varname</code> Parameter. • Studio legt den MIME-Typ automatisch korrekt fest.
Jede andere Methode	• Studio fügt die CSV-Daten zum Root-Teil der Nachricht oder zu einer Variable hinzu. • Benennen Sie die Variable ggf. mit <code>wd.prism.component.csv.input.varname</code> Parameter. • Legen Sie den MIME-Typ des Rootparts oder der Variable auf Folgendes fest: • <i>text/csv</i> • <i>plain/csv</i> • <i>application/zip</i>

2. Erstellen Sie eine Textnachricht auf dem Register Message Builder eines Schritts write und fügen Sie den JSON-Text hinzu, der die Daten beschreibt. Eine Variable ausgeben. Beispiel: `temp.schema`.

Anmerkung: Sie können den JSON-Text direkt zum Nachrichten-Builder hinzufügen. Alternativ können Sie ihn in das Tool Ihrer Wahl schreiben und dort einfügen.

3. Konfigurieren Sie die Parameter der PrismAnalytics-Komponente:

Parameter	Beschreibung
<code>wd.prism.component.schema.varname</code>	Die Variable, die das JSON-Schema enthält, das die Daten beschreibt.

Parameter	Beschreibung
<code>wd.prism.component.table.name</code>	Der Name der Prism-Tabelle. Tabellennamen: • Müssen im Datenkatalog eindeutig sein. • Können bis zu 255 Zeichen enthalten. • Dürfen nur alphanumerische Zeichen und Unterstriche enthalten. • Müssen mit einem Buchstaben beginnen. • Darf nicht mit einem Unterstrich enden.
<code>wd.prism.component.client.id</code>	Die Client-ID, die dem API-Client von Workday zugewiesen wurde. Öffnen Sie den Bericht View API Clients und zeigen Sie dann das Register API Clients for Integrations an.
<code>wd.prism.component.client.secret</code>	Der geheime Client-Schlüssel, der dem API-Client von Workday zugewiesen wurde. Öffnen Sie den Bericht View API Clients und zeigen Sie dann das Register API Clients for Integrations an.
<code>wd.prism.component.refresh.token</code>	Das dem API-Client von Workday zugewiesene Aktualisierungstoken. Öffnen Sie den Bericht View API Clients und zeigen Sie dann das Register API Clients for Integrations an.
<code>wd.prism.component.operation</code>	Der für die Prism-Tabelle ausgeführte Vorgang. Ersetzen ist die einzige derzeit unterstützte Option. Er erstellt eine Tabelle oder ersetzt eine Tabelle mit demselben Namen.
<code>wd.prism.component.csv.input.varname</code>	(Optional) Die Variable, die die CSV-Eingabe enthält. Nicht erforderlich, wenn Sie die CSV-Daten zum Root-Teil der Nachricht hinzufügen. Anmerkung: Studio wählt diesen Parameter nicht automatisch zum Einbeziehen aus, wenn Sie eine PrismAnalytics-Komponente zu einer Assembly hinzufügen. Wählen Sie sie bei Bedarf manuell aus oder fügen Sie sie später auf dem Register Parameter der Komponente hinzu.
<code>wd.prism.component.api.hostname</code>	Workday-REST-API-Endpunkt. Öffnen Sie den Bericht View API Clients. Entfernen Sie die <code>https://</code> am Anfang und die <code>/ccx/api/v1/<tenant-name></code> am Ende. Beispiel: <code>i-09e55d32d61e2e287.workdaysuv.com</code> .
<code>wd.prism.component.table.description</code>	(Optional) Eine Beschreibung der Tabelle. Kann bis zu 255 Zeichen enthalten.
<code>wd.prism.component.table.description</code>	(Optional) Der Anzeigenname der Tabelle. Kann bis zu 255 Zeichen enthalten.

Parameter	Beschreibung
<code>wd.prism.component.table.field.variable-name</code>	(Optional) Definiert eine Prism-Tabelle. Schreiben Sie die Definition in eine Variable und übergeben Sie sie dann mithilfe von <code>vars['variable-name']</code> an diesen Parameter.

Nächste Maßnahme

Stellen Sie die Integration bereit und starten Sie sie.

Referenz: Eigenschaften der PrismAnalytics-Subassembly

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Id	<code>id</code>	Die eindeutige ID der PrismAnalytics- Untergruppe.
Antwort weiterleiten an	<code>routes-response-to</code>	Das Ziel der Antwortnachricht für die Untergruppe innerhalb der Gruppe.PrismAnalytics- Untergruppe.
Ausführung bei	<code>execute-when</code>	Eine Bedingung, die bestimmt, ob die PrismAnalytics- Untergruppe ausgeführt wird.
Endpunkt	<code>endpoint</code>	Die Adresse des Zielendpunkts für die Untergruppe. Die PrismAnalytics- Komponente unterstützt: <code>vm://</code> Endpunktadresse Beispiel: <code>vm://wcc/PrismAnalytics</code> .

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Store-Nachricht	<code>store-message</code>	Geben Sie " <i>Kein(e)</i> " an, es sei denn, der Workday-Support empfiehlt anders. Um Nachrichten zu speichern, verwenden Sie den Schritt store in Studio.
Transportklasse	<code>transport</code>	Gibt bei Bedarf eine alternative Implementierungsklasse für jede Baugruppe an. Sie können die Implementierungsklassen global ändern, indem Sie die Datei „WEB-INF/classes/spring/Assembly-bean.xml“ bearbeiten.
Dynamische Endpunkte ignorieren	<code>ignore-dynamic-endpoints</code>	Gibt an, ob der WSA-Wert (Dynamische WS-Adressierung)

Eigenschaft	XML-Attributname	Beschreibung
		im Header der eingehenden Nachricht ignoriert oder zum Überschreiben des Werts in der Endpoint- Eigenschaft des Transports verwendet werden soll. Verwenden Sie den Schritt <code>set-dynamic-endpoint</code>, um die WS-Adressierungsdetails anzugeben.
Klonanforderung	<code>clone-request</code>	Gibt an, ob die Verarbeitung zwischen Untergruppen verkettet werden soll. Setzen Sie die Eigenschaft auf Wahr, um eine Kopie der ursprünglichen Nachricht zu erstellen, sodass Subassemblys, die die Anforderungsnachricht ändern, keine Auswirkungen auf die anderen haben. Setzen Sie die Eigenschaft auf Falsch, um die ursprüngliche Nachricht zu übermitteln.
Abbruch weitergeben	<code>propagate-abort</code>	Gibt an, ob ein Abbruchkennzeichen von einer Untergruppe an die aufrufende Baugruppe weitergegeben wird. Eine Untergruppe stoppt die Verarbeitung, wenn sie als markiert <code>MediationContext</code> als abgebrochen.
Nicht konfigurierte Eigenschaften	<code>unset-properties</code>	Gibt an, ob alle lokalen Ende - Eigenschaften zurückgesetzt werden, nachdem der lokale Ende- Transport eine Untergruppe aufgerufen hat.

Register „Parameter“

Parameter	Beschreibung
<code>wd.prism.component.schema.varname</code>	Die Variable, die das JSON-Schema enthält, das die Daten beschreibt.
<code>wd.prism.component.table.name</code>	Die ID der Prism-Tabelle im Datenkatalog. Tabellennamen: • Müssen im Datenkatalog eindeutig sein. • Können bis zu 255 Zeichen enthalten. • Dürfen nur alphanumerische Zeichen und Unterstriche enthalten.

Parameter	Beschreibung
	• Müssen mit einem Buchstaben beginnen. • Dürfen nicht mit einem Unterstrich enden. Sie können diese Komponente nicht verwenden, um den Namen einer Prism-Tabelle zu aktualisieren, sondern nur, um eine Tabelle zu erstellen.
<code>wd.prism.component.client.id</code>	Die Client-ID, die dem API-Client von Workday zugewiesen wurde. Öffnen Sie den Bericht View API Clients und zeigen Sie dann das Register API Clients for Integrations an.
<code>wd.prism.component.client.secret</code>	Der geheime Client-Schlüssel, der dem API-Client von Workday zugewiesen wurde.
<code>wd.prism.component.refresh.token</code>	Das dem API-Client von Workday zugewiesene Aktualisierungstoken. Öffnen Sie den Bericht View API Clients und zeigen Sie dann das Register API Clients for Integrations an.
<code>wd.prism.component.operation</code>	Der für das Prism-Dataset ausgeführte Vorgang. Beispiele: Replace, Append. Replace erstellt eine Tabelle oder ersetzt eine Tabelle mit demselben Namen.
<code>wd.prism.component.csv.input.varname</code>	(Optional) Die Variable, die die CSV-Eingabe enthält. Nicht erforderlich, wenn Sie die CSV-Daten zum Root-Teil der Nachricht hinzufügen. Anmerkung: Studio wählt diesen Parameter nicht automatisch zum Einbeziehen aus, wenn Sie eine PrismAnalytics-Komponente zu einer Assembly hinzufügen. Wählen Sie sie bei Bedarf manuell aus oder fügen Sie sie später auf dem Register Parameter der Komponente hinzu.
<code>wd.prism.component.api.hostname</code>	Workday-REST-API-Endpunkt. Öffnen Sie den Bericht View API Clients. Entfernen Sie die <code>https://</code> am Anfang und die <code>/ccx/api/v1/<tenant-name></code> am Ende. Beispiel: <code>i-09e55d32d61e2e287.workdaysuv.com</code>.
<code>wd.prism.component.table.description</code>	(Optional) Eine Beschreibung der Tabelle. Kann bis zu 255 Zeichen enthalten.
<code>wd.prism.component.table.display.name</code>	(Optional) Der Anzeigename der Tabelle. Können bis zu 255 Zeichen enthalten. Sie können diese Komponente nicht verwenden, um den Anzeigenamen einer Prism-Tabelle zu aktualisieren, sondern nur, um einen Namen zu erstellen.
<code>wd.prism.component.table.field.varname</code>	(Optional) Definiert eine Prism-Tabelle. Schreiben Sie die Definition in eine Variable und übergeben

Parameter	Beschreibung
	Sie sie dann mithilfe von <code>vars['variable-name']</code> an diesen Parameter.

Fehlerbehandlungskomponenten

Konzept: Fehlerbehandlungskomponenten

Studio bietet drei verschiedene Fehlerbehandlungen in der Palette:

Name	Funktion
send-error	Ermöglicht die Verarbeitung des Fehlers mithilfe von Standardkomponenten und -schritten. Ein gängiges Muster ist die Verwendung von <code>send-error</code> , um den Fehler durch Aufrufen der allgemeinen Komponente <code>PutIntegrationMessage</code> an das Integrationsereignis zu melden.
log-error	Protokolliert den aktuellen Fehler im Ausgabeprotokoll. Enthält die Fehlermeldung und den Stack-Trace der Java-Ausnahme. Verwenden Sie diese Option, wenn Fehler im Integrationsereignis nicht einzeln offengelegt werden müssen, die Meldung aber bei Auftreten des Fehlers protokolliert werden soll. Vermeiden Sie die Protokollierung großer Meldungen, da diese die Performance beeinträchtigen und abgeschnitten werden können, was zu einem Verlust von Diagnosedaten führt.
custom-error-handler	Ermöglicht die Bereitstellung einer Spring-Bean, die Java zur Verarbeitung des Fehlers verwendet. Verwenden Sie diese Option, wenn die Fehlerbehandlung komplexen Java-Code erfordert.

Sie können eine Fehlerbehandlung als untergeordnetes Element einer Mediationskomponente oder als einzelnes Element in der Assembly hinzufügen. Zu Mediationskomponenten hinzugefügte Fehlerbehandlungen sind lokal. Fehlerbehandlungen für die Assembly sind global.

Wenn Workday in einer Studio-Integration auf einen Fehler stößt, wird die normale Verarbeitung gestoppt und die Kette der verarbeiteten Komponenten abgewickelt, wobei nach einer verfügbaren Fehlerbehandlung gesucht wird. Es wird immer zunächst versucht, Fehler lokal zu verarbeiten. Wenn keine lokale Fehlerbehandlung verfügbar ist, geht die Verantwortung auf die globale Fehlerbehandlung über. Verwenden Sie lokale Fehlerbehandlung, wenn Sie eine genauere Kontrolle darüber wünschen, was passiert, wenn Fehler auftreten, oder wenn Sie den detaillierten Kontext der Integrationsaktivität zum Zeitpunkt des Fehlers melden müssen. Stellen Sie sich globale Fehlerbehandlung als ausfallsicher vor.

Wenn jede Fehlerbehandlung ausgelöst wurde und ein Fehler immer noch nicht als behandelt markiert ist, verarbeitet Workday die Nachricht, generiert einen Fehler aus der Ausnahme und markiert die Assembly als beendet. Die Integration wird als „Abgeschlossen mit Fehlern“ beschrieben.

Bemerkenswertes Standardverhalten:

- Eine lokale Fehlerbehandlung verarbeitet nur Fehler, die in ihrer übergeordneten Mediationskomponente auftreten. Sie kann jedoch auch Fehler in nachgelagerten Komponenten

verarbeiten. Um dieses Verhalten zu aktivieren, setzen Sie die Eigenschaft `Nachgelagerte Fehler behandeln` der Mediationskomponente, die die Fehlerbehandlung enthält, auf `wahr`.

- Workday markiert Fehler als behandelt, wenn eine Fehlerbehandlung aufgerufen wird. Wenn Sie jedoch die Eigenschaft `Fehler erneut auslösen` einer Fehlerbehandlungsroutine auf `wahr` setzen, löscht Workday den Fehler nicht. Daher wird der Fehler vom nächsten vorgelagerten Fehlerhandler oder globalen Fehlerhandler im Bereich verarbeitet. Verwenden Sie dieses Muster, um die Details eines Fehlers lokal zu melden, aber den gesamten Fehler oder Abschluss auf einer höheren Ebene in der Baugruppe zu verarbeiten.
- Wenn Workday einen Fehler verarbeitet, startet es die Nachrichtenverarbeitung aus dem aufgerufenen Fehlerhandler neu und setzt den Antwortpfad zu dem Element fort, bei dem der Fehler ausgelöst wurde. Sie können jedoch angeben, dass die Verarbeitung stattdessen an dem Element fortgesetzt werden soll, in dem der Fehler aufgetreten ist. Um dieses Verhalten zu aktivieren, setzen Sie die Eigenschaft `Weiter nach Fehler` der Mediationskomponente, die die Fehlerbehandlung enthält, auf `Wiederherstellen`. Verwenden Sie dieses Verhalten in Situationen, in denen der von der Fehlerbehandlung aufgerufene Code die Fehlerbedingung korrigieren kann, sodass ein erneuter Versuch des fehlgeschlagenen Vorgangs erfolgreich durchgeführt werden kann.

Sie haben mehr Kontrolle über die Fehlerbehandlung, wenn Sie MVEL-Bedingungsausdrücke zur Ansicht Eigenschaften hinzufügen. Wenn Bedingungsdrücke vorhanden sind, ruft Workday den Fehlerhandler nur dann auf, wenn alle als `wahr` evaluiert werden.

Sie können das Auslösen von Fehlern mit erzwingen `context.setError` oder `context.SetException`-Methoden in MVE- oder Java-Code. Bei Verwendung von `setError`-Methode können Sie eine Fehler-ID konfigurieren, die von bedingten Ausdrücken im Fehlerhandler erkannt werden kann. Verwenden Sie diese Technik, wenn Sie einen Fehler lokal behandeln, aber einige Details der Fehlerart zur allgemeinen Behandlung an einen vorgelagerten Fehlerhandler weitergeben möchten.

Sie können Fehler auch mithilfe von Routing-Komponenten behandeln:

- Die Komponente `failover-strategy` versucht, eine Route zu verwenden. Wenn ein Fehler auftritt, wird die Failover-Route ausgeführt und der Fehler als `behandelt` markiert.
- Mit der benutzerdefinierten Strategie können Sie eine Quell-Bean bereitstellen. Quellbeans implementieren `RoutingStrategy`-Schnittstelle mit einer `isHandleError` Methode. Verwenden Sie diese Methode, um anzugeben, dass die Strategie Fehler behandelt.

Referenz: Komponenteneigenschaften für benutzerdefinierte Fehlerbehandlung

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
ID des Fehlerhandlers	<code>id</code>	Die eindeutige ID der benutzerdefinierten Komponente für Fehlerhandler innerhalb der Build.
Fehler erneut starten	<code>rethrow-error</code>	Gibt an, ob die benutzerdefinierte Komponente für Fehlerhandler den Fehler nach Abschluss der Verarbeitung an einen vorgelagerten Fehlerhandler zurückgibt. Der Standardwert ist <code>"false"</code> . Dies bedeutet, dass die benutzerdefinierte Komponente "Fehlerhandler" den Fehler allein verarbeitet und dann als "Behandelt" markiert.

Eigenschaft	XML-Attributname	Beschreibung
		Sie können Fehler auch durch Aufrufen von als behandelt markieren <code>setErrorHandled</code> Methode für <code>MediationContext</code> Das Objekt Beispiel: <code>mediationContext.setErrorHandled(true)</code>
Quell Bean	ref	Die ID einer spring bean, die eine der folgenden beiden Schnittstellenklassen implementiert: • <code>OnErrorHandler</code>. • <code>ConditionalOnErrorHandler</code>, wodurch verlängert wird <code>OnErrorHandler</code> mit einer booleschen Methode, <code>invokeHandleError(MediationContext context)</code>. Wenn diese Methode zurückgegeben wird <code>true</code>, Workday ruft auf <code>handleError(MediationContext context)</code> Methode Anmerkung: • Wenn Sie das implementieren <code>ConditionalOnErrorHandler</code>-Schnittstelle: Überschreiben Sie alle von Ihnen definierten Bedingungsausdrücke.

Register Ausdrücke

Eigenschaft	XML-Attributname	Beschreibung
Bedingungsausdruck	condition-expression	Ermöglicht die direkte Eingabe von MVal-Bedingungsausdrücken. Wenn alle Ausdrücke evaluiert werden <code>true</code> . Alternativ wird der Fehlerhandler aufgerufen, wenn Sie keinen Ausdruck angeben. Andernfalls wird der Fehlerhandler nicht aufgerufen, die Fehlerbehandlung wird in der Verarbeitungskette fortgesetzt und Workday fügt einen Auditprotokolleintrag

Eigenschaft	XML-Attributname	Beschreibung
		der Art hinzu ERROR_HANDLER_SKIPPED. Um einen neuen Ausdruck unterhalb einer vorhandenen Zeilen hinzuzufügen, klicken Sie auf Hinzufügen und geben Sie dann den Ausdruck ein. Um einen neuen Ausdruck oberhalb einer vorhandenen Zeile hinzuzufügen, platzieren Sie den Cursor in dieser Zeile und klicken Sie dann im Dropdown-Menü Hinzufügen auf Einfügen. Verwenden Sie die Schaltflächen rechts neben dem Register Ausdrücke, um Zeilen zu löschen oder neu anzuordnen.

Referenz: Eigenschaften der Komponente „Log-Error“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
ID des Fehlerhandlers	id	Die eindeutige ID der Protokollfehlerkomponente innerhalb der Build.
Fehler erneut starten	rethrow-error	Gibt an, ob die Protokollfehlerkomponente den Fehler nach Abschluss der Verarbeitung an einen vorgelagerten Fehlerhandler zurückgibt. Der Standardwert ist false. Dies bedeutet, dass die Protokollfehlerkomponente den Fehler allein verarbeitet und dann als behandelt markiert. Sie können Fehler auch durch Aufrufen von als behandelt markieren <code>setErrorHandled</code> Methode für <code>MediationContext</code> Das Objekt Beispiel: <code>mediationContext.setErrorHandled(true)</code>
Ebene	level	Die Protokollebene Log4J. Es gibt fünf Optionen: • debug • info

Eigenschaft	XML-Attributname	Beschreibung
		- warn - error - fatal

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Kategorie	category	Gibt einen Teil des Log4J-Protokollierungsnamens an. Der Standardwert ist DEFAULT.

Register Ausdrücke

Eigenschaft	XML-Attributname	Beschreibung
Bedingungsausdruck	condition-expression	Ermöglicht die direkte Eingabe von MVal-Bedingungsausdrücken. Wenn alle Ausdrücke evaluiert werden true. Alternativ wird der Fehlerhandler aufgerufen, wenn Sie keinen Ausdruck angeben. Andernfalls wird der Fehlerhandler nicht aufgerufen, die Fehlerbehandlung wird in der Verarbeitungskette fortgesetzt und Workday fügt einen Auditprotokolleintrag der Art hinzu ERROR_HANDLER_SKIPPED. Um einen neuen Ausdruck unterhalb einer vorhandenen Zeilen hinzuzufügen, klicken Sie auf Hinzufügen und geben Sie dann den Ausdruck ein. Um einen neuen Ausdruck oberhalb einer vorhandenen Zeile hinzuzufügen, platzieren Sie den Cursor in dieser Zeile und klicken Sie dann im Dropdown-Menü Hinzufügen auf Einfügen. Verwenden Sie die Schaltflächen rechts neben dem Register Ausdrücke, um Zeilen zu löschen oder neu anzuordnen.

Referenz: Eigenschaften der Komponente „Send-Error“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
ID des Fehlerhandlers	id	Die eindeutige ID der Sendefehlerkomponente innerhalb der Build.
Fehler erneut starten	rethrow-error	Gibt an, ob die Sendefehlerkomponente den Fehler an einen vorgelagerten Fehlerhandler zurückgibt, wenn die Verarbeitung abgeschlossen ist. Der Standardwert ist "false". Dies bedeutet, dass die Sendefehlerkomponente den Fehler allein verarbeitet und dann als "Behandelt" markiert. Sie können Fehler auch durch Aufrufen von als behandelt markieren <code>setErrorHandled</code> Methode für <code>MediationContext</code> Das Objekt Beispiel für MVE: <code>context.setErrorHandled(true)</code> .
Weiterleitung an	routes-to	Die ID des Aggregationselements, an das die Sendefehlerkomponente die Nachricht weiterleitet.

Register „Ausdrücke“

Eigenschaft	XML-Attributname	Beschreibung
Bedingungsausdruck	condition-expression	Ermöglicht die direkte Eingabe von MVal-Bedingungsausdrücken. Wenn alle Ausdrücke evaluiert werden <code>true</code> . Alternativ wird der Fehlerhandler aufgerufen, wenn Sie keinen Ausdruck angeben. Andernfalls wird der Fehlerhandler nicht aufgerufen, die Fehlerbehandlung wird in der Verarbeitungskette fortgesetzt und Workday fügt einen Auditprotokolleintrag der Art hinzu <code>ERROR_HANDLER_SKIPPED</code> . Um einen neuen Ausdruck unterhalb einer vorhandenen Zeilen hinzuzufügen, klicken Sie

Eigenschaft	XML-Attributname	Beschreibung
		auf Hinzufügen und geben Sie dann den Ausdruck ein. Um einen neuen Ausdruck oberhalb einer vorhandenen Zeile hinzuzufügen, platzieren Sie den Cursor in dieser Zeile und klicken Sie dann im Dropdown-Menü Hinzufügen auf Einfügen. Verwenden Sie die Schaltflächen rechts neben dem Register Ausdrücke, um Zeilen zu löschen oder neu anzuordnen.

Transformationsschritte

Text mit einer Textschemadatei in XML konvertieren

Vorbereitungen

Erstellen Sie eine Baugruppe, die einen Schritt textschema enthält. Rufen Sie den Beispieltext ab, der die Grundlage für Ihr Textschema bildet.

Warum und wann dieser Vorgang ausgeführt wird

Mit dem Schritt textschema können Sie Text mithilfe eines Textschemas in XML konvertieren. Um ein Textschema zu entwickeln, müssen Sie die Struktur der Daten kennen, die Sie umwandeln möchten. Der Prozess wird durch ein kleines Beispieldokument mit repräsentativen Daten erheblich vereinfacht. In diesem Thema wird davon ausgegangen, dass Sie über ein solches Dokument verfügen.

Sie können den Schritt textschema verwenden, um XML in Text zu konvertieren oder Dateien mit Trennzeichen zu konvertieren. In den meisten Fällen ist es jedoch einfacher, den Schritt XTT bzw. den Schritt csv-to-xml zu verwenden.

Prozedur

1. Klicken Sie in der Ansicht Eigenschaften des Schritts textschema auf Create Text Schema File.
2. Wählen Sie einen übergeordneten Ordner für die Textschemadatei aus und geben Sie einen Dateinamen an. Geben Sie bei Bedarf die Ziel-Namespace-URI an.
3. Studio öffnet den Editor für Textschemata und zeigt das neue Textschema in der Baumansicht an.
4. Schneiden Sie Ihren Beispieltext aus und fügen Sie ihn in den Bereich oben rechts im Textschema-Editor ein, um ihn zu ersetzen initial-text.

Anmerkung: Um den abschließenden Zeilenumbruch aus Eingabedateien zu entfernen, fügen Sie folgendes Attribut zum Root-Schemaelement hinzu: `ts:eofStrip="
"`.

5. Beginnen Sie mit der Beschreibung des Beispieltextes. Wählen Sie das erste Feld aus, klicken Sie mit der rechten Maustaste darauf und wählen Sie Create Field.

Anmerkung: Wenn Sie kein Beispiel für den zu konvertierenden Text haben, aber dessen Struktur kennen, können Sie die Ansicht Quelle des Textschemas direkt bearbeiten.

6. Geben Sie auf der Seite Identität des Assistenten zum Erstellen von Feldern eine grundlegende Beschreibung des Felds an. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Name	Ein Name für das Feld.
Vorkommen	Es gibt folgende Optionen: • Standard: Das Feld ist obligatorisch und kommt nur einmal vor. • Erforderlich: Identisch mit Standard. • Optional: Das Feld ist optional und kann nur einmal vorkommen. • Multiple Optional: Das Feld ist optional und kann mehrmals vorkommen. • Multiple Required: Das Feld ist obligatorisch und kann mehrmals vorkommen.
Start-Tag	Ein regulärer Ausdruck, der den Text angibt, der bewirkt, dass ein Element generiert wird. Beachten Sie, dass der Text, der mit diesem Ausdruck übereinstimmt, nicht Teil des Ausgabewerts ist. Wenn der übereinstimmende Text im Ausgabewert angezeigt werden soll, verwenden Sie den ts:format Option
End-Tag	Ein regulärer Ausdruck, der den Text angibt, der das Ende eines Elements markiert. Beachten Sie, dass der Text, der mit diesem Ausdruck übereinstimmt, nicht Teil des Ausgabewerts ist. Wenn der übereinstimmende Text im Ausgabewert angezeigt werden soll, verwenden Sie den ts:format Option
Abschneiden	Gibt an, ob überlaufende Daten in einem Feld mit fester Länge abgeschnitten oder ein Fehler ausgegeben werden soll. Der Standardwert ist true.
Trennzeichen	Ein regulärer Ausdruck, der ein Zeichen angibt, das wiederholte Felder trennt. Wird nur aktiviert, wenn Sie die Option Multiple Optional oder Multiple Required für Vorkommen auswählen.
Weglassen	Gibt an, ob das Feld in der Ausgabe bei einer Transformation text-to-XML oder XML-to-text weggelassen werden soll. Standardmäßig wird das Feld nicht weggelassen.

7. Beschreiben Sie auf der Seite Delimit des Assistenten zum Erstellen von Feldern, wie viele Zeichen aus dem Beispielttext ein Feld darstellen:

Option	Bezeichnung
Format	Ein regulärer Ausdruck, der gültige Textmuster für ein Feld identifiziert. Verwenden Sie diesen Ausdruck als Alternative zu Start-Tag und End-Tag, um sicherzustellen, dass der gesamte übereinstimmende Text in der Ausgabe angezeigt wird. Sie können ihn auch verwenden, wenn es sich um mehrfach auftretende Werte handelt, damit klar ist, wann das Ende der Sequenz erreicht ist.
Feste Länge	Gibt die Länge von Feldern mit fester Länge an. Enthält Anführungszeichen und Escapezeichen.
Quoted	Gibt an, ob Zeichenfolgenwerte als Zeichenfolgen in Anführungszeichen behandelt werden. Es gibt folgende Optionen: • Standard: • always: • always-including-empty: • optional: • never:
Padding	Ein regulärer Ausdruck, der Padding-Text in Feldern mit fester Länge identifiziert.
Preferred Padding	Gibt den Wert an, der verwendet werden soll, wenn der Padding-Ausdruck mit mehr als einer möglichen Zeichenfolge übereinstimmen kann.
Ausrichten	Gibt die Ausrichtung von Feldern mit fester Länge an, die Padding erfordern.

8. Beschreiben Sie auf der Seite Interpret des Assistenten zum Erstellen von Feldern, wie viele Zeichen aus dem Beispielttext ein Feld darstellen:

Option	Bezeichnung
Art	Der Datentyp des Felds.
Standardwert	Ein Standardwert für das Feld.
Datumsformat	Gibt ein Muster für Datumsangaben oder Datetimes an. Unterstützt alle von Java zugelassenen Muster SimpleDateFormat.
Datumssprache	Gibt die Datumssprache an, die bei der Interpretation von Datumsangaben verwendet werden soll. Stellt sicher, dass der Name in der entsprechenden Sprache verarbeitet wird, wenn das Datumsformat die Namen von Monaten verwendet. Der Standardwert ist EN.
Zweistelliges Jahr	Gibt einen Wert für die Interpretation von zweistelligen Jahresangaben an. Der Textschema-Parser interpretiert zweistellige Jahresangaben, die vor diesem Wert liegen,

Option	Bezeichnung
	als im aktuellen Jahrhundert. Beispiel: Wenn Sie Folgendes eingeben: 18, der Parser interpretiert 14 als 2014 und 96 als 1996.
Zahlenformat	Gibt ein Muster für Zahlen an. Unterstützt alle von Java zugelassenen Muster <code>DecimalFormat</code> .
Dezimaltrennzeichen	Gibt das Trennzeichen für Dezimalzahlen an.
Gruppierungstrennzeichen	Gibt das Tausender-Gruppierungstrennzeichen für Zahlen an.

9. Wiederholen Sie die Schritte 6–9 für jedes Feld im Beispieltext und fügen Sie neue Elemente in der Baumansicht hinzu. Wenn Sie alle erforderlichen Felddefinitionen erstellt haben, löschen Sie die verbleibenden Unparented untergeordnetes Element.
10. Um alle von Ihnen erstellten Elemente in einen komplexen Typ einzuschließen, wählen Sie sie aus, klicken Sie mit der rechten Maustaste auf sie und wählen Sie Elemente umschließen. Geben Sie dann einen Namen und einen Vorkommenswert für das Wrapper-Element an.
11. Wählen Sie in der Ansicht „Eigenschaften“ des Wrapper-Elements `/n` aus der Dropdown-Liste End-Tag .
12. Das Standard-Root-Element für das Schema wird benannt `File`. Geben Sie auf dem Register Schema in der Ansicht Eigenschaften des Elements einen aussagekräftigeren Namen ein.
13. In der Ansicht Eigenschaften der obersten Ebene `schema`-Element hat die Eigenschaft „Root-Element“ den Wert `tns:File`. Ersetzen `File` mit dem aussagekräftigen Namen, den Sie für das Root-Element ausgewählt haben.
14. Um zusätzliche Leerzeichen aus der generierten Ausgabe zu entfernen, wählen Sie die Ansicht Eigenschaften der obersten Ebene `schema` Element Wählen Sie `+` aus der Dropdown-Liste Padding . Speichern Sie das Schema.

Konzept: Transformationsschritte

Mit Transformationsschritten können Sie einzelne Vorgänge ausführen, die Nachrichten neu strukturieren, während sie Ihre Integration durchlaufen. Workday Studio bietet die folgenden Transformationsschritte in der Palette:

Transformationsschritt	Funktion	Anmerkungen
xslt	Transformiert ein XML-Dokument mithilfe eines XSL-Transformation (XSLT)-Stylesheets.	• Unterstützt XSLT 2.0. • Kompatibel mit Assemblys mit einer Versionsnummer bis 2017.05.
XSLT+	Transformiert ein XML-Dokument mithilfe eines XSL-Transformation (XSLT)-Stylesheets.	• Unterstützt die Streaming-Funktion von XSLT 3.0. • Kompatibel mit Assemblys mit einer Versionsnummer nach 2017.05.
fop	Generiert PDF- und RTF-Dokumente aus einer XML-Datei.	• Ruft die Implementierung des Apache Formatting Objects Processor (FOP) auf.

Transformationsschritt	Funktion	Anmerkungen
		• Implementiert eine Apache FOP XML-Konfigurationsdatei zur Registrierung benutzerdefinierter Schriften.
text-excel	Konvertiert Excel-Dateien in Dateien im Textformat mit durch Kommata getrennten Werten (CSV). Konvertiert außerdem CSV-Dateien in Excel-Dateien.	
textschemata	Konvertiert Text in XML oder XML in Text mithilfe einer Textschema-Datei.	
wrap-soap	Fügt die XML-Nachricht zur Weiterleitung an einen SOAP-basierten Webservice in einen SOAP-Umschlag ein.	
xml-to-java	Wandelt XML mithilfe von JAXB in Java-Objekte um.	• Implementiert XPath, um XML aus Nachrichten zu extrahieren. • Implementiert JAXB, um XML in Java-Objekte zu konvertieren. • Fügt konvertierte Java-Objekte in ein MediationContext.
json-transformer	Verwendet ein JSON-Dokument als Eingabe und wandelt es um, indem Werte mit JSON-Pfaden gemappt werden.	
xml-to-json	Konvertiert ein XML-Dokument in das JSON-Format.	
json-to-xml	Konvertiert ein JSON-Dokument in das XML-Format.	
java-to-xml	Marshallt Java-Objekte mithilfe von JAXB in XML.	• Extrahiert Java-Objekte aus MediationContext. • Implementiert JAXB, um Java-Objekte in XML zu konvertieren. • Implementiert XPath, um konvertiertes XML in Nachrichten einzufügen.
xml-to-csv	Konvertiert XML-Dokumente in das CSV-Format, während die Ein- und Ausgabedokumente	

Transformationsschritt	Funktion	Anmerkungen
	gestreamt werden, wodurch wenig Speicher benötigt wird.	
csv-to-xml	Konvertiert CSV-Dokumente in das XML-Format, während die Ein- und Ausgabedokumente gestreamt werden, wodurch weniger Speicher benötigt wird.	
Zeichenkonvertierung	Führt Zeichenkonversionen für den Textinhalt einer eingehenden Nachricht durch.	
mtable-builder	Erstellt und speichert eine mtable Objekt in a MediationContext Eigenschaft	• Unterstützt folgende untergeordnete Elemente: • csv-mtable-reader • json-mtable-reader
mtable-writer	Ruft eine ab mtable Objekt aus a MediationContext Eigenschaft Zeigt den Inhalt von a an mtable -Objekt in einer Ausgabemeldung.	• Unterstützt folgende untergeordnete Elemente: • csv-mtable-writer • json-mtable-writer
etv	Transformiert und validiert Elemente in einem XML-Dokument gemäß den Attributen, die an diese Elemente angehängt sind.	
xtr	Konvertiert XML gemäß den in XML definierten Attributen in Text.	

Referenz: Eigenschaften des Schritts „XSLT“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des XSLT-Mediationsschritts innerhalb der Build.
Eingabe	input	Gibt an, wo der XSLT-Mediationsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht.

Eigenschaft	XML-Attributname	Beschreibung
		Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der XSLT-Mediationsschritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.

Eigenschaft	XML-Attributname	Beschreibung
Url	url	Der Speicherort der vom XSLT-Mediationsschritt verwendeten XSLT-Datei.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des XSLT- Mediationsschritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der XSLT- Schritt ausgeführt wird.
Dateigespeicherte verwaltete Daten verwenden	use-file-backed-managed-data	Gibt an, ob die gesamte Ausgabe auf Ihre lokale Festplatte gestreamt werden soll, sobald eine Datenschwelle erreicht wird.
Meldungs-Eigenschaft	messages-property	Die MediationContext-Eigenschaft, die Sie in konfigurieren möchten List Das Objekt
Wiederverwendungsstrategie	reuse-strategy	Gibt die Wiederverwendungsstrategie für XSLT-Transformatoren an. Es gibt 4 Optionen: • Standard: Die MediationContext bestimmt automatisch Ihre Wiederverwendungsstrategie. • no-reuse: Ihr Server verwendet Transformationsobjekte nicht. • Vorlagen: Ihr Server erstellt eine Templates-Objekt, aus dem neue Transformationsobjekte erstellt werden. • Transfertransformator-pool: Ihr Server verwendet einen Pool von Transformationstransformatoren. Wir empfehlen diese

Eigenschaft	XML-Attributname	Beschreibung
		Option nur, wenn die Transformationsobjekte keine Werte im Cache speichern. Beispiel: Saxon speichert die Ergebnisse von im Cache document () Anrufe.
Verarbeitung sichern	secure-processing	Gibt an, ob eine Verarbeitung aktiviert werden soll, die die Sicherheit beeinträchtigen könnte. Wenn der Wert auf Wahr gesetzt ist, verarbeitet der xslt-Mediationsschritt XML sicher und legt Grenzwerte für XML-Konstrukte fest, um Bedingungen wie Denial-of-Service-Angriffe zu vermeiden. Wenn der Wert auf Falsch gesetzt ist, ignoriert der xslt-Mediationsschritt Sicherheitsprobleme wie Grenzwerte für XML-Konstrukte.
Trafotransformatorwerkstatt	transformer-factory	Der Klassenname der XSLT TransformerFactory Wird vom XSLT- Mediationsschritt verwendet. Beispiel: Der Wert für „Sachsen“ ist <code>net.sf.saxon.TransformerFactoryImpl</code>

Referenz: Eigenschaften des Schritts „XSLT+“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des XSLT+ - Mediationsschritts innerhalb der Build.
Eingabe	input	Gibt an, wo der XSLT+ - Mediationsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht.

Eigenschaft	XML-Attributname	Beschreibung
		Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der XSLT+ - Mediationsschritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.

Eigenschaft	XML-Attributname	Beschreibung
Url	url	Der Speicherort der vom XSLT+-Mediationsschritt verwendeten XSLT-Datei.
Verarbeitung sichern	secure-processing	Gibt an, ob eine Verarbeitung aktiviert werden soll, die die Sicherheit beeinträchtigen könnte. Wenn der Wert auf Wahr gesetzt ist, verarbeitet der XSLT+-Mediationsschritt XML sicher und legt Grenzwerte für XML-Konstrukte fest, um Bedingungen wie Denial-of-Service-Angriffe zu vermeiden. Wenn der Wert auf Falsch gesetzt ist, ignoriert der XSLT+-Mediationsschritt Sicherheitsprobleme wie Grenzwerte für XML-Konstrukte.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des XSLT + -Mediationsschritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der XSLT+ -Schritt ausgeführt wird.
Meldungs-Eigenschaft	messages-property	Die MediationContext -Eigenschaft, die Sie mit konfigurieren möchten List Das Objekt
Validierungsmodus für Schema	schema-validation-mode	Gibt an, ob die Eingaben und Ausgaben basierend auf den bereitgestellten Schemas mit dem Saxon XSLT-Processor validiert werden sollen.

Register „Schema-URLs“

Eigenschaft	XML-Attributname	Beschreibung
Schema URL	schema-url	Der Speicherort der Schemadatei, die der XSLT+-

Eigenschaft	XML-Attributname	Beschreibung
		Mediationsschritt verwendet, um das Ausgabedokument zu validieren.

Referenz: Eigenschaften des Schritts „FOP“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Fop-Mediationsschritts innerhalb der Baugruppe.
Eingabe	input	Gibt an, wo der Fop-Mediationsschritt Daten als Eingabe erhält. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der Fop-Mediationsschritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch

Eigenschaft	XML-Attributname	Beschreibung
		alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus <i>rootpart</i> der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in <i>MediationContext</i>.
Verfasser	author	Gibt eine an <i>author</i> -Wert in den Metadaten des Dokuments.
Basis-URL	base-url	Die Basis-URL, die zum Auflösen relativer URLs verwendet wird. Dieser Ort wird auch für relative Schriftpositionen verwendet, falls nicht vorhanden <i>font-base-url</i> ist angegeben. Dies kann ein relativer Pfad zum Projekt sein <i>WSAR-INF</i> oder eine absolute URL. Beispiel: <i>./fop</i>
Erstellungsdatum:	creation-date	Gibt an: <i>creation date</i> -Wert in den Metadaten des Dokuments. Geben Sie den Wert im US-Datumsformat ein. Beispiel: 5/14/08 4:50 PM.
Ersteller	creator	Gibt an: <i>creator</i> -Wert in den Metadaten des Dokuments.
Keywords	keywords	Gibt an: <i>keywords</i> -Wert in den Metadaten des Dokuments.
Titel	title	Gibt an: <i>title</i> -Wert in den Metadaten des Dokuments.
Benutzerkennwort	user-password	Ermöglicht die Festlegung eines Benutzerkennworts für PDF-

Eigenschaft	XML-Attributname	Beschreibung
		und RTF-Dokumente. Das Kennwort ermöglicht folgende Dokumentberechtigungen: • allow-copy-content • allow-edit-annotations • allow-edit-content • allow-print

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Fop- Mediation-Schritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Fop- Schritt ausgeführt wird.
"Inhalt kopieren" zulassen	allow-copy-content	Gibt an, ob Sie Inhalte aus dem Dokument kopieren und einfügen können, wenn Sie es über öffnen user - password.
„Anmerkungen bearbeiten“ zulassen	allow-edit-annotations	Gibt an, ob Sie Anmerkungen im Dokument bearbeiten können, wenn Sie es über den öffnen user - password.
Inhaltsbearbeitung zulassen	allow-edit-content	Gibt an, ob Sie Inhalte im Dokument bearbeiten können, wenn Sie es über öffnen user - password.
Drucken zulassen	allow-print	Gibt an, ob Sie das Dokument drucken können, wenn Sie es über verwenden user - password.
Konfigurations-XML	configuration-xml	Der Pfad zur Azure FOP XML-Konfigurationsdatei, der sich auf Ihrer lokalen Festplatte befinden muss und keine URL ist. Diese Konfigurationsdatei ist die einzige Möglichkeit, benutzerdefinierte Schriften zu registrieren.

Eigenschaft	XML-Attributname	Beschreibung
Format des Erstellungsdatums	creation-date-format	Das Datumsformat, das zum Parsen von verwendet wird creation-date XML-Attribut. Unterstützt folgende Werte: • full • long • medium • short
Erstellungsdatum - Land für Gebietsschema	creation-date-locale-country	Die country -Wert, der zum Parsen von verwendet wird creation-date XML-Attribut.
Sprache für Gebietsschema für Erstellungsdatum	creation-date-locale-language	Die language -Wert, der zum Parsen von verwendet wird creation-date XML-Attribut. Der Wert muss ein ISO-639-Sprachcode aus zwei Buchstaben sein.
Zeitformat für Erstellung	creation-time-format	Das Zeitformat, das zum Parsen von verwendet wird creation-date XML-Attribut. Unterstützt folgende Werte: • full • long • medium • short
Basis-URL für Schrift	font-base-url	Die Basis-URL, die zum Auflösen relativer Schrift-URLs verwendet wird. Beispiel: ./fop/fonts
Basis-URL für Bindestrich	hyphenation-base-url	Die Basis-URL, die zum Auflösen relativer URLs in Dateien mit Bindestrichmustern verwendet wird.
Kennwort des Eigentümers	owner-password	Ermöglicht es Ihnen, ein Eigentümerkennwort für PDF- und RTF-Dokumente festzulegen. Das Kennwort ermöglicht folgende Dokumentberechtigungen: • allow-copy-content • allow-edit-annotations • allow-edit-content • allow-print
Producer	producer	Gibt an: producer -Wert in den Metadaten des Dokuments.

Eigenschaft	XML-Attributname	Beschreibung
Ziellösung	target-resolution	Die Ausgabeauflösung in Punkten pro Zoll (Punkte pro Zoll) für Bitmap-Bilder, die von Bitmap-Renderern generiert wurden, z. B. vom TIFF-Renderer und von Workday B Tabellen.

Referenz: Eigenschaften des Schritts „Text-Excel“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Text-Excel- Vermittlungsschritts innerhalb der Build-gruppe.
Eingabe	input	Gibt an, wo der Text-Excel -Vermittlungsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, an wen der Text-Excel -Vermittlungsschritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich

Eigenschaft	XML-Attributname	Beschreibung
		rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Stil	style	Konvertierungsstil des Schritts „Text-Excel -Mediation“. Es gibt drei Optionen: • XLSX/XLS to Text: konvertiert eine Excel-Datei im XLSX- oder XLS-Format in eine CSV-Datei. • Text to XLS: konvertiert eine CSV-Datei in eine Excel-Datei im XLS-Format. • Text to XLSX: konvertiert eine CSV-Datei in eine Excel-Datei im XLSX-Format. Anmerkung: Diese Komponente wird bei der Konvertierung in oder vom XLSX-Format gestreamt.
Excel-Stil	excel-style	Gibt an, ob der Text-Excel-Mediationsschritt eine ganze Excel-Arbeitsmappe oder ein einzelnes Excel-Tabellenblatt konvertiert.

Eigenschaft	XML-Attributname	Beschreibung
Excel-Tabellenblattnummer	excel-sheet-number	Die Zieltabellenblattnummer, die in eine Excel-Datei konvertiert werden soll. Wenn Sie die Eigenschaft Excel-Stil auf Arbeitsmappe konfigurieren, ignoriert Workday die Einstellung Excel-Tabellenblattnummer für Ihren Text-Excel-Mediationsschritt.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Text-Excel- Schritts, der im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Text-Excel- Schritt ausgeführt wird.
Werte verwenden	usevalues	Gibt an, ob der Text-Excel - Vermittlungsschritt die zugrunde liegenden Werte der Tabelle verwendet und nicht deren formatierte Werte, die auf dem Bildschirm angezeigt werden.

relatedlinks

Referenz

[Workday 31 What's New Post: Workday Studio](#)

Referenz: Eigenschaften des Schritts „TextSchema“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Textschema -Mediationsschritts innerhalb der Build.
Eingabe	input	Gibt an, wo der Textschema-Mediationsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte:

Eigenschaft	XML-Attributname	Beschreibung
		• <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der Textschema-Mediationsschritt die Ausgabe leitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten.

Eigenschaft	XML-Attributname	Beschreibung
		<i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in <code>MediationContext</code>.
Url	url	Der Speicherort der vom Textschema-Mediationsschritt verwendeten XSD-Datei für das Textschema, angegeben als relativer Pfad oder absolute URL. Sie können einen MVE-Ausdruck verwenden, um die URL aus zu extrahieren <code>MediationContext</code> . Beispiel: <code>@{parts[1].xpath('/script/location')}</code>
Stil	style	Konvertierungsstil des Vermittlungsschritts für textschema. Es gibt zwei Optionen: <code>text2xml</code>: konvertiert Text in XML. <code>xml2text</code>: konvertiert XML in Text.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Textschema-Mediationsschritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt textschema ausgeführt wird.
Dateigespeicherte verwaltete Daten verwenden	use-file-backed-managed-data	Gibt an, ob die gesamte Ausgabe auf Ihre lokale Festplatte gestreamt werden soll, sobald eine Datenschwelle erreicht wird.

Referenz: Eigenschaften des Schritts „Wrap-Soap“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Eliminierungsschritts für Wrap-Soap innerhalb der Build-gruppe.
Eingabe	input	Gibt an, wo der Wrap-Soap - Mediationsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der Schritt der Wrap-Soap- Mediation die Ausgabe leitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht.

Eigenschaft	XML-Attributname	Beschreibung
		• <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Eliminierungsschritts für Wrap-Soap, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Wrap-Soap -Schritt ausgeführt wird.
Schnittstelle	interface	Die unmittelbar untergeordnete Ebene des Elements „SOAP Text“.
Sonntagsaktion	soap-action	Der Wert des SOAP-Aktions-Headers. Sie müssen diesen Wert für SOAP-HTTP-Anforderungen angeben.
SOIF-Version	soap-version	Die Version des SOAP-Umschlags.

Referenz: Eigenschaften des Schritts „XML-to-Java“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des XML-zu-Java -Vermittlungsschritts innerhalb der Build-gruppe.
Eingabe	input	Gibt an, wo der XML-zu-Java -Vermittlungsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Eigenschaft	property	Der Name der MediationContext - Eigenschaft, in die der XML-zu-Java -Vermittlungsschritt das Java-Objekt einfügt.
Pakete	packages	Die durch Semikolon getrennte Liste der Java-Pakete, die vom XML-zu-Java -Vermittlungsschritt benötigt werden, um XML mithilfe von JAXB in Java-Objekte zu konvertieren.
Unwapping JAXB Element aufheben	unwrap-jaxbelement	Gibt an, ob der XML-zu-Java -Vermittlungsschritt das Java-

Eigenschaft	XML-Attributname	Beschreibung
		Objekt im JAXB-Element speichert.
Namespaces	namespaces	Die von Ihrem XPath-Ausdruck verwendeten Namespaces.
XPath	xpath	Der XPath-Ausdruck, der XML aus der Nachricht extrahiert, um sie in Java-Objekte zu konvertieren.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Schritts „XML-zu-Java -Vermittlung“, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt "XML-zu-Java" ausgeführt wird.
XPath-Version	xpath-version	Die vom XML-zu-Java-Vermittlungsschritt verwendete XPath-Version. Beispiel: Version 1.0, Version 2.0 oder Version 3.0.

Referenz: Eigenschaften des JSON-Transformators

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Vermittlungsschritts „json-transformer“ innerhalb der Layouts.
Eingabe	input	Gibt an, wo der JSON-Transformer -Mediationsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn

Eigenschaft	XML-Attributname	Beschreibung
		die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der Vermittlungsschritt „json-transformer“ die Ausgabe leitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Vermittlungsschritts „json-transformer“, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt „json-transformer“ ausgeführt wird.

Register „Mappings“

Eigenschaft	XML-Attributname	Beschreibung
Json-Pfad	json-path	Gibt die Position im JSON-Baum an, an der der zugehörige Wert zugewiesen wird.
Fehlschlagen - Keine Übereinstimmung	fail-no-match	Bestimmt, was geschieht, wenn der angegebene JSON-Pfad keinen Knoten zurückgibt: • Wahr: Ein Fehler wird ausgelöst. • fail: Der fehlende Pfad wird ignoriert.

Referenz: Eigenschaften des Schritts „XML-to-JSON“**Register „Allgemein“**

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Schritts für die XML-zu-Json - Vermittlung innerhalb der Build-Tag-Build.
Eingabe	input	Gibt an, wo der XML-zu-Json -Vermittlungsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn

Eigenschaft	XML-Attributname	Beschreibung
		die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der XML-zu-Json-Vermittlungsschritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Schema URL	schema-url	Gibt eine relative URL an, damit beim Vermittlungsschritt „xml-to-json“ die XML-Schemadatei

Eigenschaft	XML-Attributname	Beschreibung
		als Ressource im Build-Webserviceprojekt abgerufen wird.
Schema	schema	Gibt an, wo der Schritt „xml-to-json“ das XML-Schema erhält. Es gibt fünf mögliche Werte: • message • soapbody • attachment • rootpart • variable
Attribute ignorieren	ignore-attributes	Gibt an, ob der Schritt "xml-to-json" Attribute ignoriert, wenn XML-Dokumente in JSON konvertiert werden.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Schritts „XML-zu-JSON -Vermittlung“, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt „xml-to-json“ ausgeführt wird.

Referenz: Eigenschaften des Schritts „JSON-to-XML“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des JSON-zu-XML- Vermittlungsschritts innerhalb der Build.
Eingabe	input	Gibt an, wo der JSON-zu-XML- Mediationsschritt Daten als Eingabe abruft. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus

Eigenschaft	XML-Attributname	Beschreibung
		rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der JSON-zu-XML -Vermittlungsschritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.

Eigenschaft	XML-Attributname	Beschreibung
Name des Root-Elements	root-element-name	Der Name des obersten Zeilenumbruch-Tags des XML-Dokuments.
Name des verschachtelten Arrays	nested-array-name	Ein Name für anonyme verschachtelte Arrays in der JSON-Eingabe.
Name des verschachtelten Objekts	nested-object-name	Ein Name für anonyme verschachtelte Objekte in der JSON-Eingabe.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des JSON-zu-XML -Vermittlungsschritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt „json-to-xml“ ausgeführt wird.

Referenz: Eigenschaften des Schritts „Java-to-XML“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Java-zu-XML- Vermittlungsschritts innerhalb der Build-Asset-Datei.
Ausgabe	output	Gibt an, an wen der Java-zu-XML -Vermittlungsschritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. <i>attach</i>: ein MIME-Anhang an die Nachricht. Geben

Eigenschaft	XML-Attributname	Beschreibung
		Sie den Index des Anhangs an. <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Eigenschaft	property	Der Name der MediationContext - Eigenschaft, aus der der Java-zu-XML -Vermittlungsschritt die Java-Objekte extrahiert.
Pakete	packages	Die durch Semikolon getrennte Liste der Java-Pakete, die vom Java-zu-XML- Mediationsschritt benötigt werden, um Java-Objekte mithilfe von JAXB in XML zu konvertieren.
Name des Root-Elements	root - element - name	Der Name des Root-Elements, das für das Marshallen von Java-Objekten in XML verwendet wird.
Namespace des Root-Elements	root - element - namespace	Der Namespace des Root-Elements, das für das Marshallen von Java-Objekten in XML verwendet wird.
Namespaces	namespaces	Die von Ihrem XPath-Ausdruck verwendeten Namespaces.
XPath	xpath	Der XPath-Ausdruck, der XML in Ihre Meldung einfügt.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Java-zu-XML- Vermittlungsschritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten,

Eigenschaft	XML-Attributname	Beschreibung
		für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt „Java-zu-XML“ ausgeführt wird.
XPath-Version	xpath-version	Gibt an, ob XPath Version 1.0, 2.0 oder 3.0 verwendet werden soll. Geben Sie bis Workday 11 den Wert 1 für Assembly-Versionen ein. Geben Sie die Werte 2 oder 3 für Assembly-Versionen ab Workday 12 ein. Sie können in Workday die Version dynamisch festlegen, indem Sie eine MVAL-Vorlage verwenden, die einen Systemeigenschaftswert einfügt. Beispiel: <code>@{props.get('the_version')}</code>

Referenz: Eigenschaften des Schritts „XML-to-CSV“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des XML-zu-CSV-Schritts innerhalb der Build.
Eingabe	input	Gibt an, wo der XML-zu-CSV-Schritt Daten als Eingabe abruft. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Index des Anhangs an. <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird

Eigenschaft	XML-Attributname	Beschreibung
		die gesamte Nachricht ausgewählt. <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der XML-zu-CSV- Schritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. <i>attach</i>: ein MIME-Anhang an die Nachricht. Geben Sie den Index des Anhangs an. <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Trennzeichen	separator	Das Trennzeichen, das nach jedem Dateneintrag in der CSV-Ausgabedatei eingefügt werden soll.
Zeilentrennzeichen	line-separator	Das neue Zeilentrennzeichen, das nach jeder Dateneingabe in der CSV-Ausgabedatei eingefügt werden soll. Unterstützt folgende Werte: - LF für Mac OS X und höher - CR für Mac OS Version 9 oder niedriger.

Eigenschaft	XML-Attributname	Beschreibung
		• CRLF für Microsoft Windows.
Header-Zeile schreiben	writeHeaderLine	Gibt an, ob der Schritt "xml-to-csv" die Namen der sequenziellen untergeordneten Elemente aus dem XML-Dokument als kommagetrennte Werte in die erste Zeile des Ausgabedokuments einfügt.
Format	format	Gibt an, ob Ihre CSV-Ausgabedokumente RFC-4180-konform sind. Sie können einen der folgenden Werte eingeben: • <i>rfc4180</i> • <i>simple</i> Wenn Sie <i>rfc4180</i> eingeben, entsprechen Ihre CSV-Ausgabedokumente den RFC-4180 Standards. Wenn Sie <i>simple</i> eingeben oder das Feld leer lassen, führt der Schritt xml-to-csv Folgendes aus: • Behandelt Apostrophe als alternative Anführungszeichen. • Schneidet anfängliche Leerzeichen in Werten ab. • Akzeptiert alle Zeilenenden (CR, LF oder CRLF).

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Schritts „XML-zu-CSV“, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt xml-to-csv ausgeführt wird.

Referenz: Eigenschaften des Schritts „CSV-to-XML“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des CSV-zu-XML- Schritts innerhalb der Build.
Eingabe	input	Gibt an, wo der CSV-zu-XML- Schritt Daten als Eingabe abruft. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Index des Anhangs an. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der CSV-zu-XML- Schritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Geben

Eigenschaft	XML-Attributname	Beschreibung
		Sie den Index des Anhangs an. <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Spaltennamen	colNames	Die Namen der Spalten, die in Ihrer XML-Ausgabedatei angezeigt werden. Geben Sie den Wert für diese Eigenschaft als Liste mit durch Kommas getrennten Namen ein. Beispiel: col1, col2, col3.
Namespace Prefix	namespacePrefix	Das Namespace-Präfix, das in Ihrer XML-Ausgabedatei angezeigt wird.
Namespace URI	namespaceURI	Die Namespace-URI, die im Root-Element in Ihrer XML-Ausgabedatei angezeigt wird. Beispiel: Wenn Sie eine URI von angeben <code>http://example.workday.com/SupplyChainManagement/DataTypes.xsd</code> und das Namespace-Präfix <code>ns</code>, erstellt der CSV-zu-XML- Schritt folgendes Root-Element in Ihrer XML-Ausgabedatei: <pre><ns:root xmlns:ns="http://example.workday.com/SupplyChainManagement/DataTypes.xsd"></pre>
Root-Name	rootName	Der Name des Root-Elements, das in Ihrer XML-Ausgabedatei angezeigt wird.

Eigenschaft	XML-Attributname	Beschreibung
Zeilenname	rowName	Die Namen der Zeilen, die in Ihrer XML-Ausgabedatei angezeigt werden. Geben Sie den Wert für diese Eigenschaft als Liste mit durch Kommas getrennten Namen ein. Beispiel: row1, row2, row3.
Trennzeichen	separator	Das Trennzeichen, das in Ihrer CSV-Eingabedatei verwendet wird.
Format	format	Gibt an, ob Ihre CSV-Eingabedatei RFC-4180-konform ist. Sie können einen der folgenden Werte eingeben: <i>rfc4180</i> <i>simple</i> Wenn Sie <i>rfc4180</i> eingeben, wird beim Schritt csv-to-xml davon ausgegangen, dass Ihre CSV-Eingabedatei den Standards RFC-4180 entspricht. Wenn Sie <i>simple</i> eingeben oder das Feld leer lassen, führt der Schritt csv-to-xml Folgendes aus: - Behandelt Apostrophe als alternative Anführungszeichen. - Schneidet anfängliche Leerzeichen in Werten ab. - Akzeptiert alle Zeilenenden (CR, LF oder CRLF).
Erste Zeile als Header verwenden	useFirstLineAsHeader	Gibt an, ob der CSV-zu-XML-Schritt die erste Zeile Ihrer CSV-Eingabedatei verwendet, um die Spalten zu benennen, die in Ihrer XML-Ausgabedatei angezeigt werden.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des CSV-zu-XML-Schritts, die im konsolidierten Protokoll

Eigenschaft	XML-Attributname	Beschreibung
		angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt CSV-zu-XML ausgeführt wird.

Referenz: Eigenschaften des Schritts „Character-Conversion“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Vermittlungsschritts für die Zeichenkonvertierung innerhalb dergruppe.
Eingabe	input	Gibt an, wo der Vermittlungsschritt für die Zeichenkonvertierung Daten als Eingabe erhält. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der Vermittlungsschritt für die Zeichenkonvertierung die

Eigenschaft	XML-Attributname	Beschreibung
		Ausgabe leitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Akzente entfernen	remove-accent s	Gibt an, ob der Schritt der Zeichenkonvertierung Zeichen mit Akzent in der Ausgabemeldung durch Zeichen ohne Akzent ersetzt. Beispiel: wenn eingestellt <code>true</code>, ersetzt der Schritt der Zeichenkonvertierung <code>é</code> durch <code>e</code>.
"Eols" entfernen	remove-eols	Gibt an, ob beim Schritt der Zeichenkonvertierung Zeilenende-Zeichen in der Ausgabemeldung entfernt werden.
Großbuchstaben	uppercase	Gibt an, ob beim Zeichenkonvertierungsschritt alle Zeichen in der Ausgabemeldung in Großbuchstaben konvertiert werden.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Vermittlungsschritts für die Zeichenkonvertierung, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt der Zeichenkonvertierung ausgeführt wird.
Steuerelemente ersetzen	replace-controls	Gibt an, ob der Zeichenkonvertierungsschritt ASCII-Steuerzeichen in der Ausgabemeldung durch Leerzeichen ersetzt.

Referenz: Eigenschaften des Schritts „Mtable-Builder“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Vermittlungsschritts „mtable-Builder“ innerhalb der Build.
Eingabe	input	Gibt an, wo der Vermittlungsschritt für den MTabellen-Builder Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine

Eigenschaft	XML-Attributname	Beschreibung
		MIME-Datei ist, wird die gesamte Nachricht ausgewählt. <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Mz-Tabellenname	mtable-name	Der Name der MediationContext Eigenschaft, in der gespeichert wird mtable Das Objekt

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Vermittlungsschritts für den MTabellen-Builder, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt "mtable-Builder" ausgeführt wird.

Referenz: Eigenschaften des Schritts „Mtable-Writer“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Vermittlungsschritts Mtable Writer in der Build.
Ausgabe	output	Gibt an, wohin der Vermittlungsschritt "mtable Writer" die Ausgabe leitet. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht.

Eigenschaft	XML-Attributname	Beschreibung
		• <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Mz-Tabellenname	mtable-name	Der Name der MediationContext Eigenschaft, in der gespeichert wird mtable Das Objekt

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Vermittlungsschritts "mtable Writer" , die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt "mtable- Writer " ausgeführt wird.

Referenz: Eigenschaften des Schritts „ETV“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des "etv" - Mediationsschritts innerhalb der Baugruppe.
Eingabe	input	Gibt an, wo der etv-Mediationsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte:

Eigenschaft	XML-Attributname	Beschreibung
		• <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, an wen der etv-Mediationsschritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten.

Eigenschaft	XML-Attributname	Beschreibung
		<i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in <code>MediationContext</code>.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	<code>description</code>	Eine Beschreibung des Eliminierungsschritts "etv", die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	<code>execute-when</code>	Eine Bedingung, die bestimmt, ob der Schritt "etv" ausgeführt wird.
Meldungen anhängen	<code>append-messages</code>	Gibt an, ob neue Nachrichten vorhandene Nachrichten ersetzen oder an diese angehängt werden.
Nachrichteneigenschaft	<code>message-property</code>	Der Name der <code>MediationContext</code> - Eigenschaft, die Nachrichten speichert, die vom Vermittlungsschritt "etv" erstellt wurden.

Referenz: Eigenschaften des Schritts „XTT“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	<code>id</code>	Die eindeutige ID des xtt-Mediationsschritts innerhalb der Build.
Eingabe	<code>input</code>	Gibt an, wo der xtt - Mediationsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn

Eigenschaft	XML-Attributname	Beschreibung
		die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wird die gesamte Nachricht ausgewählt. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der xtt-Mediationsschritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des xtt-Mediationsschritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der xtt- Schritt ausgeführt wird.
Meldungen anhängen	append-messages	Gibt an, ob neue Nachrichten vorhandene Nachrichten ersetzen oder an diese angehängt werden.
Nachrichteneigenschaft	message-property	Der Name der MediationContext Eigenschaft, die Nachrichten speichert, die vom xtt-Mediationsschritt erstellt wurden.

Untergeordnete Elemente des Mtable-Builder**Konzept: Untergeordnete Elemente des Mtable-Builders**

Der Schritt mtable-builder unterstützt folgende untergeordnete Elemente:

Untergeordnetes Element	Beschreibung
csv-mtable-reader	Parst CSV-Inhalte in eine mtable Das Objekt
json-mtable-reader	Parst JSON-Inhalte in eine mtable Das Objekt

Referenz: Eigenschaften des CSV-Mtable-Reader**Register „Allgemein“**

Eigenschaft	XML-Attributname	Beschreibung
Spaltennamen	column-names	Eine durch Kommas getrennte Liste von Spaltennamen, die in angezeigt werden mtable Das Objekt
Enthält Header-Zeile	contains-header-row	Gibt an, ob Ihre mtable Das Objekt "" enthält eine Header-Zeile.

Eigenschaft	XML-Attributname	Beschreibung
Indexspalte	index-column	Der Name der Indexspalte, die in angezeigt wird mtable Das Objekt
Spaltennamen normalisieren	normalise-column-names	Gibt an, ob die Spaltennamen in Ihrer mtable Objektanzeige mit Leerzeichen. Wenn eingestellt, true, die Spaltennamen in Ihrer mtable Objektanzeige mit Leerzeichen. Wenn eingestellt, false, die Spaltennamen in Ihrer mtable Objektanzeige ohne Leerzeichen.
Trennzeichen	separator	Das Trennzeichen zum Trennen von Feldern in Ihrem mtable Das Objekt
Header-Zeile verwenden	use-header-row	Gibt an, ob Ihre mtable Das - Objekt verwendet die Header-Zeile, um Spaltennamen anzuzeigen.

Referenz: Eigenschaften des JSON-Mtable-Reader

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Parsing-Ziel	parse-target	Der Name des JSON-Root-Objekts oder -Arrays, das Sie in parsen möchten mtable Das Objekt
Spalte "Id" ignorieren	ignore-id-column	Gibt an, ob die ID-Spalte aus Ihren JSON-Eingabedaten in angezeigt wird mtable Das Objekt Wenn eingestellt, true, ignoriert Workday die Spalte "ID" aus Ihren JSON-Eingabedaten in der mtable Das Objekt Wenn eingestellt, false, zeigt Workday die Spalte "ID" aus Ihren JSON-Eingabedaten in an mtable Das Objekt

Untergeordnete Elemente des Mtable-Writer

Konzept: Untergeordnete Elemente des Mtable-Writer

Der Schritt mtable-writer unterstützt folgende untergeordnete Elemente:

Untergeordnetes Element	Beschreibung
csv-mtable-writer	Zeigt den Inhalt von a an mtable -Objekt im CSV-Format.
json-mtable-writer	Zeigt den Inhalt von a an mtable -Objekt im CSV-Format.

Referenz: Eigenschaften des CSV-Mtable-Writer

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Spaltennamen	column-names	Eine durch Kommas getrennte Liste von Spaltennamen, die in angezeigt werden mtable Das Objekt
Zeilentrennzeichen	line-separator	Das neue Zeilentrennzeichen in Ihrer mtable Das Objekt Unterstützt folgende Werte: • LF für Android und Mac OS X • CR für Mac OS Version 9 oder niedriger. • CRLF für Microsoft Windows.
Trennzeichen	separator	Das Trennzeichen zum Trennen von Feldern in Ihrem mtable Das Objekt
Spaltennamen eingeben	write-column-names	Gibt an, ob Ihre mtable Das Objekt "" verwendet eine Header-Zeile, um Spaltennamen in anzuzeigen MediationContext.

Referenz: Eigenschaften des JSON-Mtable-Writer

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Feldname	field-name	Der Name des JSON-Felds, in dem der Inhalt von angezeigt wird mtable Das Objekt

Eigenschaft	XML-Attributname	Beschreibung
Tabellenblatt als Objekt schreiben	write-mtable-as-object	Gibt an, ob einzeilige mobile Tabellenobjekte wie folgt angezeigt werden: • JSON-Arrays, die ein Objektelement und mehrere Felder enthalten (Wahr). • JSON-Objekte mit mehreren Feldern (Falsch). Die Standardeinstellung ist „Falsch“.

Protokollierungs- und Ereignisschritte

Konzept: Protokollierungs- und Ereignisschritte

Mit Protokollierungsschritten können Sie benutzerdefinierte Protokollmeldungen für die Ausgabe an verschiedene Ziele erstellen. Workday Studio bietet die folgenden Schritte für Protokollierung und Ereignisse in der Palette an:

Name	Funktion	Anmerkungen
log	Workday erstellt eine benutzerdefinierte Log4j-Nachricht, die an Workday geschrieben wird.	• Drücken Sie Strg +Leertaste, wenn Sie einen MVEL-Ausdruck einfügen, um auf die Inhaltshilfe zuzugreifen.
cloud-log	Mit diesem Schritt können Sie benutzerdefinierte Nachrichten zu einem Integrationsprozess im HTML- oder CSV-Format ausgeben, die Sie an den Endpunkt Ihrer Wahl senden können.	• Der Schritt cloud-log generiert eine Ausgabedatei, leitet sie jedoch nicht weiter. Verwenden Sie einen anderen Schritt, um die Speicherung oder den Transfer der Datei zu verwalten. Beispiel: Verwenden Sie den Schritt Store, um die Ausgabedatei an Ihren Mandanten zu senden. • Die Ausgabedatei, die vom Schritt cloud-log

Name	Funktion	Anmerkungen
		erstellt wird, enthält immer fünf Spalten: • Zeitstempel • Ebene • Referenz-ID • Nachrichtenzusammenfassung • Nachrichtendetails Verwenden Sie das Register Extra Columns, um zusätzliche Informationen hinzuzufügen. Die maximale Größe für die Ausgabedatei beträgt 256 MB. Wenn Ihre Ausgabe 256 MB überschreitet, werden Sie benachrichtigt, dass die Ergebnisse abgeschnitten werden.

Referenz: Eigenschaften des Schritts „Log“

Register „Message Builder“

Der Message Builder vereinfacht für einige Assembly-Schritte das Definieren des Nachrichteninhalts.

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Protokollschritts innerhalb der Build.
Eingabe	input	Gibt an, wo der Protokollschritt Daten als Eingabe erhält. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an.

Eigenschaft	XML-Attributname	Beschreibung
		<i>rootpart</i>: Der Inhalt von <i>rootpart</i> einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. <i>variable</i>: Verwenden Sie eine benannte Variable aus dem <i>MediationContext</i>.
Ebene	level	Die Protokollebene Log4J. Es gibt fünf Optionen: - debug - info - warn - error - fatal

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Protokollschritts , die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Protokollschritt ausgeführt wird.
Kategorie	category	Teil des Logger-Namen. Wenn Sie keinen Wert angeben, verwendet Workday DEFAULT.

Sie können die verwenden *static-file* Untergeordnetes Element, um eine Zeichencodierungseigenschaft für eine Eingabedatei anzugeben. Beispiel:

```
<cc:log-message>
  <cc:static-file character-encoding="UTF-16" input-
file="logfile.html"/>
</cc:log-message>
```

Wenn Sie keine angeben *character-encoding* -Wert: Workday verwendet den Zeichensatz, der in angegeben ist *output-mimetype* Eigenschaft Wenn auch dort kein Zeichensatz angegeben ist, verwendet Workday UTF-8.

Referenz: Eigenschaften des Schritts „Cloud-Log“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Cloud-Protokollschritts in der Build.
Variablenname	variable-name	Der Name der Variable, in der die Ausgabedatei gespeichert wird. Sie können mehreren cloud-log-Schritten denselben Variablennamen geben, damit ihre Nachrichten zu einem einzigen Bericht verkettet werden. Sie können verschiedene Variablennamen verwenden, um separate Ausgabedateien zu erstellen.
Ebene	level	Schweregrad der Protokollmeldung. Es gibt sechs Optionen: • debug • info • warn • error • fatal • custom Wenn Sie custom auswählen, müssen Sie Ihre benutzerdefinierte Eigenschaft mit einem MVEL-Ausdruck im nebenstehenden Feld angeben.
Header	header	Der Text für den Datei-Header der Cloud-Protokolldatei. Wenn Sie dieses Feld leer lassen, verwendet Studio den Integrationsnamen als Header-Text. Wenn Ihre Assembly mehrere cloud-cog-Schritte mit definierten Headern enthält, verwendet Studio den Header des ersten Schrittes im Ablauf.
Nachricht	message	Der Text der Protokollmeldung.
Nachrichtendetails	message-details	Eine Beschreibung der Protokollmeldung.
Referenz-ID	reference-id	Die Referenz-ID eines Workday-Objekts. Hilfreich bei der Problembeseitigung.

Eigenschaft	XML-Attributname	Beschreibung
Bedingung	condition	Ein boolescher MSEL-Ausdruck, der die Fälle angibt, unter denen Workday die Meldung protokolliert. Wenn Sie keine Bedingung angeben, protokolliert Workday die Nachricht immer.
Ausgabedateityp	output-file-type	Die Art der Datei, die vom Cloud- Protokollschritt erstellt wird. Es gibt drei Optionen: • HTML • CSV • XLSX Die maximale Größe für die Ausgabedatei beträgt 256 MB. Wenn die Ausgabe 256 MB überschreitet, werden Sie von Studio benachrichtigt und die Ergebnisse werden abgeschnitten.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung der Protokollmeldung.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Cloud- Protokollschritt ausgeführt wird.

Register „Zusätzliche Spalten“

Eigenschaft	XML-Attributname	Beschreibung
Key	key	Die ID für eine zusätzliche benutzerdefinierte Nachricht. Eindeutig innerhalb einer Protokollvariablen.
Label	label	Das für den Benutzer sichtbare Label für eine zusätzliche benutzerdefinierte Nachricht.
Wert	value	Der Text einer zusätzlichen benutzerdefinierten Nachricht.

relatedlinks

Referenz[Workday 31 What's New Post: Workday Studio](#)

Schritte zur Nachrichtенbearbeitung

Konzept: Schritte zur Nachrichtенbearbeitung

Mit den Schritten zur Bearbeitung von Nachrichten können Sie den Inhalt von Nachrichten benutzerdefiniert anpassen, während sie Ihre Integration durchlaufen. Workday Studio bietet die folgenden Schritte zur Nachrichtенbearbeitung in der Palette:

Name	Funktion	Anmerkungen
copy	Extrahiert einzelne Teile eingehender Nachrichten und kopiert sie in verschiedene Teile ausgehender Nachrichten.	
javascript	Implementiert ein JavaScript-Skript oder eine Funktion, um die eingehenden Nachrichten zu bearbeiten.	
set-headers	Bearbeitet den Header einer Mediationsnachricht.	
write	Ermöglicht es Ihnen, benutzerdefinierte Daten zu erstellen und diese als Inhalt einer Mediationsnachricht festzulegen.	• Drücken Sie Strg +Leertaste, wenn Sie einen MVEL-Ausdruck einfügen, um auf die Inhaltshilfe zuzugreifen.

Referenz: Eigenschaften des Schritts „Copy“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Kopierschritts innerhalb der Build.
Eingabe	input	Gibt an, wo der Kopierschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an.

Eigenschaft	XML-Attributname	Beschreibung
		<i>rootpart</i>: Der Inhalt von <i>rootpart</i> einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. <i>variable</i>: Verwenden Sie eine benannte Variable aus dem <i>MediationContext</i>.
Geben Sie XPath ein	input-xpath	Ein XPath-Ausdruck, der einen übereinstimmenden Abschnitt von XML-Eingabedaten aus einem Nachrichtenteil oder einer Variable in extrahiert <i>MediationContext</i> .
Ausgabe	output	Gibt an, wohin der Vermittlungsschritt für das Kopieren die Ausgabe leitet. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich <i>rootpart</i> und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus <i>rootpart</i> der Nachricht. <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in <i>MediationContext</i>.
Ausgabe-XPath	output-xpath	Ein XPath-Ausdruck, der einen übereinstimmenden Abschnitt von XML-Ausgabedaten

Eigenschaft	XML-Attributname	Beschreibung
		aus einem Nachrichtenteil oder einer Variable in ersetzt MediationContext.
An Ausgabeelement anhängen	append-to-output-element	Gibt an, ob beim Kopierschritt das Eingabe-XML-Element an das Element angehängt wird, das mit dem übereinstimmt output-xpath-Wert
Namespaces	namespaces	Die von Ihrem XPath-Ausdruck verwendeten Namespaces im Format prefix namespace prefix2 namespace2. Überschreibt Namespace-Mappings in der Datei Asset.xml.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Kopierschritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Kopierschritt ausgeführt wird.
Stream XPath	stream-xpath	Gibt an, ob der Kopierschritt Stream-Unterstützung für einfache XPath-Ausdrücke bietet.
XPath-Version	xpath-version	Gibt an, ob XPath Version 1.0 oder 2.0 verwendet werden soll. Geben Sie den Wert 1 für Assembly-Versionen bis Workday 11 und 2 für Assembly-Versionen ab Workday 12 ein. Workday ignoriert diese Eigenschaft, wenn der Wert stream-xpath auf Wahr gesetzt ist. Sie können in Workday die Version dynamisch festlegen, indem Sie eine MVAL-Vorlage verwenden, die einen Systemeigenschaftswert

Eigenschaft	XML-Attributname	Beschreibung
		einfügt. Beispiel: <code>@{props.get('the_version')}</code> }).

Referenz: Eigenschaften des Schritts „JavaScript“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des JavaScript- Schritts innerhalb dergruppe.
Eingabe	input	Gibt an, wo der JavaScript-Schritt Daten als Eingabe erhält. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der JavaScript-Vermittlungsschritt die Ausgabe leitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden.

Eigenschaft	XML-Attributname	Beschreibung
		• <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Sie müssen den Index des Anhangs angeben. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Funktion	function	Ein Dateiname, der eine JavaScript-Funktion enthält, die sich im Verzeichnis ws/WSAR-INF Ihres Projekts befinden muss. Geben Sie einen Wert für die Eigenschaft Funktion oder Script an, aber nicht für beide. Sie können keinen Javascript-Schritt bereitstellen, wenn beide Werte konfiguriert sind.
Skript	script	Ein Dateiname mit einem JavaScript-Skript, das sich im Verzeichnis ws/WSAR-INF Ihres Projekts befinden muss. Geben Sie einen Wert für die Eigenschaft Funktion oder Script an, aber nicht für beide. Sie können keinen Javascript-Schritt bereitstellen, wenn beide Werte konfiguriert sind.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des JavaScript- Schritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion

Eigenschaft	XML-Attributname	Beschreibung
		des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der JavaScript- Schritt ausgeführt wird.

Referenz: Eigenschaften des Schritts „Set-Headers“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Schritts zum Festlegen von Headern in der Build-Build.
Ausgabe	output	Gibt an, wohin der Vermittlungsschritt "Header setzen" die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. <i>attach</i>: ein MIME-Anhang an die Nachricht. Geben Sie den Index des Anhangs an. <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Schritts „Header setzen“ . Wird im konsolidierten Protokoll angezeigt. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt „Header setzen“ ausgeführt wird.
Alle löschen	clear-all	Löscht alle Header aus den Vermittlungsmeldungen.
Alle ausschließen deaktivieren	clear-all-exclude	Eine Liste der Header-Namen, die nicht gelöscht werden, wenn "Alle löschen" aktiviert ist <code>true</code> .

Register „Header“

Header hinzufügen

Eigenschaft	XML-Attributname	Beschreibung
Name	name	Der Name des Headers, den Sie zu einer Vermittlungsnachricht hinzufügen möchten.
Wert	value	Der Wert des Headers, den Sie zu einer Vermittlungsnachricht hinzufügen möchten.

Header entfernen

Eigenschaft	XML-Attributname	Beschreibung
Name	name	Der Name des Headers, den Sie aus einer Vermittlungsnachricht entfernen möchten.

Referenz: Eigenschaften des Schritts „Write“**Register „Allgemein“**

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Schreisschritts innerhalb der Build.
Eingabe	input	Gibt an, wo der Schreisschritt Daten als Eingabe abruft. Es gibt 5 mögliche Werte:

Eigenschaft	XML-Attributname	Beschreibung
		• <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, an wen der Schreibvermittlungsschritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang an die Nachricht. Geben Sie den Index des Anhangs an. • <i>rootpart</i>: der Inhalt des Root-Teils einer MIME-Nachricht oder, wenn es sich nicht um eine MIME-Nachricht handelt, die Nachricht selbst. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten.

Eigenschaft	XML-Attributname	Beschreibung
		<i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in <code>MediationContext</code>.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	<code>description</code>	Eine Beschreibung des Schreisschritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	<code>execute-when</code>	Eine Bedingung, die bestimmt, ob der Schreisschritt ausgeführt wird.

Validierungsschritte

Konzept: Validierungsschritte

Mit Mediationsschritten zur Validierung können Sie Nachrichten evaluieren, während sie Ihre Integration durchlaufen. Workday Studio bietet die folgenden Schritte zur Validierung in der Palette:

Name	Funktion	Anmerkungen
<code>eval</code>	Ermöglicht die direkte Evaluierung von Ausdrücken.	
<code>validate</code>	Validiert eine Mediationsnachricht, um sicherzustellen, dass sie einem XML-Schema oder einer Dokumentartdefinition (DTD) entspricht, oder wohlgeformtes XML enthält.	- Wenn die Validierung fehlschlägt, wird je nach Wert der Filter-Eigenschaft entweder ein Fehler ausgelöst oder die Meldung verworfen. - Sie können eine benutzerdefinierte Fehlermeldung angeben. - Sie können ein Schema oder eine DTD angeben, um nur ausgewählte Elemente zu validieren, und diese Elemente aus einer Mediationsnachricht auswählen, indem Sie XPath-Ausdrücke angeben.

Name	Funktion	Anmerkungen
validate-exp	Validiert eine eingehende Meldung anhand eines oder mehrerer MVEL-Ausdrücke.	MVEL-Ausdrücke sind als Sequenz untergeordneter Elemente definiert.
validieren-xpfad	Validiert eine Nachricht mithilfe eines XPath-Ausdrucks, der als boolescher Wert evaluiert wird.	Die Validierung gilt als erfolgreich, wenn der zurückgegebene XPath-Ausdruck nicht leer ist.

Referenz: Eigenschaften des Schritts „Eval“

Register „Ausdrücke“

Eigenschaft	XML-Attributname	Beschreibung
Ausdruck	expression	Ermöglicht die direkte Eingabe von MVal-Bedingungsausdrücken. Um einen neuen Ausdruck unterhalb vorhandener Zeilen hinzuzufügen, klicken Sie auf Hinzufügen und geben Sie den Ausdruck ein. Um einen neuen Ausdruck oberhalb einer vorhandenen Zeile hinzuzufügen, platzieren Sie den Cursor in dieser Zeile und klicken Sie dann im Dropdown-Menü Hinzufügen auf Einfügen. Verwenden Sie die Schaltflächen rechts neben dem Register Ausdrücke, um Zeilen zu löschen oder neu anzuordnen.

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Evaluierungsschritts innerhalb der Baugruppe.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Evaluierungsschritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion

Eigenschaft	XML-Attributname	Beschreibung
		des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Evaluierungsschritt ausgeführt wird.
Sprache	language	Gibt die Ausdruckssprache an, die Sie verwenden möchten. MVEL ist die einzige derzeit unterstützte Sprache.

Referenz: Eigenschaften des Schritts „Validieren“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Validierungsschritts innerhalb der Baugruppe.
Eingabe	input	Gibt an, wo der Validierungsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Studio die gesamte Nachricht aus. <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Schema	schema	Der Speicherort der Schemadatei, die der Schritt zum Validieren der Meldung verwendet. Wenn Sie anstelle

Eigenschaft	XML-Attributname	Beschreibung
		einer MVEL-Vorlage einen Speicherort angeben, verwendet Workday Pool-Validierung.
Namespaces	namespaces	Die von Ihrem XPath-Ausdruck verwendeten Namespaces im Format <code>prefix namespace</code> <code>prefix2 namespace2</code> . Überschreibt Namespace-Mappings in der Datei <code>Asset.xml</code> .
Xpath	xpath	Ein XPath-Ausdruck, der Elemente für die Validierung auswählt.
Fehlermeldung	failure-message	Eine benutzerdefinierte Fehlermeldung, die die Standardfehlermeldung überschreibt, wenn ein Validierungsfehler auftritt. Fehlermeldungen werden nur angezeigt, wenn Sie die Eigenschaft Filter auf falsch setzen.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Validierungsschritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Validierungsschritt ausgeführt wird.
Modus	mode	Die Art der durchgeführten Validierung. Folgende Optionen stehen zur Verfügung: • <code>schema</code>: validiert anhand eines W3C-XML-Schemas. • <code>dtd</code>: validiert anhand einer Definition der Dokumentart. • <code>well-formed</code>: Verifiziert, ob das XML wohlgeformt ist.

Eigenschaft	XML-Attributname	Beschreibung
DTD	dtd	Der Speicherort der DTD-Datei, die der Schritt zum Validieren der Meldung verwendet.
Nachricht ersetzen	replace-message	Gibt an, ob der benutzerdefinierte Fehlermeldungssatz als Eigenschaft " Fehlermeldung" verwendet werden soll, wenn die Validierung fehlschlägt.
XPath-Version	xpath-version	Gibt an, ob XPath Version 1.0 oder 2.0 verwendet werden soll. Geben Sie den Wert 1 für Assembly-Versionen bis Workday 11 und 2 für Assembly-Versionen ab Workday 12 ein. Workday ignoriert diese Eigenschaft, wenn der Wert stream-xpath auf Wahr gesetzt ist. Sie können in Workday die Version dynamisch festlegen, indem Sie eine MVAL-Vorlage verwenden, die einen Systemeigenschaftswert einfügt. Beispiel: <code>@{props.get('the_version')}</code>
Filter	filter	Gibt an, ob die Verarbeitung der Vermittlungsnachricht unterbrochen werden soll, wenn die Validierung fehlschlägt. Der Standardwert ist Falsch, d. h. Studio leitet Validierungsfehler an die in der Komponente definierten Fehlerbehandlungen weiter. Wenn der Wert auf Wahr gesetzt ist, führt Studio Antwortschritte für die Komponente aus, verwirft jedoch die Nachricht und ruft keine weiteren Schritte auf. Es wird kein Fehler ausgelöst.

Referenz: Eigenschaften des Schritts „Validate-Exp“

Register „Ausdrücke“

Eigenschaft	XML-Attributname	Beschreibung
Wert	expression	Ermöglicht die direkte Eingabe von MVal-Bedingungsausdrücken. Um einen neuen Ausdruck unterhalb vorhandener Zeilen hinzuzufügen, klicken Sie auf Hinzufügen und geben Sie den Ausdruck ein. Um einen neuen Ausdruck oberhalb einer vorhandenen Zeile hinzuzufügen, platzieren Sie den Cursor in dieser Zeile und klicken Sie dann im Dropdown-Menü Hinzufügen auf Einfügen. Verwenden Sie die Schaltflächen rechts neben dem Register Ausdrücke, um Zeilen zu löschen oder neu anzuordnen.
Fehlermeldung	failure-message	Text, der zur Ausnahmenmeldung hinzugefügt werden soll, wenn ein Validierungsausdruck fehlschlägt.
Fehlernummer	error-number	Eine Fehlernummer, die zu der Ausnahmenmeldung hinzugefügt werden soll, die zurückgegeben wird, wenn ein Validierungsausdruck fehlschlägt.
Ersetzen	replace	Gibt an, ob die von Workday zurückgegebene Fehlermeldung durch den in der Eigenschaft "Fehlermeldung" angegebenen Text ersetzt werden soll.

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Schritts validation-exp in der Build.
Filter	filter	Gibt an, ob die Verarbeitung der Vermittlungsnachricht unterbrochen werden

Eigenschaft	XML-Attributname	Beschreibung
		soll, wenn die Validierung fehlschlägt. Der Standardwert ist Falsch, d. h. Workday leitet Validierungsfehler an die in der Komponente definierten Fehlerbehandlungen weiter. Wenn der Wert auf Wahr gesetzt ist, führt Workday Antwortschritte für die Komponente aus, verwirft jedoch die Nachricht und ruft keine weiteren Schritte auf. Es wird kein Fehler ausgelöst.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Schritts validation-exp , die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt validieren-exp ausgeführt wird.

Referenz: Eigenschaften des Schritts „Xpath validieren“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Schritts validieren-xpfad innerhalb der Build-gruppe.
Eingabe	input	Gibt an, wo der Schritt valid-xpfad Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält.

Eigenschaft	XML-Attributname	Beschreibung
		• <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Filter	filter	Gibt an, ob die Verarbeitung der Vermittlungsnachricht unterbrochen werden soll, wenn die Validierung fehlschlägt. Der Standardwert ist Falsch, d. h. Workday leitet Validierungsfehler an die in der Komponente definierten Fehlerbehandlungen weiter. Wenn der Wert auf Wahr gesetzt ist, führt Workday Antwortschritte für die Komponente aus, verwirft jedoch die Nachricht und ruft keine weiteren Schritte auf. Es wird kein Fehler ausgelöst.
Namespaces	namespaces	Die von Ihrem XPath-Ausdruck verwendeten Namespaces im Format <code>prefix namespace</code> <code>prefix2 namespace2</code> . Überschreibt Namespace-Mappings in der Datei <code>Asset.xml</code> .
Xpath	xpath	Ein XPath-Ausdruck, der Elemente für die Validierung auswählt.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Schritts <code>validieren-xpfad</code> , die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten,

Eigenschaft	XML-Attributname	Beschreibung
		für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt valid-xpfad ausgeführt wird.
XPath-Version	xpath-version	Gibt an, ob XPath Version 1.0 oder 2.0 verwendet werden soll. Geben Sie den Wert 1 für Assembly-Versionen bis Workday 11 und 2 für Assembly-Versionen ab Workday 12 ein. Workday ignoriert diese Eigenschaft, wenn der Wert stream-xpath auf Wahr gesetzt ist. Sie können in Workday die Version dynamisch festlegen, indem Sie eine MVAL-Vorlage verwenden, die einen Systemeigenschaftswert einfügt. Beispiel: <code>@{props.get('the_version')}</code>

Codierungsschritte

Konzept: Codierungsschritte

Mit Kodierungsschritten können Sie den Inhalt von codieren, decodieren, drücken und dekomprimieren `MediationContext`.

Workday Studio bietet folgende Kodierungsschritte in der Palette:

Name	Funktion	Anmerkungen
pgp-encrypt	Ruft einen öffentlichen Schlüssel ab und verschlüsselt ausgehende Daten mit PGP.	- Um Nachrichten mit in Workday gespeicherten Schlüsseln zu verschlüsseln, konfigurieren Sie einen Übermittlungsservice für Ihre Integration. Studio-Integrationen haben keinen Zugriff auf in Workday gespeicherte Schlüssel. - Sie können auch mithilfe des Protokolls „file:“ auf einen lokalen PGP-Schlüssel verweisen und diesen dann mit der Integration bereitstellen. Die

Name	Funktion	Anmerkungen
		Übermittlungsservicemethode ist jedoch die Best Practice für Studio-Integrationen.
pgp-entschlüsseln	Ruft einen privaten Schlüssel ab und entschlüsselt eingehende Daten, die mit PGP verschlüsselt wurden.	
base64-decode	Decodiert eine Base64-Codierung von Anhangdaten in das Binärformat.	
base64-encode	Codiert binäre Anhangdaten im Base64-Codierungsformat.	
ZIP	Transformiert den Root-Teil und die Anhänge aus dem MediationContext in eine ZIP-Datei.	
unzip	Extrahiert den Inhalt einer ZIP-Datei.	Wenn die ZIP-Datei mehr als eine Datei enthält, wird eine mehrteilige Struktur wie die Mediationsnachricht bevorzugt. Beispiel: Wenn Sie die Standardausgabeoption <i>message</i> verwenden, wird die erste Datei in der ZIP-Datei zum Root-Teil und die anderen Dateien in der ZIP-Datei werden zu Anhängen.
Komprimieren	Transformiert eine einzelne Datei in MediationContext in eine GZIP- oder BZIP2-Datei.	Im Gegensatz zum ZIP-Schritt unterstützt dieser Schritt keine Dateiarchivierung. Verwenden Sie diese Option, um einzelne Dateien zu komprimieren.
decompress	Extrahiert den Inhalt einer einzelnen GZIP- oder BZIP2-Datei.	

Konzept: PGP-Dateinamen

Wenn eine Klartextdatei mit PGP verschlüsselt wird, kann aus Sicherheitsgründen der verschlüsselten Datei ein Dateiname gegeben werden, der keinen Bezug zum Original hat und keinen Hinweis auf ihren Inhalt gibt. Der ursprüngliche Dateiname wird jedoch nicht verworfen. Er wird in die verschlüsselte Datei eingebettet, damit er später vom Entschlüsselungssystem des Empfängers abgerufen werden kann. Beispiel: Sie verschlüsseln eine Datei mit dem Namen „myPlaintext.txt“. Die resultierende Datei

heißt „myEncryption.pgp“. Teil des verschlüsselten Inhalts ist der ursprüngliche Dateiname. Wenn der Empfänger die Datei entschlüsselt, nennt sein System die resultierende Datei „myPlaintext.txt“.

Workday-Übermittlungs- und Abrufservices

Die Workday-Übermittlungs- und Abrufservices folgen nach Möglichkeit der typischen PGP-Dateinamenpraxis.

Der Workday-Übermittlungsservice bettet den Klartext-Dateinamen in die ausgehende PGP-Datei ein, damit er während der Entschlüsselung wiederhergestellt werden kann.

Wenn der Workday-Abrufservice eine eingehende PGP-Datei entschlüsselt, die an ein Integrationsereignis angehängt ist, verwendet er den eingebetteten Klartext-Dateinamen, um die entschlüsselte Datei zu benennen.

Manchmal hat eine eingehende PGP-Datei jedoch keinen eingebetteten Klartext-Dateinamen. In diesem Fall verwendet der Workday-Abrufservice den Namen der PGP-Datei als Basis für den Namen der entschlüsselten Datei:

- Wenn der PGP-Dateiname mit .pgp endet, entfernt der Abrufservice dieses Suffix aus dem entschlüsselten Dateinamen. Beispiel: Die PGP-Datei „myFile.csv.pgp“ wird als „myFile.csv“ entschlüsselt und die PGP-Datei „myFile.pgp“ wird als „myFile“ entschlüsselt.
- Wenn der PGP-Dateiname mit .xlsx endet, lässt der Abrufservice ihn unverändert. Beispiel: Die PGP-Datei „myFile.xlsx“ wird als „myFile.xlsx“ entschlüsselt.
- In allen anderen Fällen fügt der Abrufservice das Suffix „.plaintext“ hinzu. Beispiel: Die PGP-Datei „myFile.doc“ wird als „myFile.plaintext“ entschlüsselt.

Workday Studio

Wenn Sie den Schritt pgp-encrypt in Studio zum Verschlüsseln einer Nachricht verwenden, können Sie den Klartext-Dateinamen mithilfe der Schritteigenschaft Entschlüsselter Dateiname angeben. Studio bettet diesen Dateinamen dann wie gewohnt zum späteren Abrufen in die verschlüsselte Datei ein.

Ebenso verwendet der Schritt pgp-decrypt in Studio eingebettete Klartext-Dateinamen, um die entschlüsselte Datei zu benennen. Sie können die Schritteigenschaft Entschlüsselter Dateiname verwenden, um einen Dateinamen für Situationen anzugeben, in denen kein Dateiname eingebettet ist.

Problembeseitigung

Probleme können auftreten, wenn eine externe Partei, die eine Datei verschlüsselt, unbeabsichtigt den eingebetteten Klartext-Dateinamen auf einen anderen Wert als den vom Empfängersystem erforderlichen Wert festlegt. Da der PGP-Dateiname völlig unabhängig vom Klartext-Dateinamen ist, kann letzterer nicht abgeleitet werden. Der Empfänger hat in diesem Szenario keine Optionen.

Beispiel: Ein Abrufservice ist so konfiguriert, dass er eine Datei abrufen und entschlüsselt. Bei der Ausführung des Service wird eine verschlüsselte PGP-Datei mit dem Namen „ExternalStudents.pgp“ erfasst, aber eine Datei mit dem Namen „_CONSOLE“ ist an das Integrationsereignis zur Verarbeitung durch eine Integration angehängt. In diesem Fall hat das externe System „_CONSOLE“ als eingebetteten Klartext-Dateinamen festgelegt. Das Ergebnis ist, dass die Datei für Workday-Integrationen als „_CONSOLE“ identifiziert wird.

Die einzige Möglichkeit, solche Probleme zu vermeiden, besteht darin, sicherzustellen, dass die Absender von PGP-Dateien wissen, wie der erforderliche Klartext-Dateiname korrekt eingebettet wird.

Referenz: Eigenschaften des Schritts „PGP-Encrypt“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des pgp-encrypt- Schritts innerhalb der Build.
Eingabe	input	Gibt an, wo der Schritt "pgp-encrypt" Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der Schritt „pgp-encrypt“ die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht.

Eigenschaft	XML-Attributname	Beschreibung
		• <i>attach</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
ASCII-geschützt	ascii-armored	Gibt an, ob der Schritt „pgp-encrypt“ PGP-Daten im ASCII-geschützten Format (eine PGP-Version von Base64) oder in reinem Binärformat generiert. Die ASCII-geschützte Formatierung erstellt E-Mail-optimierten Text, einschließlich Header und Fußzeile.
Zertifikat	certificate	Das für die Verschlüsselung verwendete Zertifikat, das über eine der folgenden Arten von URLs bereitgestellt wird: • Eine Anhang-URL zum Abrufen des öffentlichen Schlüssels vom Mandanten. Sie müssen dem Muster folgen <code>attachment:certificate/WIDvalue</code>, wobei WIDvalue ist die WID des im Mandanten gespeicherten öffentlichen Schlüssels. Beispiel: Sie können den Bezug mithilfe eines Integrationsanhangs im Formular konfigurieren <code>attachment:certificate/{intsys.getAttributeReferencedEncryption Key,'WID'}}</code> wobei PGP Encryption Key

Eigenschaft	XML-Attributname	Beschreibung
		ist der Name des Integrationsattributs. • Eine mctx-URL zum Abrufen des öffentlichen Schlüssels aus der Kontextvariable, in der die Integration gespeichert hat. Beispiel: <code>mctx:vars/MyPublicKey</code> wobei <code>MyPublicKey</code> ist der Name der Kontextvariablen. • Der Dateiname eines öffentlichen Schlüssels, der im Ordner <code>WSAR-INF</code> der Baugruppe oder in einem ihrer Unterordner gespeichert ist. Beispiele: <code>PublicKey.asc</code> oder <code>/keys/PublicKey.asc</code>.
Einschließlich Integritätsprüfung	<code>containing-integrity-check</code>	Gibt an, ob der Schritt "pgp-encrypt" eine Nachrichtensignatur zur Integritätsprüfung durch den Empfänger generiert.
Entschlüsselter Dateiname	<code>decrypted-filename</code>	Einen Namen für die entschlüsselte Datei. Beispiel: <code>yourname.plaintext.mail</code> . Workday schließt den angegebenen Wert in die Hash-Zeichenfolge ein.
Pgp26x-kompatibel	<code>pgp26x-compatible</code>	Gibt an, ob der Schritt "pgp-encrypt" verschlüsselte Daten generiert, die abwärtskompatibel mit der älteren Version <code>pgp26x</code> sind.
Kennwort für privaten Schlüssel	<code>private-key-passphrase</code>	Gibt die Passphrase für den privaten Schlüssel an.
Privater Signaturschlüssel	<code>signing-private-key</code>	Der private Schlüssel zum Signieren von Nachrichten als Dateiname unter Verwendung von <code>file://</code> Protokoll Anmerkung: Die Best Practice für Workday Studio-Integrationen ist, einen Übermittlungsservice anstelle des Schritts „pgp-encrypt“ für die Verschlüsselung zu verwenden.

Eigenschaft	XML-Attributname	Beschreibung
Textmodus	textmode	Angeben <code>true</code> in, um anzugeben, dass die zu verschlüsselnden Daten im Textformat mit den Zeilenabschlüssen <code><CR><LF></code> vorliegen. Anmerkung: Wenn Sie angeben <code>true</code> für Daten, die keine Zeilenabschlüsse <code><CR><LF></code> enthalten, meldet Workday einen Fehler.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Schritts „pgp-encrypt“, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, auf die er angewendet wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt "pgp-encrypt" ausgeführt wird.

[relatedlinks](#)

Referenz

[Workday 31 What's New Post: Workday Studio](#)

Referenz: Eigenschaften des Schritts „PGP-Decrypt“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des pgp-decrypt- Schritts innerhalb der Build.
Eingabe	input	Gibt an, wo der Schritt "pgp-decrypt" Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn

Eigenschaft	XML-Attributname	Beschreibung
		die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der Schritt „pgp-decrypt“ die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Privater Schlüssel	private-key	Der Dateiname oder die URI für den privaten Workday-Schlüssel. Verwenden Sie die

Eigenschaft	XML-Attributname	Beschreibung
		<code>file://</code> -Protokoll, um es als Dateinamen anzugeben.
Passwörter	<code>passphrase</code>	Gibt die Passphrase für den privaten Schlüssel an.
Entschlüsselter Dateiname	<code>decrypted-filename</code>	Legt fest <code>content-disposition</code> Attribut für die Ausgabe des Schritts „pgp-decrypt“ oder, wenn die Ausgabe eine Variable ist, den Dateinamen. Wenn der Schritt „pgp-encrypt“ jedoch einen Wert für den entschlüsselten Dateinamen angibt, hat dieser Wert Vorrang.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	<code>description</code>	Eine Beschreibung des Schritts „pgp-decrypt“, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	<code>execute-when</code>	Eine Bedingung, die bestimmt, ob der Schritt „pgp-decrypt“ ausgeführt wird.

Referenz: Eigenschaften des Schritts „Base64-Decode“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	<code>id</code>	Die eindeutige ID des Base64-Decodierungsschritts innerhalb der Build.
Eingabe	<code>input</code>	Gibt an, wo der Base64-Decodierungsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich <i>rootpart</i> und Anhänge. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus <i>rootpart</i> der Nachricht. Nur anwendbar, wenn

Eigenschaft	XML-Attributname	Beschreibung
		die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der Base64-Decodierungsschritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Schrittes zur Base64-Decodierung , die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt zur Base64-Decodierung ausgeführt wird.

Referenz: Eigenschaften des Schritts „Base64-Encode“**Register „Allgemein“**

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Base64-Codierungsschritts innerhalb der Build.
Eingabe	input	Gibt an, wo der Base64-Codierungsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.

Eigenschaft	XML-Attributname	Beschreibung
Ausgabe	output	Gibt an, wohin der Base64-Codierungsschritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.

Register Erweitert

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Base64-Codierungsschritts , die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt „base64-Codierung“ ausgeführt wird.

Referenz: Eigenschaften des Schritts „Zip“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Zip - Schritts innerhalb der Build.
Eingabe	input	Gibt an, wo der ZIP- Schritt Daten als Eingabe bezieht. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der ZIP- Schritt die Ausgabe leitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-

Eigenschaft	XML-Attributname	Beschreibung
		Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Format	format	Das Format der gezippten Datei. Es gibt folgende Optionen: <i>zip</i> <i>tar</i> Anmerkung: Das TAR-Format unterstützt die Archivierung, aber keine Komprimierung. Verwenden Sie zum Komprimieren von TAR-Dateien einen der folgenden Befehle: <i>gzip</i>, das die Dateierweiterung <i>.tar</i> durch <i>.tar.gz</i> oder <i>.tgz</i> ersetzt <i>bzip2</i>, das die Dateierweiterung <i>.tar</i> durch <i>.tar.bz2</i> ersetzt Sie können TAR-Archive, die im zip-Schritt erstellt wurden, mit den Schritten compress bzw. decompress komprimieren und dekomprimieren.
Ebene	level	Komprimierungsgrad im Bereich von 0 bis 9, wobei 9 die maximale Komprimierung bedeutet. Der Standardwert ist -1.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des ZIP-Schritts. Wird im konsolidierten Protokoll angezeigt. Mit diesem Feature können Sie die Funktion des Schritts und die Daten,

Eigenschaft	XML-Attributname	Beschreibung
		für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der ZIP- Schritt ausgeführt wird.
Kommentar	comment	Ein Kommentar, der in die ZIP-Datei eingebettet ist. Der Kommentar ist eine beliebige ASCII-Zeichenfolge mit bis zu 65.535 Bytes.

Referenz: Eigenschaften des Schritts „Unzip“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Entnahmeschritts in der Build.
Eingabe	input	Gibt an, wo beim Entpacken von Daten Daten als Eingabe abgerufen werden. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der Entpacken-Schritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte:

Eigenschaft	XML-Attributname	Beschreibung
		• <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Format	format	Das Format der gezippten Datei. Es gibt folgende Optionen: • <i>zip</i> • <i>tar</i> Anmerkung: Das TAR-Format unterstützt die Archivierung, aber keine Komprimierung. Verwenden Sie zum Komprimieren von TAR-Dateien einen der folgenden Befehle: • <i>gzip</i>, das die Dateierweiterung <i>.tar</i> durch <i>.tar.gz</i> oder <i>.tgz</i> ersetzt • <i>bzip2</i>, das die Dateierweiterung <i>.tar</i> durch <i>.tar.bz2</i> ersetzt Sie können TAR-Archive, die im zip-Schritt erstellt wurden, mit den Schritten compress bzw. decompress komprimieren und dekomprimieren.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung zum Entpacken MediationContext.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt zum Entpacken ausgeführt wird.
Dateiname	filename	Der Name der Datei, die beim Entpacken des Inhalts erstellt wird.

Referenz: Eigenschaften des Schritts „Compress“**Register „Allgemein“**

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Komprimierungsschritts innerhalb der Baugruppe.
Eingabe	input	Gibt an, wo der Komprimierungsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der Komprimierungsschritt die

Eigenschaft	XML-Attributname	Beschreibung
		Ausgabe leitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Format	format	Das Format der gezippten Datei. Die Optionen sind <i>gzip</i> und <i>bzip2</i> . Das Standardformat ist <i>gzip</i> .

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Komprimierungsschritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Komprimierungsschritt ausgeführt wird.

Eigenschaft	XML-Attributname	Beschreibung
Dateiname	filename	Der Name der Datei, die beim Entpacken des Inhalts erstellt wird.

Referenz: Eigenschaften des Schritts „Decompress“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Dekomprimierungsschritts in der Baugruppe.
Eingabe	input	Gibt an, wo der Dekomprimierungsschritt Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der dekomprimierte Schritt die Ausgabe leitet. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch

Eigenschaft	XML-Attributname	Beschreibung
		alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Format	format	Das Format der gezippten Datei. Die Optionen sind <i>gzip</i> und <i>bzip2</i> . Das Standardformat ist <i>gzip</i> .

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Dekomprimierungsschritts , die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der dekomprimierungsschritt ausgeführt wird.
Dateiname	filename	Der Name der Datei, die beim Entpacken des Inhalts erstellt wird.

Andere Schritte

Konzept: Andere Schritte

Mit Schritten können Sie einzelne Vorgänge an Nachrichten durchführen, während diese Ihre Integration durchlaufen. Workday Studio bietet die folgenden Anderen Schritte in der Palette:

Name	Funktion
custom	Ermöglicht die Anwendung eines mit Spring Beans geschriebenen benutzerdefinierten Codes auf eingehende Nachrichten.
set-dynamic-endpoint	Konfiguriert WS-Adressierungs- (WSA) Endpunktinformationen für einen ausgehenden Transport.
store	Erzeugt eine dauerhafte Kopie eines Dokuments, die für andere Benutzer verfügbar ist und/oder als Anhang des Integrationsereignisses aufgerufen werden kann. Die Größenbeschränkung für einzelne Nachrichten für Speicherschritte beträgt 1 GB (komprimiert). Der kumulative Grenzwert beträgt 3 GB (komprimiert). Workday beendet Integrationen, die diese Grenzwerte überschreiten. Verwenden Sie nach Möglichkeit Splitter oder Streaming, um große Dateien in Ihren Integrationen aufzuteilen.
retrieve	Ruft eine dauerhafte Kopie eines Dokuments ab, das zuvor in einem store-Schritt gespeichert wurde.
xmldiff	Vergleicht zwei XML-Eingabedokumente.
enqueue-message	Organisiert Nachrichten in einer Warteschlange.

Referenz: Eigenschaften des Schritts „Custom“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Gibt die eindeutige ID des benutzerdefinierten Schritts innerhalb der Build an.
Eingabe	input	Gibt an, wo der benutzerdefinierte Schritt Daten als Eingabe abruft. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge.

Eigenschaft	XML-Attributname	Beschreibung
		• <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der benutzerdefinierte Schritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.

Eigenschaft	XML-Attributname	Beschreibung
Methodenname	method - name	Der Name der Java-Methode in Ihrer benutzerdefinierten Java-Klasse.
Quell Bean	ref	Die benutzerdefinierte springbean-Java-Klasse, die durch den benutzerdefinierten Schritt implementiert wird.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des benutzerdefinierten Schritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute - when	Eine Bedingung, die bestimmt, ob der benutzerdefinierte Schritt ausgeführt wird.

Referenz: Eigenschaften des Schritts „Set-Dynamic-Endpoint“**Register „Allgemein“**

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Schritts „Dynamischer Endpunkt festlegen“ innerhalb der Build-gruppe.
Endpunkt	endpoint	Die Adresse des dynamischen Endpunkts.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Schritts „dynamische Endpunkte festlegen“, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute - when	Eine Bedingung, die bestimmt, ob der Schritt „Dynamischer

Eigenschaft	XML-Attributname	Beschreibung
		Endpunkt definieren“ ausgeführt wird.

Referenz: Eigenschaften des Schritts „Store“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Filialschritts in der Baugruppe.
Eingabe	input	Gibt an, wo der Geschäftsschritt Daten als Eingabe erhält. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der Geschäftsschritt die Ausgabe leitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht.

Eigenschaft	XML-Attributname	Beschreibung
		• <i>attach</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext. Der Speicherschnitt speichert den MIME-Typ einer Nachricht zusammen mit der Nachricht selbst. Beispiel: <i>text/plain</i>. Wenn Sie die Nachricht mit einem retrieve-Schnitt abrufen, markiert Workday sie automatisch mit dem gespeicherten MIME-Typ, damit spätere Schritte sie korrekt verarbeiten können. Die Größenbeschränkung für einzelne Nachrichten für Speicherschnitte beträgt 1 GB (komprimiert). Der kumulative Grenzwert beträgt 3 GB (komprimiert). Workday beendet Integrationen, die diese Grenzwerte überschreiten. Verwenden Sie nach Möglichkeit Splitter oder Streaming, um große Dateien in Ihren Integrationen aufzuteilen.
Einzug	collection	Der Name der Sammlung, in der Ihr Dokument gespeichert wird.
Läuft ab in	expiresIn	Die Zeitperiode, bevor Ihr Dokument abläuft. Geben Sie den Wert im XSD-Format für die Dauer an. Beispiel: Um 1 Jahr, 3 Monate, 5 Tage, 8 Stunden, 45 Minuten und 10 Sekunden anzugeben, geben Sie Folgendes ein: P1Y3M5DT8H45M10S.

Eigenschaft	XML-Attributname	Beschreibung
Zusammenfassung	summary	Eine Zusammenfassung Ihres Dokuments. Wird in Ihrem Workday-Mandanten angezeigt.
Titel	title	Der Titel Ihres Dokuments.
Dokumentreferenz erstellen	createDocumentReference	Gibt an, ob der Geschäftsschritt Workday automatisch aufruft, einen Dokumentbezug zu erstellen.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Geschäftsschritts , die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Geschäftsschritt ausgeführt wird.
Ausschluss von Inhalten	contentDisposition	Der Header für die Inhaltsverteilung für Ihr Dokument. Überschreibt den Eigenschaftswert title, falls angegeben.
Blobitory-URI	blobitoryURI	Die URI der cc-blobitory - Service, der in Ihrem Workday-Mandanten implementiert ist. Beispiel: <code>https://machine-name.workday.com:8080/ccx/cc-blobitory</code> .
Läuft ab am	expiresOn	Das Datum und die Uhrzeit, an denen Ihr Dokument abläuft. Überschreibt den Eigenschaftswert Expires In , falls angegeben. Geben Sie den Wert für diese Eigenschaft in ein CCYY-MM-DDThh:mm:ssZ Format. Beispiel: Um am 31. Dezember 2009 23,59 Uhr anzugeben, geben Sie Folgendes ein: <code>2009-12-31T23:59:00Z</code> .

Eigenschaft	XML-Attributname	Beschreibung
Schema	schema	Die URI für das Schema, das Ihr Dokument referenziert. Es gibt zwei mögliche Werte: • http://www.w3.org/2005/Atom • urn:com.workday/bsvc/blob
Eingabe-ID	entryID	Gibt für jedes Dokument eine Eingabe-ID an. Muss in Ihrem Mandanten eindeutig sein. Wenn Sie keinen Wert für die Eingabe-ID angeben, generiert Workday eine ID. Wenn Sie ein bereits vorhandenes Dokument angeben, überschreibt Workday das Dokument, sofern es veränderbar ist. Sie können eine Eingabe-ID nicht wiederverwenden, wenn das vorhandene Dokument unveränderlich ist. Wenn Sie Dokumente über die REST-API aktualisieren, gilt Folgendes: • Verwenden Sie die HTTP-Methode PUT für Dokumente mit vordefinierten Eingabe-IDs. • Verwenden Sie die HTTP-Methode POST, wenn die Eingabe-ID nicht angegeben ist oder als „null“ oder als leere Zeichenfolge evaluiert wird.
Ist Anhang	isAttachment	Gibt an, ob Ihr Dokument ein OMS-Anhang ist.
Unveränderbar	immutable	Gibt an, ob Sie Ihr Dokument nach der Erstellung ändern können.
Eigentümer zuweisen	assignOwner	Gibt an, ob Workday Sie als Eigentümer Ihres Dokuments zuweist.
Beschreibung des Dokumentreferenzplans	documentReferenceDescription	Eine Beschreibung Ihres Dokuments. Wird in Ihrem Workday-Mandanten angezeigt.

Referenz: Eigenschaften des Schritts „Retrieve“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Gibt die eindeutige ID des Abrufschritts innerhalb der Baugruppe an.
Ausgabe	output	Gibt an, wohin der Abrufschritt die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext. Wenn Sie einen Dokumentabrufservice für einen Geschäftsprozess konfigurieren, verwendet Workday die Zeichenfolge, die Sie im Feld Benutzerdefinierte Inhaltsart eingeben, als MIME-Typ für das von diesem Service gespeicherte Dokument. Wenn Sie das Dokument mit einem retrieve-Schritt entweder direkt oder über einen Dokumenten-Iterator oder ein Hilfsobjekt für

Eigenschaft	XML-Attributname	Beschreibung
		den Dokumentenzugriff, markiert Workday das Dokument automatisch mit dem gespeicherten MIME-Typ, damit spätere Schritte es korrekt verarbeiten können. Wenn Sie keine Benutzerdefinierte Inhaltsart eingeben, verwendet Workday die UTF-8-Codierung. Um Fehler zu vermeiden, geben Sie immer die entsprechende Benutzerdefinierte Inhaltsart für Daten ein, die nicht UTF-8-codiert sind.
Einzug	collection	Der Name der Sammlung, in der Ihr Dokument gespeichert wird.
Eingabe	entry	Die ID des Dokuments in Ihrer Sammlung. Abrufbar in MediationContext unter Verwendung eines MVE-Ausdrucks. Beispiel: <code>@{parts[0].xpath('//blob:entry', 'blob urn:com.workday/bsvc/blob')}</code>
Kennwort	password	Die Kennwortzugangsdaten für den Zugriff auf Ihr Dokument.
Benutzername	username	Die Zugangsdaten des Benutzernamens für den Zugriff auf Ihr Dokument.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Abrufschritts, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Abrufschritt ausgeführt wird.

Referenz: Eigenschaften des Schritts „Xmldiff“

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Gibt die eindeutige ID des xmldiff -Schritts innerhalb der Build an.
Erste Eingabe	first-input	Gibt die erste Dateneingabequelle für den Schritt xmldiff an. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Zweite Eingabe	second-input	Gibt die zweite Dateneingabequelle für den Schritt xmldiff an. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht.

Eigenschaft	XML-Attributname	Beschreibung
		Geben Sie den Anhangindex an. <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, wohin der Schritt xmldiff die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch alle Anhänge entfernt werden. <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. <i>attach</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
In Ausgabe kopieren	copy-to-output	Gibt an, ob der xmldiff-Schritt die Ausgabe in die Vermittlungsmeldung oder in eine Variable kopiert. Wenn der Wert auf Wahr gesetzt ist, kopiert der Schritt xmldiff die Ausgabe in die Mediationsnachricht.

Eigenschaft	XML-Attributname	Beschreibung
		Wenn der Wert auf Falsch gesetzt ist, kopiert der Schritt xmlDiff die Ausgabe in eine Variable.
Attribute ignorieren	ignore-attributes	Gibt an, ob der xmlDiff- Schritt Attribute beim Vergleich von zwei XML-Dokumenten ignoriert.
Namespaces	namespaces	Namespace-Überschreibungsdefinitionen für XPath-Ausdrücke, die durch den Schritt xmlDiff implementiert werden.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Schritts xmlDiff, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt xmlDiff ausgeführt wird.
XPath-Version	xpath-version	Gibt an, ob XPath Version 1.0 oder 2.0 verwendet werden soll. Geben Sie den Wert 1 für Assembly-Versionen bis Workday 11 und 2 für Assembly-Versionen ab Workday 12 ein. Workday ignoriert diese Eigenschaft, wenn der Wert stream-xpath auf Wahr gesetzt ist. Sie können in Workday die Version dynamisch festlegen, indem Sie eine MVAL-Vorlage verwenden, die einen Systemeigenschaftswert einfügt. Beispiel: <code>@{props.get('the_version')}</code>

Register „Definitionen“

Eigenschaft	XML-Attributname	Beschreibung
Name	name	Gibt den Namen der Stelle an sub-diff Diese Domäne ist beim Suchen von Ergebnissen des Schritts xmldiff zu verwenden.
Löschungen ignorieren	ignore-deletions	Gibt an, ob der Schritt xmldiff das Löschen in der Ausgabe ignoriert.
Nur andere Knoten	result-document-different-nodes-only	Gibt an, welche Knoten im Ausgabedokument für den Schritt xmldiff angezeigt werden. Wenn eingestellt, true, werden im Ausgabedokument nur Knoten der obersten Ebene angezeigt. Wenn eingestellt, false, werden alle untergeordneten Knoten im Ausgabedokument angezeigt.
Einschluss von Ergebnisdokument	result-document-inclusion	Gibt an, ob der Schritt xmldiff Differenzen zwischen Knoten im Ausgabedokument aufzeichnet. Es gibt drei mögliche Werte: • <i>Always</i>: der ausgewählte Knoten und seine untergeordneten Knoten werden immer im Ergebnisdokument angezeigt. • <i>Different</i>: der ausgewählte Knoten und seine untergeordneten Knoten werden nur dann im Ergebnisdokument angezeigt, wenn Differenzen gefunden werden. • <i>Never</i>: der ausgewählte Knoten und seine untergeordneten Knoten werden nie im Ergebnisdokument angezeigt.
XPath	xpath	Der XPath-Ausdruck, der durch den Schritt xmldiff implementiert wird, um Knoten in Ihrem Eingabedokument zu suchen.

Register „Matcher XPath“

Auf dem Register Matcher XPath können Sie XPath-Ausdrücke hinzufügen, die nicht-sequenzielle übereinstimmende Knoten in Ihren Eingabedokumenten suchen.

Referenz: Eigenschaften des Schritts „Enqueue-Message“**Register „Allgemein“**

Eigenschaft	XML-Attributname	Beschreibung
Schritt-ID	id	Die eindeutige ID des Schritts zum Aktivieren der Meldung in der Mandantenwarteschlange.
Eingabe	input	Gibt an, wo der Schritt zum Aktivieren der Meldung aus der Warteschlange Daten als Eingabe abrufen. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. Nur anwendbar, wenn die Eingabe eine SOAP-Meldung enthält. • <i>attachment</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. • <i>variable</i>: Verwenden Sie eine benannte Variable aus dem MediationContext.
Ausgabe	output	Gibt an, an wen der Schritt zum Aktivieren der Meldung in der Warteschlange die Ausgabe weiterleitet. Es gibt 5 mögliche Werte: • <i>message</i>: die gesamte Nachricht, einschließlich rootpart und Anhänge. Dieser Wert erstellt eine neue Nachricht, wodurch

Eigenschaft	XML-Attributname	Beschreibung
		alle Anhänge entfernt werden. • <i>SOAPbody</i>: Der Inhalt des SOAP-Texts aus rootpart der Nachricht. • <i>attach</i>: ein MIME-Anhang in der Nachricht. Geben Sie den Anhangindex an. • <i>rootpart</i>: Der Inhalt von rootpart einer MIME-Nachricht. Wenn die Nachricht keine MIME-Datei ist, wählt Workday die gesamte Nachricht aus. Bei diesem Wert werden alle Anhänge der ursprünglichen Nachricht beibehalten. • <i>variable</i>: Speichern Sie die Ausgabe als benannte Variable in MediationContext.
Inhaltsart	contentType	Überschreibt die Inhaltsart Ihrer Ausgabemeldung, die standardmäßig text/xml ist.
Warteschlangenname	queueName	Der Name der Warteschlange, an die der Schritt "enqueue-message" die Ausgabe leitet.
Dateiname	fileName	Überschreibt den Dateinamen im Header der Vermittlungsmeldung.

Register „Erweitert“

Eigenschaft	XML-Attributname	Beschreibung
Beschreibung	description	Eine Beschreibung des Schritts zum Aktivieren der Meldung in der Warteschlange, die im konsolidierten Protokoll angezeigt wird. Mit diesem Feature können Sie die Funktion des Schritts und die Daten, für die er ausgeführt wird, beschreiben.
Ausführung bei	execute-when	Eine Bedingung, die bestimmt, ob der Schritt zum Aktivieren der Meldung ausgeführt wird.

Assembly-Module

Assembly-Module erstellen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Die Assembly-Module von Workday Studio bieten parametrisierte Vorlagen für die Erstellung von Assemblys. Damit können Sie vorkonfigurierte Gruppen von Komponenten und Schritten erstellen und wiederverwenden, die bestimmte Aufgaben ausführen.

Prozedur

1. Fügen Sie die Elemente hinzu, die Sie in Ihr Assembly-Modul aufnehmen möchten, und konfigurieren Sie deren Eigenschaften nach Bedarf. Sie können ein Modul aus einer funktional abgeschlossenen Assembly oder aus einer kleineren Menge von Komponenten und Schritten erstellen. Benutzer können Parameter anpassen, wenn sie das Modul implementieren.
2. Speichern Sie Ihre Assembly.
3. Klicken Sie im Project Explorer mit der rechten Maustaste auf das Assembly-Projekt und wählen Sie Create Assembly Module... aus.
4. Geben Sie einen Namen und eine Beschreibung für das Modul an. Die Beschreibung wird als QuickInfo für das Modul angezeigt.
5. Wählen Sie ein Symbol für das Modul aus. Sie können das Standardsymbol verwenden oder ein eigenes hochladen.
6. Studio organisiert Modulparameter in Sammlungen, die als Seiten bezeichnet werden. Ein Parameter muss zu einer Seite gehören. Jede Seite hat einen Namen und eine Beschreibung. Jeder Parameter hat einen Namen, ein Label (wird im Assistenten Modul konfigurieren angezeigt), eine Beschreibung, eine Art und einen Standardwert. Es gibt sechs Parameterarten:

Art	Beschreibung	Beispiel:
Zeichenfolge	Ein Textwert.	"HelloWorld"
XMLRequest	Ein XML-Literal für eine Anforderungsnachricht. Modulbenutzer geben einen Wert für die Webservice-Anforderung entweder über den Assistenten Create Literal XML oder Create Workday Web Service Request an.	<pre><?xml version="1.0" encoding="UTF-8"? > <env:Envelope xmlns:env="http:// schemas.xmlsoap.org/ soap/envelope/" xmlns:xsd="http:// www.w3.org/2001/ XMLSchema"> <env:Body> <wd:Get_Workers_Request xmlns:wd="urn:com.workday/ bsvc" wd:version="v23.2"> <wd:Request_References wd:Skip_Non_Existing_Instances="t <wd:Worker_Reference> <wd:ID wd:type="Employee_ID">abcdef</ wd:ID> </</pre>

Art	Beschreibung	Beispiel:
		<pre> wd:Worker_Reference> </ wd:Request_References> </ wd:Get_Workers_Request> </env:Body> </ env:Envelope> </pre>
Werteliste	Sie können Class Report Fields (CRFs) verwenden, um eine Werteliste für Startparameter von Workday-Objekten einer bestimmten Art bereitzustellen. Um eine CRF-Referenz anzugeben, klicken Sie auf Referenz-ID auswählen. Anmerkung: Um eine CRF-Referenz anzugeben, müssen Sie eine Verbindung zu einem Workday-Mandanten herstellen.	fbbe44a681251000039acc2ff657000a
Datei	Ein Dateiname. Sie können die Dateierweiterung als angeben data -Wert	*.xslt
Enum	Ein einzelner Literalwert oder mehrere Werte, getrennt durch ein Komma oder ein Leerzeichen.	code1 code2 code3
Boolesch	Ein boolescher Wert.	true

- Wenn Sie mit dem Hinzufügen von Parametern fertig sind und das Modul speichern, öffnet Studio die zugehörige XMI-Datei im XML-Editor. Die Datei enthält eine params Element „ für jeden Parameter, den Sie angegeben haben. Die assemblyText und diagramText -Elemente enthalten die in .gekapselten Build-XML- und Diagrammdetails <![CDATA[]]> Tags Sie können die XMI-Datei direkt bearbeiten.
- Schließen Sie die XMI-Datei.

Assembly-Module zu einem Projekt hinzufügen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt oder öffnen Sie ein vorhandenes Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Die Assembly-Module von Workday Studio bieten parametrisierte Vorlagen für die Erstellung von Assemblys. Damit können Sie vorkonfigurierte Gruppen von Komponenten und Schritten erstellen und wiederverwenden, die bestimmte Aufgaben ausführen.

Prozedur

1. Ziehen Sie die Assembly per Drag-and-Drop aus dem Abschnitt Module der Palette in den Assembly-Editor. Studio öffnet den Assistenten Module konfigurieren.

Anmerkung: In der Modulbibliothek werden nur die Assembly-Module angezeigt, die Sie auf der Systemeinstellungsseite Fenster > - Systemeinstellungen > Workday > - Assembly Editor > - Module ausgewählt haben.

2. Konfigurieren Sie die Parameter des Moduls.
3. Wenn das Modul Build einen Parameter des Typs enthält XMLRequest klicken Sie entweder auf Create Literal XML oder Create Workday Web Service Request , um einen Wert für die Webservice-Anforderung anzugeben. Bei der Bearbeitung dieser Aufgabe sind folgende Informationen zu beachten:

Assistent	Operations	Anmerkungen
Literal-XML erstellen	Get, Put	• Kann für jeden Webservice verwendet werden. • Wählen Sie den Namen und die Version des Webservices für die Anforderung aus den Listen unter dem Textbereich aus.
Workday-Webserviceanforderung erstellen	Get	• Verwenden Sie diese Option nur für Workday-Webservices. • Studio füllt den Namen und die Version des Webservices automatisch aus, wenn Sie einen Vorgang auswählen.

4. Studio zeigt eine Vorschau der im Modul enthaltenen Assembly-Elemente an. Wenn Sie den Assistenten abschließen, fügt Studio diese Elemente zur Assembly hinzu.

Konzept: Assembly-Module

Die Assembly-Module von Workday Studio bieten parametrisierte Vorlagen für die Erstellung von Assemblys. Damit können Sie vorkonfigurierte Gruppen von Komponenten und Schritten erstellen und wiederverwenden, die bestimmte Aufgaben ausführen. Sie können ein Modul aus einer gesamten Assembly oder aus einem Segment erstellen.

Eine Assembly, die Sie mit einem Modul erstellen, unterscheidet sich nicht von einer Assembly, die Sie von Grund auf neu erstellen. Sie sehen identisch aus und verhalten sich identisch.

Studio speichert ein Modul als Vorlage im Format XML Metadata Interchange (XMI) mit der Dateierweiterung xmi. Greifen Sie über die Modulbibliothek in der Studio-Palette auf Ihre gespeicherten Module zu. Damit andere Entwickler auf Ihre Module zugreifen können, speichern Sie sie an einem geteilten Speicherort.

Die gemeinsame Nutzung von bewährten Assembly-Modulen trägt dazu bei, die Integrationsentwicklung zu beschleunigen und die Kosten zu reduzieren.

In der Modulbibliothek werden nur die Assembly-Module angezeigt, die Sie auf der Systemeinstellungsseite Fenster > Systemeinstellungen > Workday > Assembly Editor > Module ausgewählt haben.

Subassemblys

Subassemblys erstellen

Vorbereitungen

Erstellen Sie eine Assembly.

Warum und wann dieser Vorgang ausgeführt wird

Eine Subassembly ist eine Verarbeitungskette, deren erste Komponente ein local-in-Transport ist. Sie rufen ihn mit einem local-out-Transport auf, entweder aus derselben oder aus einer separaten Assembly.

Prozedur

1. Fügen Sie den local-in-Transport hinzu, der den Anfang der Subassembly definiert, und konfigurieren Sie ihn.
 - a) Fügen Sie einen local-in-Transport am Anfang der Verarbeitungskette hinzu, der in eine Subassembly umgewandelt werden soll.
 - b) Konfigurieren Sie auf den Registern Allgemein und Erweitert der Ansicht Eigenschaften die Eigenschaften des Transports. Sie können ein Symbol für die Subassembly und Text angeben, der als ihre QuickInfo angezeigt wird.
 - c) Definieren Sie auf dem Register Parameter die Parameter, die für die Subassembly als Eingabe erforderlich sind. Sie können diese Parameter für den lokalen - ausgehenden Transport konfigurieren, der die Untergruppe aufruft. Studio speichert sie als MediationContext -Eigenschaften
 - d) Definieren Sie auf dem Register Abmeldeparameter die MediationContext Eigenschaften, die die Untergruppe festlegt.
 - e) Speichern Sie die Baugruppe.
2. Fügen Sie den local-out-Transport hinzu, der die Subassembly aufruft, und konfigurieren Sie ihn.
 - a) Fügen Sie der Assembly, die die Subassembly aufruft, einen local-out-Transport hinzu.
 - b) Konfigurieren Sie auf den Registern Allgemein und Erweitert in der Ansicht Eigenschaften die zugehörigen Eigenschaften. Geben Sie die Untergruppe - local-in als Endpunkt- Eigenschaft an. Verwenden Sie das -Format `vm://assembly-name/local-out-ID`, oder klicken Sie auf Untergruppe mit lokalem Eingang auswählen.
 - c) Legen Sie auf dem Register Parameter Folgendes fest: MediationContext -Eigenschaften Sie können neue Parameter erstellen, indem Sie auf "Satzelement erstellen" klicken. Sie können auch beliebige Parameter auswählen, die Sie im local-in-Transport definieren, und ihnen Werte zuweisen, indem Sie auf Select Parameter klicken.

Anmerkung: Im Assistenten Select Parameters zeigt ein Sternchen an, dass die Eigenschaft erforderlich ist. Ein Fragezeichen zeigt an, dass die Verwendung dynamisch ist.

 - d) Speichern Sie die Assembly.

Konzept: Assemblys

Eine Subassembly ist eine Verarbeitungskette, deren erste Komponente ein local-in-Transport ist. Local-in definiert einen Einstiegspunkt in die Verarbeitungskette, sodass sie von einem Local-out-Transport in derselben oder in einer anderen Assembly aufgerufen werden kann. Sie können mehrere Subassemblys und Local-out-Transporte in dieselbe Assembly einschließen.

Anmerkung: Sie können Subassemblys nicht allein für Workday bereitstellen, da sie nicht mit einem Workday-In-Transport beginnen.

Da Sie sie nach Belieben aufrufen können, können Sie mit Subassemblies die Verarbeitungslogik zur Wiederverwendung kapseln. Sie können sie sich als Programmier-Unterprogramme ansehen. Die Standard-ID für einen lokal eingehenden Transport lautet `SubRoutine`. Die Standard-ID für einen ausgehenden lokalen Transport lautet `CallSubRoutine`.

Untergruppen haben Parameter. Sie geben sie für den Transport lokal an und weisen ihnen Werte für den Transport lokal ausgehend zu. Diese Parameter definieren Eigenschaften in der `MediationContext`.

Wenn Sie eine Untergruppe speichern, fügt Workday Studio automatisch einen lokalen ausgehenden Transport, der die Untergruppe aufruft, zur Kategorie „Arbeitsbereichskomponenten“ auf der Palette hinzu. Dieser Local-out-Transport verwendet das benutzerdefinierte Symbol aus dem Local-in-Transport der Subassembly, falls Sie eines bereitgestellt haben. Andernfalls wird ein Fragezeichensymbol verwendet.

In der Palette werden immer öffentliche Subassemblies angezeigt. Private Subassemblies werden nur angezeigt, wenn sie in der Assembly vorkommen. Studio stellt in der Kategorie Gemeinsame Komponenten der Palette mehrere vorkonfigurierte Subassemblies bereit.

Um den Local-in-Transport zu finden, der von einem Local-out referenziert wird, klicken Sie mit der rechten Maustaste auf den Local-out und wählen Sie Deklaration in Arbeitsbereich.

Um die Local-out-Transporte zu finden, die von einem Local-in referenziert werden, klicken Sie mit der rechten Maustaste auf den Local-in und wählen Sie Referenzen in Arbeitsbereich.

relatedlinks

Konzepte

Konzept: [Allgemeine Komponenten](#) auf Seite 106

Benutzerdefinierte Assembly-Komponenten

Benutzerdefinierte Java-Klassen und Spring-Beans erstellen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Sie können vier benutzerdefinierte Assembly-Elemente aus der Palette in Workday Studio erstellen:

- Komponente custom-mediation
- Schritt custom
- Transport custom-out
- custom-error-handler

Ein fünftes benutzerdefiniertes Element, die benutzerdefinierte Routingstrategie, ist über die Kontextsymbolleiste der Komponente Route verfügbar.

Prozedur

1. Öffnen Sie eine Assembly im Assembly-Editor.
2. Fügen Sie der Assembly eines der folgenden benutzerdefinierten Elemente hinzu:
 - Komponente custom-mediation
 - Schritt custom
 - Transport custom-out
 - custom-error-handler
3. Konfigurieren Sie die Eigenschaften des benutzerdefinierten Elements in der Ansicht Eigenschaften.

4. Klicken Sie rechts neben dem Feld Spring-Bean auf Eigenschaften für Java-Klassen und Spring-Beans erstellen. Die von Studio angezeigten Eigenschaften variieren je nach Art des benutzerdefinierten Elements geringfügig. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
ID	Geben Sie eine eindeutige Spring-Bean-ID an.
Umfang	- Wählen Sie <i>singleton</i>, um bei jeder Referenz dieselbe Bean-Instanz zurückzugeben. - Wählen Sie einen <i>prototype</i> aus, um mit jeder Referenz eine andere Bean-Instanz zurückzugeben.
Paket	Geben Sie das Java-Paket an.
Name	Geben Sie den Klassennamen an.
Eingabeart	Wählen Sie die Eingabeart aus, die an die Java-Methode übergeben werden soll. Sie müssen keine Implementierungsdetails angeben, um auf den Nachrichtenteil zuzugreifen und ihn in die entsprechende Art zu konvertieren. Workday prüft den ersten Parameter der Methode und konvertiert den Nachrichtenteil in dieselbe Art.
Ausgabeart	Wählen Sie die Ausgabeart der Methode (return) aus.
Methodenname	Geben Sie den Namen der Java-Methode ein, die Sie innerhalb der Java-Klasse definieren möchten. Der Standardwert ist <code>process</code> .
Schließen Sie Argument für dynamischen Endpunkt ein	Gibt an, ob ein Argument für einen dynamischen Endpunkt eingeschlossen werden soll.
Anmerkung zur Java-Klasse für die Anzeige in der Palette hinzufügen	Aktivieren Sie dieses Kontrollkästchen, um das benutzerdefinierte Element in der Palette anzuzeigen. Mit dieser Option wird eine Anmerkung zur generierten Java-Klasse hinzugefügt, die sich im Ordner <code>src</code> Ihres Assembly-Projektordners befindet.
QuickInfo	Geben Sie den Text ein, der als Beschreibung des benutzerdefinierten Elements angezeigt wird.

5. Speichern Sie die Assembly.

Ergebnisse

Studio generiert die Java-Klasse, die das benutzerdefinierte Element innerhalb des angegebenen Pakets darstellt. Außerdem wird ein Verweis auf die neue benutzerdefinierte Spring-Bean am unteren Rand der Datei „Assembly.xml“ hinzugefügt.

Nächste Maßnahme

Fügen Sie Ihrer benutzerdefinierten Java-Klasse den erforderlichen Code hinzu.

Konzept: Benutzerdefinierte Assembly-Elemente

Sie können in Workday Studio fünf Arten benutzerdefinierter Assembly-Elemente erstellen:

Benutzerdefiniertes Element	Verfahren
Mediationskomponente	Fügen Sie eine Komponente custom-mediation aus der Kategorie Komponenten der Palette hinzu.
Mediationsschritt	Fügen Sie einen benutzerdefinierten Schritt aus der Kategorie Andere in der Palette hinzu.
Ausgehender Transport	Fügen Sie einen Transport custom-out aus der Kategorie Transporte (Ausgehend) der Palette hinzu.
Fehlerhandler	Fügen Sie eine Komponente custom-error-handler aus der Kategorie Fehlerhandler der Palette hinzu.
Routing-Strategie	Wählen Sie in der Kontext-Symbolleiste einer Routing-Komponente die Option Benutzerdefinierte Strategie erstellen.

In jedem Fall besteht das benutzerdefinierte Element aus drei Teilen:

- Die zugrunde liegende Java-Implementierung, die definiert, welche Klassen und Methoden das Element verwendet.
- Die Spring-Bean-Konfiguration, die Workday mitteilt, wie Instanzen der Java-Klassen erstellt werden sollen.
- Die Konfiguration der Assembly-Komponenten, die Ihr benutzerdefiniertes Element in der grafischen Assembly-Ansicht platziert.

Benutzerdefinierte Assembly-Elemente können bei der Verarbeitung eingehender Daten normal auf Nachrichtenteile zugreifen. Sie können auch den Nachrichtenteil angeben, an den Workday die Ausgabe des benutzerdefinierten Elements weiterleiten soll.

Um ein benutzerdefiniertes Element zu teilen, exportieren Sie das Java-Paket als binäre JAR-Datei. Andere Entwickler können die JAR dann zum Klassenpfad ihres Projekts hinzufügen.

Benutzerdefinierte Schritte

Beachten Sie beim Schreiben einer Java-Implementierung für einen benutzerdefinierten Schritt die folgenden Punkte:

- Die Methode muss angegeben werden `public` sodass sie von außerhalb der Klasse aufgerufen werden kann.
- Die Methode muss nicht statisch sein.
- Jede Methode muss einen eindeutigen Namen haben.
- Parameterlisten müssen mit den unterstützten Kombinationen übereinstimmen.
- Die Methode kann jede Art von Ausnahme auslösen.
- Wenn Sie keinen Methodennamen angeben, wählt Workday eine beliebige öffentliche Methode (ohne Berücksichtigung von Getter und Setter) mit einer Parameterliste aus, die mit den unterstützten Methodensignaturen übereinstimmt. Bei Unklarheiten meldet Workday während der Bereitstellung einen Fehler, den Sie durch Angabe eines Methodennamens beheben können.

Ein benutzerdefinierter Schritt muss eine Methode bereitstellen, die einer der folgenden Signaturen entspricht:

Unterschrift	Verwendung	Beispiel:
<code>public return type method(input parameter type input) [throws any Exception]</code>	Definiert die Rückgabeart der Methode und einen Eingabeparameter.	<code>public String addTimestamp(byte[] messageContent) throws IOException {</code>
<code>public void method(input parameter type input, output parameter type output) [throws any Exception]</code>	Definiert unterstützte Eingabe- und Ausgabeparameter.	<code>public void addTimestamp(byte[] messageContent, OutputStream out) throws IOException {</code>

Der benutzerdefinierte Schritt unterstützt die folgenden Arten von Eingabe-, Rückgabe- und Ausgabeparametern:

Eingabearten	Rückgabearten	Ausgabeparameterarten
• <code>java.lang.String</code> • <code>byte[]</code> • <code>java.io.InputStream</code> • <code>javax.activation.DataHandler</code> • <code>javax.activation.DataSource</code> • <code>org.w3c.dom.Element</code> • <code>org.w3c.dom.Document</code> • <code>javax.xml.transform.Source</code> • <code>javax.xml.transform.stream.StreamSource</code> • <code>javax.xml.transform.dom.DOMSource</code>	• <code>void</code> • <code>boolean</code> (Nurbenutzerdefinierter Schritt) • <code>java.lang.String</code> • <code>byte[]</code> • <code>java.io.InputStream</code> • <code>javax.activation.DataHandler</code> • <code>javax.activation.DataSource</code> • <code>org.w3c.dom.Element</code> • <code>org.w3c.dom.Document</code> • <code>javax.xml.transform.Source</code>	• <code>java.io.OutputStream</code> • <code>javax.xml.transform.Result</code>

Wenn ein benutzerdefinierter Schritt Folgendes zurückgibt: `void` Durch den Typ "" oder einen Wert "null" wird die Vermittlungsmeldung nicht geändert.

Konzept: Verpackung von benutzerdefinierten Elementen

Anmerkungen zur Java-Klasse

Wenn Sie ein benutzerdefiniertes Assembly-Element in verschiedenen Projekten wiederverwenden möchten, verwenden Sie Anmerkungen, damit das Element in der Palette angezeigt wird.

Aktivieren Sie dazu das Kontrollkästchen `Annotate Java Class`, damit sie in Palette angezeigt wird, wenn Sie die Java-Klasse und die Quellbean erstellen. Studio fügt eine hinzu `@Component` in der generierten Java-Klasse, die sich im Ordner `src` Ihres Build-Projekts befindet. Beispiel:

```
import com.capeclear.assembly.annotation.Component;
import com.capeclear.assembly.annotation.ComponentMethod;
import com.capeclear.assembly.annotation.Property;
import static com.capeclear.assembly.annotation.Component.Type.*;

@Component(
    name = "TimeStamperCustomMediationStepImpl",
    type = mediation,
    tooltip = "This custom code step adds the current time on the first
line of the message.",
    scope = "singleton",
```

```

        smallIconPath = "icons/TimeStamperCustomMediationStepImpl_16.png",
        largeIconPath = "icons/TimeStamperCustomMediationStepImpl_24.png"
    )

```

Sie können diese Informationen manuell bearbeiten.

Beachten Sie, dass Studio die Symbole für das benutzerdefinierte Element im Projekt speichert src/icons Ordner. Es gibt zwei Versionen, 16 x 16 und 24 x 24. Um benutzerdefinierte Symbole zu verwenden, ersetzen Sie die Dateien _16.png und _24.png durch Ihre eigenen Bilder.

Eigenschaften von benutzerdefinierten Assembly-Elementen offenlegen

Wenn Ihr benutzerdefinierter Java-Code Eigenschaften hat, die Benutzer festlegen sollen, können Sie diese auf dem Register Bean der Ansicht Eigenschaften des benutzerdefinierten Elements anzeigen.

Geben Sie dazu Get- und Setter-Methoden an, die dem Bean-Muster folgen. Beispiel: für Eigentum anbieten Getter- und Setter-Methoden wie z. B. getAbc und setAbc.

Fügen Sie dann die hinzu @Property Anmerkung zu einer der set oder die get Methode Die name Der Wert "" definiert das Label für die Eigenschaft in der Ansicht "Eigenschaften" und in der tooltip Der Wert "Wert" definiert den Text, den Studio anzeigt, wenn Sie den Cursor über das Label bewegen. Sie können die Reihenfolge, in der Eigenschaften aufgelistet werden, mit einem Wert für die Reihenfolge steuern. Kleinere Werte werden zuerst angezeigt. Ein editelement Der Wert gibt die Art der Schaltfläche an, die neben dem Eingabefeld für Eigenschaften angezeigt wird, wie folgt:

```

// Displays a text box with a Select button to enable you to select a Spring
// bean.
@property(name="Text", edit=Edit.BEAN_REF)

// Displays a text box with a Select button to enable you to select a Java
// class.
@property(name="Text", edit=Edit.CLASS)

// Displays a text box with a dialog box that enables you to specify a cron
// expression.
@property(name="Text", edit=Edit.CRON)

// Displays a text box without a button.
@property(name="Text", edit=Edit.DEFAULT)

// Displays a Browse button to enable you to specify a file.
@property(name="Text", edit=Edit.FILE)

// Displays a Browse button to enable you to specify a folder.
@property(name="Text", edit=Edit.FOLDER)

```

Studio sucht im Projektklassenpfad des Projekts, das die gerade bearbeitete Datei Montag.xml enthält, nach Anpassungsklassen. Um Ihre benutzerdefinierte Komponente an mehrere Assembly-Projekte zu verteilen, verpacken Sie die Anpassung in binärer Form als JAR-Datei, die die Klassendateien und Symbolbilder enthält. Andere Benutzer können dieses JAR im Klassenpfad des Projekts enthalten, damit es in ihrem Assembly-Editor verfügbar ist.

Der Assembly-Editor lädt beim Start automatisch alle benutzerdefinierten Komponenten, die er in seinem Klassenpfad findet. Klicken Sie mit der rechten Maustaste auf den Hintergrund des Assembly-Diagramms und wählen Sie Anpassungen aktualisieren aus, um alle zuletzt hinzugefügten benutzerdefinierten Komponenten neu zu laden.

Spring Bean-Konfiguration

Konzept: Spring Bean-Konfiguration

Sie können die Eigenschaften von Spring Beans auf dem Register Beans in der Ansicht Eigenschaften des Assembly-Editors bearbeiten.

Studio zeigt alle Beans der Assembly links in der Ansicht unter einem Beans-Knoten der obersten Ebene an. Die mit dem Beans-Knoten verknüpften Eigenschaften sind global und wirken sich auf alle Beans in der Assembly aus.

Über den Beans werden verschiedene Optionen angezeigt. Ihre Verfügbarkeit hängt von Ihrer Auswahl in der Beans-Liste ab. Es gibt folgende Optionen:

Option	Beschreibung
„Bean“ erstellen	Workday erstellt ein Spring Bean-Element in der Assembly-Datei. Wählen Sie den Beans-Knoten der obersten Ebene, um diese Option verfügbar zu machen.
„Alias“ erstellen	Workday erstellt einen Alias für ein Spring-Bean-Element in der Assembly-Datei. Wählen Sie den Beans-Knoten der obersten Ebene, um diese Option verfügbar zu machen.
„Eigenschaft“ erstellen	Workday erstellt ein Eigenschaftselement für die ausgewählte Bean.
"constructor-arg" erstellen	Erstellt eine constructor - arg -Element für die ausgewählte Bean.
"Ref" erstellen	Erstellt eine ref -Element, sodass andere Beans die ausgewählte Eigenschaft referenzieren können.
"Liste" erstellen	Erstellt eine list -Element für die ausgewählte Eigenschaft, mit dem Sie die Eigenschaften einer Listen-Java-Sammlung definieren können.
"Karte" erstellen	Erstellt eine map -Element für die ausgewählte Eigenschaft, mit dem Sie die Eigenschaften einer Map Java-Sammlung definieren können.
"Import" erstellen	Erstellt eine oberste Ebene import -Element in der Bean-Definition der Zusammensetzung, sodass Sie eine Ressourcendatei angeben können, die eine oder mehrere Bean-Definitionen enthält. Wählen Sie die Beans der obersten Ebene aus, um diese Option verfügbar zu machen.
"Wert" erstellen	Erstellt eine verschachtelte value Element innerhalb der ausgewählten Eigenschaftsdefinition, sodass Sie die Eigenschaft als lesbare Zeichenfolge angeben können.

Referenz: Globale Bean-Eigenschaften

Eigenschaft	XML-Attributname	Beschreibung
Standardvorgabe für automatische Verkettung	default-autowire	Prüft die BeanFactory , um Beans automatisch aufzulösen.
Standard für Kandidaten für automatische Verknüpfung	default-autowire-candidate	Ermöglicht es Ihnen, die automatische Wiring auf einen Satz von Bean-Kandidaten zu beschränken, basierend auf dem Musterabgleich mit Bean-Namen.
Standardvorgabe für Abhängigkeitsprüfung	default-dependency-check	Gibt an, ob Studio nach nicht aufgelösten Bean-Abhängigkeiten sucht.
Standardvernichtungsmethode	default-destroy-method	Überprüft jede Bean für die angegebene Callback-Methode zum Löschen.
Standardinitialisierungsmethode	default-init-method	Eine Callback-Methode zur Initialisierung für jede Bean.
Standard-Lazy Init	default-lazy-init	Gibt an, ob Bean-Instanzen beim Start oder bei der ersten Anforderung erstellt werden. Wenn Sie nicht möchten, dass eine Singleton-Bean bei der Verwendung von vorinstanziiert wird ApplicationContext, auf „true“setzen.
Standardzusammenführung	default-merge	Bestimmt, ob Studio standardmäßig Sammlungen zusammenführt.

Referenz: Individuelle Bean-Eigenschaften

Eigenschaft	XML-Attributname	Beschreibung
Id	id	Die eindeutige ID des benutzerdefinierten Bean-Elements innerhalb der Composite.
abstrakt	abstract	Gibt an, ob die Bean nur als übergeordnete Bean-Definition verwendet werden kann, die als Vorlage für untergeordnete Definitionen dient. Der Standardwert ist false.
Autowire	autowire	Prüft die BeanFactory , um Beans automatisch aufzulösen.
Kandidaten automatisch verknüpfen	autowire-candidate	Ermöglicht es Ihnen, die automatische Wiring auf einen

Eigenschaft	XML-Attributname	Beschreibung
		Satz von Bean-Kandidaten zu beschränken, basierend auf dem Musterabgleich mit Bean-Namen.
Klasse	class	Name der Implementierungsklasse.
Abhängigkeitsprüfung	dependency-check	Gibt an, ob Studio nach ungelösten Bean-Abhängigkeiten sucht.
Abhängig von	depends-on	Erzwingt die Initialisierung einer oder mehrerer Beans, bevor die Bean, die dieses Element verwendet, initialisiert wird. Um die Abhängigkeit von mehreren Beans auszudrücken, listen Sie deren Namen durch Kommas, Leerzeichen oder Semikolons getrennt auf.
Vernichtungsmethode	destroy-method	Überprüft die Bean für die angegebene Callback-Methode zum Löschen.
Bean	factory-bean	Die Bean im aktuellen oder übergeordneten Container, der die Instanzmethode enthält, durch deren Aufruf das Objekt erstellt wird.
Herstellermethode	factory-method	Der Name der Herstellermethode, die bei der Instanziierung der Bean verwendet werden soll.
Init-Methode	init-method	Eine generische Initialisierungsmethode für die Bean.
Lasche Init	lazy-init	Gibt an, ob die Bean-Instanz beim Start oder bei der ersten Anforderung erstellt werden soll. Wenn Sie nicht möchten, dass eine Singleton-Bean bei der Verwendung von vorinstanziiert wird ApplicationContext, auf „true“setzen. Standardmäßig übernimmt diese Eigenschaft den Wert aus der Einstellung auf Containerebene.
Name	name	Einen Namen für die Bean.
Übergeordnet	parent	Übergeordnete Bean. Wenn angegeben, erbt die aktuelle Bean ihre Konfigurationsdaten.

Eigenschaft	XML-Attributname	Beschreibung
Primär	primary	Legt die aktuelle Bean-Definition als primären Kandidaten für die automatische Verknüpfung fest. Der Standardwert ist false.
Umfang	scope	Der Bereich der Objekte, die aus der Bean-Definition erstellt wurden. Der Standardwert ist singleton.

Referenz: Alias-Eigenschaften

Eigenschaft	XML-Attributname	Beschreibung
Alias	alias	Der Aliasname.
Name	name	Die ID der ursprünglichen Bean.

Referenz: Eigenschafts-Eigenschaften

Eigenschaft	XML-Attributname	Beschreibung
Name	name	Name der Eigenschaft
Quell Bean	spring-bean	Die ID einer anderen Bean im Container.
Wert	value	Eigenschaftswert.

Referenz: Constructor-Arg-Eigenschaften

Eigenschaft	XML-Attributname	Beschreibung
Index	index	Ein 0-basierter Index für das Konstruktargument.
Quell Bean	spring-bean	Die ID einer anderen Bean im Container. Die in der Bean-Definition angegebenen Konstruktargumente werden verwendet, um als Argumente an den Konstruktor der referenzierten Bean übergeben zu werden.
Art	type	Die Art des Konstruktargumentes.
Wert	value	Der Wert des Konstruktargumentes.

Referenz: Referenzeigenschaften

Eigenschaft	XML-Attributname	Beschreibung
Bean	bean	Referenziert eine Zielbean durch Angabe eines der Werte in dessen Name oder Id.
Local	local	Referenziert eine lokale Ziel-Bean durch Angabe ihres Id-Attributs. Ermöglicht dem XML-Parser die Validierung von XML-ID-Referenzen innerhalb derselben Datei.
Übergeordnet	parent	Referenziert eine Ziel-Bean in einem übergeordneten Container des aktuellen Containers. Gibt einen der Werte in ihrem Namen oder ihrer ID an.

Referenz: Listeneigenschaften

Eigenschaft	XML-Attributname	Beschreibung
Zusammenführen	merge	Gibt an, ob Java-Listensammlungen zusammengeführt werden. Standardmäßig gilt merge. Die Eigenschaft "" übernimmt ihren Wert von der Containerebene default-merge Einstellung „“.
Wertart	value-type	Die Java-Art für value Elemente in der Liste.

Referenz: Map-Eigenschaften

Eigenschaft	XML-Attributname	Beschreibung
Zusammenführen	merge	Gibt an, ob Java-Kartensammlungen zusammengeführt werden. Standardmäßig gilt merge. Die Eigenschaft "" übernimmt ihren Wert von der Containerebene default-merge Einstellung „“.
Wertart	value-type	Die Java-Art für value Elemente in der Liste.
Schlüsselart	key-type	Die Java-Art für key Elemente in der Karte.

Message Builder

XML-Literale basierend auf Schema oder WSDLs erstellen

Vorbereitungen

Erstellen Sie eine Assembly, die eines der folgenden Elemente enthält:

- Ein write-Schritt
- Ein log-Schritt
- Eine log-error-Fehlerbehandlung

Warum und wann dieser Vorgang ausgeführt wird

Mit dem Literal Builder von Workday Studio können Sie einfache XML-Dokumente erstellen, die dem ausgewählten Schema entsprechen, oder SOAP-Anforderungsnachrichten, die den ausgewählten WSDLs entsprechen. Sie können WS-Security-Header an SOAP-Anforderungsnachrichten anhängen.

Prozedur

1. Fügen Sie einer der folgenden Assembly-Elemente ein Textelement hinzu:
 - Ein write-Schritt
 - Ein log-Schritt
 - Eine log-error-Fehlerbehandlung
2. Klicken Sie im rechten Bereich auf XML-Literal generieren.
3. Wählen Sie im Assistenten für den Literal Builder das Schema oder WSDL-Element aus, für das Sie das XML-Literal generieren möchten. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Nachrichten anzeigen	Standardmäßig zeigt Studio WSDL-Elemente als Meldungen an. Um sie als Schnittstellendefinitionen mit anzuzeigen portTypes, Vorgänge, Eingabemeldungen und Ausgabemeldungen: Deaktivieren Sie diese Option.
SOAP einschließen	Schließt die ausgewählte XML-Definition in einen SOAP-Umschlag ein.
WS-Security-Header hinzufügen	Fügt eine SOAP hinzu Header Element "" zur SOAP-Anforderung. Der SOAP-Header enthält die WS-Sicherheitszugangsdaten, die Sie in bereitstellen UsernameToken Element Nur verfügbar, wenn Sie Wrap SOAPauswählen.
XSLT einschließen	Erstellt ein XML-Literal, das in ein XSL-Stylesheet eingeschlossen ist. Sie können das Stylesheet zur späteren Verwendung als XSLT-Datei speichern.
Auswahl abfragen	Sie werden aufgefordert, beim Erstellen und Kopieren eines XML- oder SOAP-Literals eine bestimmte Auswahl aus einem Schema zu treffen. Wenn Sie diese Option nicht

Option	Bezeichnung
	auswählen, verwendet Studio die erste Auswahl in einem bestimmten Set.

Ergebnisse

Studio zeigt eine Vorschau des XML-Literals an, das Sie bearbeiten können.

XPath-Ausdrücke basierend auf Schema oder WSDLs erstellen

Vorbereitungen

Erstellen Sie eine Assembly, die einen der folgenden Schritte enthält:

- java-to-xml
- validate
- xml-to-java

Warum und wann dieser Vorgang ausgeführt wird

In Workday Studio können Sie mit dem XPath Expression Builder XPath-Ausdrücke basierend auf ausgewählten Schemas oder WSDLs erstellen.

Prozedur

1. Klicken Sie in der Ansicht Eigenschaften für einen dieser Schritte auf Build XPath Expression, neben der Eigenschaft XPath:
 - java-to-xml
 - validate
 - xml-to-java
2. Wählen Sie im unteren Bereich des Assistenten XPath Expression Builder das Schema oder WSDL-Element aus, mit dem der XPath-Ausdruck übereinstimmen soll. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Nachrichten anzeigen	Standardmäßig zeigt Studio WSDL-Elemente als Meldungen an. Um sie als Schnittstellendefinitionen mit anzuzeigen portTypes, Vorgänge, Eingabemeldungen und Ausgabemeldungen: Deaktivieren Sie diese Option.
Präfix „SOAP“.	Standardmäßig stellt Studio den von Ihnen erstellten XPath-Ausdrücken folgenden Ausdruck voran, wie im oberen Textbereich gezeigt: env:Envelope/env:Body. Wenn Sie das Präfix entfernen möchten, deaktivieren Sie diese Option.

3. Bei Bedarf können Sie den XPath-Ausdruck im oberen Textbereich manuell bearbeiten. Beispiel: Sie möchten ein Vergleichselement verwenden, um das Nodeset zu filtern, das vom XPath zurückgegeben wird.

Konzept: Workday Studio Message Builder

Der Message Builder von Workday Studio bietet eine grafische Ansicht, die die Definition des Nachrichteninhalts für die folgenden Assembly-Elemente vereinfacht:

- Für den Schritt write
- Für den Schritt log
- Für die Fehlerbehandlung log-error

Das Register Message Builder im Assembly-Editor bietet einen zentralen Ort, an dem Sie Nachrichteninhalte für jedes relevante Element der Assembly definieren können. Der Message Builder ist auch als Register in der Ansicht Eigenschaften einzelner write, log, und log-error-Elemente verfügbar.

Im linken Bereich Message Builder können Sie den Inhalt der Nachricht definieren. Verwenden Sie die oberen Optionen, um verschiedene Arten von Nachrichtenelementen hinzuzufügen:

Option	Beschreibung
„message-content“ erstellen	Fügt den Inhalt der Mediationsnachricht zur Nachricht hinzu.
„text“ erstellen	Fügt der Nachricht ein Zeichenfolgeliteral hinzu. Es stehen drei Optionen zur Verfügung: • Geben Sie den Text manuell ein und klicken Sie auf Änderungen übernehmen. • Klicken Sie auf MVEL-Ausdruck einfügen, geben Sie MVEL-Ausdrücke ein (falls erforderlich in Kombination mit Nur-Text) und klicken Sie dann auf „Anwenden“. • Klicken Sie auf XML-Literal generieren und wählen Sie das definierende Schema oder WSDL-Element aus.
„exception“ erstellen	Fügt der Nachricht eine Ausnahme hinzu, falls im Mediationskontext vorhanden. Nur zur Verwendung in der Fehlerbehandlung log-error.
„line-separator“ erstellen	Fügt der Nachricht ein Zeilenumbruchzeichen hinzu.
„xpath“ erstellen	Fügt der Nachricht das Ergebnis eines XPath-Ausdrucks hinzu.
„xslt“ erstellen	Fügt der Nachricht das Ergebnis einer XSL-Transformation hinzu.
„xml-stream“ erstellen	Erstellt den Inhalt für den Schritt log oder write.
„regex-match“ erstellen	Wendet einen regulären Ausdruck auf den Nachrichteninhalt an und fügt alle übereinstimmenden Vorkommen zur Mediationsnachricht hinzu.
„static-file“ erstellen	Gibt eine Eingabedatei an, die die Nachricht mit ihrem gesamten Inhalt ausfüllt.
„message-store-id“ erstellen	Fügt die ID des Nachrichtenspeichers der Nachricht hinzu. Gilt nur, wenn Sie Folgendes festlegen: store-message Eigenschaft für den Transport.

Option	Beschreibung
"Header" erstellen	Fügt der Nachricht den Wert eines benannten Nachrichten-Headers hinzu.
„Headers“ erstellen	Fügt vollständige Header-Sets in Form von Name-Wert-Header-Paaren hinzu, die durch Zeilenumbrüche getrennt sind.
"customer-id" erstellen	Fügt die Kunden-ID aus der hinzu MediationContext zur Meldung. In der Meldung wird „null“ angezeigt, wenn die Kunden-ID nicht festgelegt wurde.

Im rechten Bereich definieren Sie den Wert des Nachrichtenelements. Der Inhalt des Wertebereichs hängt von der Art des Nachrichtenelements ab.

Verwenden Sie die Schaltflächen auf der rechten Seite des Inhaltsbereichs, um Nachrichtenelemente neu anzuordnen oder zu löschen. Ihre Reihenfolge im Message Builder wird in der XML-Datei gespiegelt.

Referenz: Eigenschaften von Nachrichtenelementen

Verwenden Sie die Schaltflächen oben im Message Builder, um Nachrichtenelemente zu den Elementen Schreiben, Protokollieren und Protokollfehler hinzuzufügen.

Nicht jedes Nachrichtenelement hat Eigenschaften.

message-content

Eigenschaft	XML-Attributname	Beschreibung
Grenzwert	limit	Beschränkung für die Anzahl der Zeichen, die die Nachricht enthalten darf. Der Standardwert Null gibt an, dass keine Zeichenbeschränkung vorhanden ist.

exception

Eigenschaft	XML-Attributname	Beschreibung
Escape-XML	escape - xml	Ermöglicht das Escape von XML bei Ausnahmen im Schreibschritt .

xpath

Eigenschaft	XML-Attributname	Beschreibung
Namespaces	namespaces	Ermöglicht die Angabe eines beliebigen für den XPath-Ausdrucks erforderlichen Namespace im Format <prefix1> <namespace1> <prefix2> <namespace2>.

Eigenschaft	XML-Attributname	Beschreibung
XPath	xpath	Der XPath-Ausdruck, mit dem ein Wert aus der eingehenden Meldung extrahiert wird.
XPath-Version	xpath-version	Workday Studio unterstützt die XPath-Versionen 1.0 und 2.0. Der Standardwert ist 2 für Assembly-Versionen ab Workday 12 und 1 für frühere Versionen.

xslt

Eigenschaft	XML-Attributname	Beschreibung
Generierung von Saxon-Bytecode zulassen		Gibt an, ob die Generierung von sächsischem Bytecode zulässig ist. Der Standardwert ist <code>false</code> .
Wiederverwendungsstrategie		Gibt die Wiederverwendungsstrategie für XSLT-Transformatoren an: • Standard: Wenn Sie den angeben <code>url</code> - Eigenschaft unter Verwendung des URL-Protokolls <code>mctx</code> oder eines MVE-Ausdrucks hinzufügen, lautet das Standardverhalten <code>no-reuse</code>. Andernfalls <code>transformer-pool</code>. • Anmerkung: Wenn Sie „Transformator-Factory“ auf „Sachsen“ setzen, ist das Standardverhalten <code>templates</code>. • <code>no-reuse</code>: Workday verwendet Transformationsobjekte nicht. • <code>templates</code>: Workday erstellt ein Vorlagen-Objekt, aus dem neue Transformer-Objekte erstellt werden. Diese Transformer werden nicht wiederverwendet. • <code>transformer-pool</code>: Workday verwendet einen Pool von Transformatoren. Nicht empfohlen, wenn das Transformer-

Eigenschaft	XML-Attributname	Beschreibung
		Objekt Werte im Cache zwischenspeichert.
Sichere Verarbeitung		Gibt an, ob ein Transformer nur die XSLT-Standardverarbeitung ausführt. Auf Wahr setzen, um Grenzwerte für XML-Konstrukte festzulegen. Beispiel: Ignorieren Sie die Verarbeitung von JavaScript- oder Java-Erweiterungen. Auf Falsch setzen, um XML gemäß den XML-Spezifikationen zu verarbeiten.
Trafotransformatorwerkstatt		Der Klassenname von TransformerFactory, die Sie verwenden möchten. Der Standardwert ist die werkseitige JAXP-Version. Es gibt noch drei weitere Optionen: • <code>net.sf.saxon.TransformerFactory</code> für Sachsen. • <code>org.apache.xalan.processor.Tr</code> für Xalan • <code>net.sf.joost.trax.TransformerF</code> für Job
Url		Speicherort der XSLT-Datei, entweder ein relativer Pfad oder eine absolute URL. Unterstützt MVEL-Ausdrucksvorlagen. Sie können XSLT mithilfe des URL-Protokolls mctx (Medienkontext) direkt aus dem Nachrichteninhalte laden. Beispiel: Um eine XSLT-Datei in einem Anhang zu laden, verwenden Sie <code>mctx:parts/1</code> .

xml-stream

Eigenschaft	XML-Attributname	Beschreibung
Namespaces	namespaces	Ermöglicht die Angabe eines beliebigen für den XPath-Ausdrucks erforderlichen Namespace im Format <code><prefix1> <namespace1> <prefix2> <namespace2></code> .
XPath	xpath	Der XPath-Ausdruck, der beim Abrufen von XML-Elementen auf die Meldung angewendet werden soll. Workday verwendet

Eigenschaft	XML-Attributname	Beschreibung
		die Streaming-API für die XML-Verarbeitung (StAX). Diese Bibliothek liest XML-Nachrichten und übersetzt sie in Ereignisse statt in DOM. Sie führt selbst keine Transformationen durch. Der XPath-Ausdruck ist dieselbe vereinfachte XPath-Teilmenge, die Sie in verwenden xml-stream-splitter Strategie. Workday unterstützt qualifizierte oder lokale Elementnamen und Sternchen. Der XPath-Standard wird nicht vollständig unterstützt. Beispiele: • /order/item, /ns:order/ns:order • //item, //ns:item, /ns:order/item • /*/*, /* /*/*
XPath-Version	xpath-version	Workday Studio unterstützt die XPath-Versionen 1.0 und 2.0. Der Standardwert ist 2 für Assembly-Versionen ab Workday 12 und 1 für frühere Versionen.
Grenzwert	limit	Schränkt die Anzahl der zu vergleichenden XML-Fragmente basierend auf dem XPath-Ausdruck ein. Hauptsächlich aus Performance-Gründen nützlich. Wenn Sie wissen, dass es nur ein übereinstimmendes XML-Fragment gibt, legen Sie den Grenzwert auf fest 1, um ein Pull-Parsing des Dokuments zu verhindern, nachdem das Ergebnis abgerufen wurde. Der Standardwert ist -1, was eine unbegrenzte Ergebnismenge bedeutet.

Regex-Übereinstimmung

Eigenschaft	XML-Attributname	Beschreibung
Trennzeichen	delimiter	Gibt eine Zeichenfolge an, die zum Trennen von zwei oder mehr Vorkommen verwendet

Eigenschaft	XML-Attributname	Beschreibung
		wird. Der Standardwert ist ein einzelnes Leerzeichen.
Grenzwert	limit	Gibt die Anzahl der Vorkommen an, die in den Ergebnissatz aufgenommen werden sollen. Der Standardwert ist Null, was alle Vorkommen bedeutet.
Regex	regex	Definiert den Abgleichausdruck.

static-file

Eigenschaft	XML-Attributname	Beschreibung
Zeichencodierung	character - encoding	Gibt die Zeichencodierung an. Der Standardwert ist UTF - 8.
Eingabedatei	input - file	Gibt die Datei an.

header

Eigenschaft	XML-Attributname	Beschreibung
Namen einschließen	include - name	Bestimmt, ob Studio den Header-Namen einfügt.
Name	name	Der Name des Headers.
Wert nicht festgelegt	not - set - value	Eine Zeichenfolge, die ausgegeben wird, wenn der Header nicht gefunden wird oder leer ist.

Transportbindungen

Konzept: Transportbindungen

Transportbindungen sind optionale untergeordnete Elemente, die Sie für Ausgangstransporte in Studio konfigurieren können. Transportbindungen ermöglichen Ausgangstransporte zur Unterstützung von Webservices in einem bestimmten Format.

Beispiel: Sie können ein untergeordnetes Element mit rest-binding für einen http-out-Transport konfigurieren, wenn Sie einen RESTful-Webservice aufrufen möchten.

Sie können Transportverbindungen für die folgenden Ausgangstransporte konfigurieren:

- http-out
- ftp-out
- ftps-out
- sftp-out
- xmpp-out
- custom-out
- email-out

Studio unterstützt folgende Transportbindungen:

Transportbindung	Beschreibung	Anmerkungen
rest-binding	Ermöglicht den Aufruf von RESTful-Webservices mithilfe von HTTP-Anforderungsmethoden.	Unterstützt folgende HTTP-Methoden: • GET • PUT • POST
wsdl-http-binding	Ermöglicht den Aufruf von Webservices mithilfe von HTTP-Anforderungsmethoden und einer WSDL-Datei.	Unterstützt folgende HTTP-Methoden: • GET • PUT • POST
wsdl-soap-binding	Ermöglicht den Aufruf von SOAP-basierten Webservices mithilfe von HTTP-Anforderungsmethoden.	Unterstützt folgende HTTP-Methoden: • GET • PUT • POST Unterstützt die folgenden untergeordneten Elemente • custom-policy • ws-rm • ws-security • ws-ut

Referenz: Eigenschaften für WSDL-HTTP-Bindung

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Endpunktname	endpoint-name	Der Endpunkt, von dem aus das Element wsdl-http-binding auf einen Webservice zugreift.
Name der Schnittstelle	interface-name	Die portType Domänenname (oder Schnittstelle) für den Webservice.
Servicename	service-name	Der Name des Webservice.
WSDL-Datei	wsdl-name	Die WSDL-Datei, die den Webservice beschreibt.
Parameter serialisieren	serializeParams	Wird aktuell nicht verwendet.

Referenz: Eigenschaften für WSDL-SOAP-Bindung

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Endpunktname	endpoint-name	Der Endpunkt, von dem aus das Element wsdl-soap-verbindlich auf einen Webservice zugreift.
Name der Schnittstelle	interface-name	Die portType Domänenname (oder Schnittstelle) für den Webservice.
Servicename	service-name	Der Name des Webservices.
WSDL-Datei	wsdl-name	Die WSDL-Datei, die den Webservice beschreibt.
Fehlerbehandlung	fault-handling	Gibt den Fehlerbehandlungsprozess des Elements „wsdl-soap-verbindlich“ für HTTP 500-Antwortcodes an. Folgende Optionen sind verfügbar: • ignore • error • abort

Untergeordnete Elemente mit WSDL-SOAP-Bindung

Konzept: Untergeordnete Elemente mit WSDL-SOAP-Bindung

Die Transportbindung wsdl-soap-Bindung unterstützt folgende untergeordnete Elemente:

Untergeordnetes Element	Beschreibung
custom-policy	Unterstützt eine benutzerdefinierte Webservice-Sicherheitsrichtlinie für einen Ausgangstransport.
ws-rm	Unterstützt eine Messaging-Sicherheitsrichtlinie für einen Ausgangstransport.
ws-security	Implementiert eine lokale Datei mit der Webservice-Sicherheitsrichtlinie für SOAP-Anfragen auf einem Ausgangstransport.
ws-ut	Ermöglicht dem Workday-Integrationsserver, die Sicherheitszugangsdaten wie Benutzernamen und Kennwort für eine Assembly auf einem Ausgangstransport zu extrahieren, zu validieren und weiterzugeben.

Referenz: Eigenschaften für benutzerdefinierte Richtlinie

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Methode	method	Die Methode, die in Ihrer benutzerdefinierten Sicherheitsrichtlinie aufgerufen wurde.
Bohne	ref	Die benutzerdefinierte spring bean, die Ihre benutzerdefinierte Sicherheitsrichtlinie implementiert.

Referenz: WS-Rm-Eigenschaften

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Datei	file	Der Pfad zu der Datei, die die Konfiguration Ihrer Sicherheitsrichtlinie für Messaging beschreibt.

Referenz: WS-Sicherheitseigenschaften

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Datei	file	Der Pfad zu der Datei, die Ihre Sicherheitskonfiguration für den Webservice beschreibt.

Referenz: WS-Ut-Eigenschaften

Register „Allgemein“

Eigenschaft	XML-Attributname	Beschreibung
Benutzername	username	Fügt einen Benutzernamen in den Sicherheits-Header ausgehender SOAP-Nachrichten in ein MediationContext.
Kennwort	password	Fügt ein Kennwort in den Sicherheits-Header ausgehender SOAP-Nachrichten in ein MediationContext.

Eigenschaft	XML-Attributname	Beschreibung
Kennwort-Digest-Algorithmus	password-digest-algorithm	Wendet einen Hash-Algorithmus auf Kennwörter im Sicherheits-Header ausgehender SOAP-Nachrichten in an MediationContext. Folgende Optionen sind verfügbar: • <i>SHA-1</i> • <i>SHA-256</i> • <i>SHA-384</i> • <i>SHA-512</i> • <i>none</i>
Quell Bean	ref	Eine optionale Quell-Bean-Domäne, die die Werte für Benutzername und Kennwort im Sicherheits-Header ausgehender SOAP-Nachrichten in authentifiziert MediationContext.
Erstellt	created	Erstellt ein Zeitstempелеlement im Sicherheits-Header ausgehender SOAP-Nachrichten in MediationContext.
Ohne Ausnahme	nonce	Erstellt ein Nonce-Element im Sicherheits-Header ausgehender SOAP-Nachrichten in MediationContext.

Schema Explorer

WSDL- und XSD-Dateien anzeigen

Warum und wann dieser Vorgang ausgeführt wird

In der Ansicht Schema Explorer werden WSDL- und XSD-Dateien im schreibgeschützten Format angezeigt. Sie können Inhalt aus diesen Dateien in Ihre Zwischenablage kopieren, sodass Sie die Daten in anderen Texteditor-Ansichten in Studio wiederverwenden können.

Prozedur

1. Wählen Sie in Fenster Fenster > "Schema-Explorer > anzeigen" aus.
2. Klicken Sie in der Ansicht Schema-Explorer auf WSDL oder XSD hinzufügen.
3. Wählen Sie eine WSDL- oder XSD-Datei aus.
4. Blenden Sie in der Ansicht Schema Explorer die WSDL- oder XSD-Datei ein, um ihre Struktur anzuzeigen.

Ergebnisse

Studio zeigt die WSDL- oder XSD-Datei in der Ansicht Schema Explorer an.

Nächste Maßnahme

Kopieren Sie den Inhalt aus der WSDL- oder XSD-Datei in Ihre Zwischenablage.

Workday Webservice-WSDLs anzeigen

Warum und wann dieser Vorgang ausgeführt wird

In der Ansicht Schema Explorer werden WSDL- und XSD-Dateien im schreibgeschützten Format angezeigt. Sie können Inhalt aus diesen Dateien in Ihre Zwischenablage kopieren, sodass Sie die Daten in anderen Texteditor-Ansichten in Studio wiederverwenden können.

Prozedur

1. Wählen Sie in Fenster Fenster > "Schema-Explorer > anzeigen" aus.
2. Klicken Sie in der Ansicht Schema-Explorer auf Workday-Webservices hinzufügen.
3. Wählen Sie einen Workday Webservice und blenden Sie ihn ein.
4. Wählen Sie wsdl.
5. Blenden Sie in der Ansicht Schema Explorer die Workday Webservice-WSDL ein, um ihre Struktur anzuzeigen.

Ergebnisse

Studio zeigt die Workday Webservice-WSDL in der Ansicht Schema Explorer an.

Nächste Maßnahme

Kopieren Sie Inhalte aus der Workday Webservice-WSDL in Ihre Zwischenablage.

Benutzerdefinierte RaaS-Berichtsschemas anzeigen

Warum und wann dieser Vorgang ausgeführt wird

In der Ansicht Schema Explorer werden WSDL- und Schemadateien (XSD) im schreibgeschützten Format angezeigt. Sie können Inhalt aus diesen Dateien in Ihre Zwischenablage kopieren, sodass Sie die Daten in anderen Texteditor-Ansichten in Studio wiederverwenden können.

Prozedur

1. Wählen Sie in Fenster Fenster > "Schema-Explorer > anzeigen" aus.
2. Klicken Sie in der Ansicht „Schema-Explorer“ auf Benutzerdefiniertes Raas-Berichtsschema hinzufügen.
3. Im Fenster Benutzerdefinierten RaaS-Bericht hinzufügen ist Folgendes zu beachten:

Option	Bezeichnung
Umwelt	Die Workday-Umgebung, die Ihren benutzerdefinierten Bericht enthält.
ReportName (Berichtsname)	Der Name Ihres benutzerdefinierten Berichts.

4. Wählen Sie im Fenster Berichte auswählen einen benutzerdefinierten Bericht aus.
5. Klicken Sie auf Beenden.

6. Blenden Sie in der Ansicht Schema Explorer das benutzerdefinierter Berichtsschema ein, um dessen Struktur anzuzeigen.

Ergebnisse

Studio zeigt das benutzerdefinierte Berichtsschema in der Ansicht Schema Explorer an.

Nächste Maßnahme

Kopieren Sie Inhalt aus dem benutzerdefinierten Berichtsschema in Ihre Zwischenablage.

Konzept: Schema Explorer

In der Ansicht Schema Explorer werden WSDL- und XSD-Dateien im schreibgeschützten Format angezeigt. Sie können Inhalte aus diesen Dateien in Ihre Zwischenablage kopieren, sodass Sie die Daten in anderen Texteditor-Ansichten in Studio wiederverwenden können.

Sie können die Inhalte der Ansicht Schema Explorer filtern, indem Sie Schlüsselwörter in das Feld type filter text eingeben.

Standardmäßig wird die Ansicht Schema Explorer oben rechts auf dem Bildschirm angezeigt. Sie können die Ansicht Schema Explorer per Drag & Drop an die von Ihnen bevorzugte Position auf der Studio-Benutzeroberfläche ziehen.

MVEL

Konzept: MVEL-Grundlagen

Syntax

MVEL ist eine Ausdruckssprache für Java-basierte Anwendungen. Seine Syntax ähnelt der von Java, hat aber einige Zusätze, die es beim Abgleichen von Inhalten effizienter machen.

Hinweis:

- MVEL verwendet Java-Namespaces und -Klassen, kann aber diese beiden nicht selbst deklarieren.
- Im Gegensatz zu Java ist MVEL dynamisch typisiert, mit optionaler Typisierung.
- MVEL-Ausdrücke können einzelne Anweisungen oder mehrzeilige Schriftzeichen sein. Verwenden Sie bei mehrzeiligen Schriftzeichen ein Semikolon, um die einzelnen Anweisungen zu trennen.
- MvUp basiert auf dem Konzept des letzten Werts (Ende). Es unterstützt die return Schlüsselwort „“, aber im Allgemeinen nicht erforderlich.
- MVAL unterstützt Rangfolgeregeln für Operatoren und verwendet Klammern, um die Ausführungsreihenfolge zu steuern.
- Sie können MVEL-Zeichenfolgenlitterale in einfache oder doppelte Anführungszeichen setzen.
- Das Semikolon ist ein reserviertes Zeichen in MVEL. Sie können es nicht in Ausdrücken verwenden. Verwenden (char) 59 stattdessen. Beispiel: props['string']='a;b' löst einen Fehler aus. Dieser Code:

```
props['string']= (char) 59
props['string_new']='a'props['string']'b'
```

MSELECT-Ausdrücke können Folgendes umfassen:

- Boolesche Ausdrücke
- Methodenaufrufe
- Eigenschaftsausdrücke
- Variablenzuweisungen

Ein Beispiel für einen einfachen MSEL-Eigenschaftsausdruck:

```
context.errorMessage
```

Die Eigenschaft wird extrahiert `errorMessage` aus der Variable oder dem Kontextobjekt `context`.

MSELECT unterstützt alle üblichen Java-Vergleiche, mathematischen und logischen Operatoren. In dieser Tabelle sind weitere beachtenswerte MVEL-Operatoren zusammengefasst:

Operator	Beschreibung	Beispiel:
<code>new</code>	Instanziert ein Objekt.	<code>new String("foo")</code>
<code>with</code>	Führt mehrere Vorgänge für eine einzelne Objektinstanz aus.	<code>with (value) { name = 'Foo', age = 50, sex = Sex.MALE }</code>
<code>assert</code>	Bestätigt, dass ein Wert wahr ist, oder schlägt mit <code>AssertionError</code> fehl.	<code>assert foo != null</code>
<code>contains</code>	Bestimmt, ob der Wert links den Wert rechts enthält.	<code>var contains "Foo"</code>
<code>is oder instance of</code>	Bestimmt, ob der Wert links ein Element der Klasse rechts ist.	<code>var instanceof String</code>
<code>strsim</code>	Vergleicht Zeichenfolgen und gibt ihre Übereinstimmung als Prozentsatz zurück.	<code>"foobol" strsim "foobar"</code>
<code>soundlike</code>	Führt einen Lautvergleich zwischen zwei Zeichenfolgen durch.	<code>"foobar" soundlike "fubar"</code>
<code>&</code>	Führt einen bitweisen UND-Vorgang aus.	<code>foo & 5</code>
<code> </code>	Führt einen bitweisen ODER-Vorgang aus.	<code>foo 5</code>
<code>^</code>	Führt einen bitweisen XOR-Vorgang durch.	<code>foo ^ 5</code>
<code>+</code>	Überladener Operator für die Verkettung von zwei Zeichenfolgen.	<code>"foo" + "bar"</code>
<code>#</code>	Verkettet 2 Literale als Zeichenfolgen.	<code>8 # 9</code>
<code>in</code>	Untersucht Objekte in einer Sammlung.	<code>(foo in list)</code>
<code>=</code>	Weist einer Variable einen Wert zu.	<code>var = "foobar"</code>

Wertetest

In Java der Operator `==` prüft, ob Referenzen auf dasselbe Objekt verweisen. In Mpunkte jedoch der Vergleichsoperator `==` prüft, ob die Werte gleich sind. Der Ausdruck MVEL `foo == 'bar'` ist das Äquivalent von Java `foo.equals("bar")`.

Ein Beispiel für einen booleschen MSEL-Ausdruck:

```
user.name == 'John Smith'
```

Es wird zurückgegeben `true` wenn die `name`-Eigenschaft der Variable oder des Kontextobjekts `user` ist `John Smith`.

Mvge verwendet Arterzwingung. Wenn Sie zwei Werte unterschiedlichen Typs vergleichen, versucht MVEL, den Wert auf der rechten Seite auf den Typ auf der linken Seite zu erzwingen. Wenn das nicht möglich ist, wird versucht, den Wert auf der linken Seite auf den Typ auf der rechten Seite zu erzwingen. Beispiel:

```
"123" == 123;
```

Dieser Ausdruck gibt zurück `true`. Dies ist der Fall, da MVEL den Wert rechts in eine Zeichenfolge umwandelt, bevor der Vergleich durchgeführt wird.

Listen, Karten und Arrays

Schließen Sie MVEL-Listen in eckige Klammern ein. Trennen Sie Elemente durch Kommas. Beispiel:

```
["Paris", "London", "Berlin"]
```

Schließen Sie Mpunkte-Karten in eckigen Klammern. Trennen Sie Schlüssel und Werte durch Doppelpunkte. Beispiel:

```
["Foo" : "Bar", "Bar" : "Foo"]
```

Schließen Sie Mpunkte-Arrays in geschweiften Klammern ein. Trennen Sie Elemente durch Kommas. Beispiel:

```
{"Bill", "Bob", "Ben"}
```

Eigenschaften – Navigation

MVEL bietet eine einheitliche Syntax für den Zugriff auf Eigenschaften, statische Felder, Listen, Maps und Arrays.

In Java können Sie mit folgender Anweisung auf eine Eigenschaft eines Objekts zugreifen:

```
context.getException().getCause()
```

In Mpunkte können Sie auf dieselbe Eigenschaft mit folgendem Ausdruck zugreifen:

```
context.exception.cause
```

Greifen Sie auf dieselbe Weise auf Listen und Arrays zu. Beispiel:

```
parts[2]
```

Sie greifen auf Maps wie auf Arrays zu, können jedoch jedes Objekt als Indexwert übergeben. Beispiel:

```
props["foobar"]
```

Wenn eine Karte eine Zeichenfolge als Schlüssel verwendet, können Sie die Karte selbst als virtuelles Objekt behandeln. Beispiel:

```
props.foobar
```

MVAL behandelt Zeichenfolgen zu Indizierungszwecken als Arrays.

In diesem Beispiel wird zurückgegeben B:

```
foo = "Berlin";  
foo[0];
```

Zuweisung

Sie können Variablen in einem MVEL-Ausdruck zuweisen, die innerhalb des Ausdrucks verwendet oder zur Laufzeit extrahiert werden sollen.

Da MVEL eine dynamisch typisierte Sprache ist, müssen Sie keinen Typ angeben, wenn Sie eine neue Variable deklarieren. Aber Sie können es tun.

Beide folgenden Beispiele sind in Mpunkte gültig:

```
str = "Some string";  
String str = "Some string";
```

Bei der Zuweisung eines Werts zu einer typenvariablen versucht MVEL, eine automatische Typkonvertierung durchzuführen. Beispiel:

```
String num = 1;
```

Sie können den Wert einer dynamisch modellierten Variable in einen anderen Typ umwandeln. Beispiel:

```
String num = 1;
```

Abschnitt: Eingabe und Nötigung

MVEL verwendet eine dynamische Typisierung mit optionaler statischer Typisierung. Native Java-Objekte sind statisch typisiert. Um mit ihnen interagieren zu können, muss MVEL manchmal eine Typenzwangssteuerung durchführen, was sich auf die Performance auswirken kann.

MVEL wählt automatisch den Typ aus, der für den aktuellen Wert am besten geeignet ist. Sie müssen bei der Arbeit mit Zahlen vorsichtig sein. Bei der Gleitkomma-Arithmetik können Fehler auftreten, da MSEL automatisch von der unendlichen Genauigkeit abweicht `java.math.BigDecimal` bis `java.lang.Double` und `java.lang.Long`. Dies kann zu Problemen führen, wenn die Art des gespeicherten Werts nicht erwartet wird. Im Allgemeinen sollten Sie es vermeiden, in MVEL nicht-ganzzahlige Arithmetik mit Finanzwerten durchzuführen.

Die Typenzwangssteuerung von MVEL bedeutet, dass Sie Java-Methoden aufrufen können, ohne die Eingaben anzupassen. Der Interpreter oder Compiler analysiert die Typen, die Sie der Methode übergeben, und bestimmt, welcher Zwang erforderlich ist.

Im Falle eines Array-Typ-Konflikts versucht MVEL, das gesamte Array auf den benötigten Eingabetyp zu erzwingen.

MVEL führt alle erforderlichen Erzwingungen für eine statisch typisierte Variable durch, sofern der zugewiesene Wert erzwingbar ist. Ist dies nicht der Fall, tritt eine Ausnahme ein.

Konzept: MVEL und Studio

In Workday Studio können Sie mit MVEL Nachrichtenteile und Komponentenparameter in Ihren Assemblys prüfen und bearbeiten. Mit MVEL haben Sie folgende Möglichkeiten:

- Informationen aus einer Nachricht extrahieren.
- Zahlen und Datumsangaben formatieren.
- Bestimmte Zeichen in einer Zeichenfolge ersetzen.
- Integrationsattribute oder Startparameter abrufen.

Sie sollten vermeiden, mit MVEL große Datenzeichenfolgen zu bearbeiten. Beispiel: Das Konvertieren von Nachrichten in Zeichenfolgen kann zu Performance- und Skalierbarkeitsproblemen führen.

In einem MVEL-fähigen Feld können Sie die Tastenkombination Strg+Leertaste drücken, um ein Popup-Fenster für die Inhaltshilfe anzuzeigen.

MVEL-Ausdrücke können einzelne Anweisungen oder mehrzeilige Schriftzeichen sein. Verwenden Sie bei mehrzeiligen Schriftzeichen ein Semikolon, um die einzelnen Positionen zu trennen.

Anmerkung: Studio evaluiert mehrzeilige Ausdrücke als Einheit und nicht nacheinander.

MVEL-fähige Felder werden eingeblendet, um neue Positionen aufzunehmen. Nach 6 Positionen werden sie nicht mehr eingeblendet und können gescrollt werden. Studio wendet Syntaxfärbung auf MVEL-Ausdrücke an.

MVEL implementiert die Operatorrangfolge während der Ausführung. Sie können Klammern verwenden, um die Ausführungsreihenfolge von Mpunkten zu steuern.

Anmerkung: Da sie im Speicher verarbeitet werden, gibt es eine Beschränkung von 1 Million Zeichen für XPath-Methoden wie `xpath`, `xpathBund` und `xpathF`. Sie erhalten keinen Fehler, wenn Ihre Abfrage diesen Grenzwert überschreitet. Sie sollten Streaming-APIs verwenden, z. B. für den Schreibschritt `xml-stream`-Element, wenn Sie große Nachrichten verarbeiten müssen.

Referenz: Studio-Hilfsobjekte

Studio stellt eine Reihe von Hilfsobjekten bereit, mit denen Sie mithilfe von MVEL auf die Laufzeiteigenschaften einer Integration zugreifen können. Sie können die Hilfsobjekte verwenden, um einfache Änderungen an Laufzeitwerten vorzunehmen oder diese zu extrahieren und an Komponenten in der Baugruppe zu übergeben.

Hilfsobjekt	Zweck
context	Diese Domäne bietet Zugriff auf <code>MediationContext</code>-Objekt, d. h. auf den dynamischen Status der Integration zur Laufzeit. Die <code>MediationContext</code> Das Objekt "" enthält Links zur aktuellen Meldung, zu Medien-Eigenschaften, Medien und aktuellen Fehlerinformationen. Beispiel: Dieser Ausdruck ruft den Wert der Eigenschaft <code>ab myProperty</code> aus dem <code>MediationContext</code>: <pre>context.getProperty('myProperty')</pre> Beispiel: Dieser Ausdruck ruft die Eigenschaft <code>ab someProperty</code> aus dem <code>MediationContext</code>

Hilfsobjekt	Zweck
	wenn der Eigenschaftswert auf festgelegt ist someValue: <pre>context.getProperty('someProperty').equals('someValue')</pre>
da	Das Helper-Objekt „Dokumentzugriff“. Bietet Zugriff auf die Liste der Dokumente, die an das Integrationsereignis angehängt wurden, bevor diese Integration ausgeführt wurde. Kombiniert effektiv einen Webservice-Aufruf Get Event Documents mit dem Abruf der Dokumente, die in der Webservice-Antwort referenziert werden. Die da Für das Helper-Objekt gelten einige Einschränkungen: • Er muss explizit Logik enthalten, um Fälle zu behandeln, in denen eine unerwartete Anzahl von Anhängen verfügbar ist. • Es kann nicht auf den Dateityp des Anhangs zugreifen. • Es kann nicht auf die Zeichencodierung des Anhangs zugreifen. Die Router-Strategie doc-iterator verarbeitet eine unerwartete Anzahl von Anhängen automatisch und kann sowohl auf den MIME-Typ als auch auf die Zeichencodierung zugreifen. Für eine Integration ist es oft sinnvoller, diese Strategie zum Abrufen von Eingabedokumenten zu verwenden, als dieses Hilfeobjekt. Beispiel: Dieser Ausdruck ruft ein Dokument aus einem Integrationsereignis mit dem Dateinamen eib_transform_output.txt ab und fügt eine Variable mit dem Namen hinzu var1: <pre>da.toVar('eib_transform_output.txt', 'var1')</pre> Beispiel: Dieser Ausdruck ruft ein Dokument mit dem angegebenen Label ab: <pre>da.getDocsMatching('label')</pre>
ftp	Diese Domäne bietet Zugriff auf die Liste der Dateien, die beim Ausgeben der LIST-Anforderung an einen FTP-, FTP- oder SFTP-Server zurückgegeben werden. Beispiel: Dieser Ausdruck ruft eine Liste von Dateien für eine URL ab: <pre>ftp.list('URL')</pre>
intsys	Bietet bequemen und leistungsstarken Zugriff auf die Integrationssystemkonfiguration, wie sie

Hilfsobjekt	Zweck
	vom Webservice-Aufruf "Get Integration Systems" zurückgegeben wird. Beispiel: Dieser Ausdruck ruft ein Integrationsattribut mit dem Namen ab username: <pre>intsys.getAttribute('username')</pre> Beispiel: Dieser Ausdruck wird als "Wahr" angezeigt, wenn ein Integrationservice mit dem angegebenen Namen vorhanden ist: <pre>intsys.isServiceEnabled('name')</pre> Anmerkung: Wenn Sie verwenden intsys Um auf ein Attribut zuzugreifen, müssen Sie zuerst einen Service für definieren workday - in Transport für das Integrationssystem.
lp	Bietet Zugriff auf Startparameter und andere Informationen im Integrationsstartereignis. Beispiel: Dieser Ausdruck ruft den Wert eines Startparameters mit dem angegebenen Namen ab: <pre>lp.getSimpleData('name')</pre> Beispiel: Dieser Ausdruck wird als „true“ angezeigt, wenn ein Parameter mit dem angegebenen Label vorhanden ist: <pre>lp.exists('label') == true</pre> Anmerkung: Vor der Verwendung von lp Variable: Sie müssen der Variable die XML-Startparameter zuweisen wd.launchparameters.
message	Diese Domäne bietet Zugriff auf die aktuelle Nachricht und ihre Header-Informationen. Verwenden Sie dieses Objekt, um der Nachricht neuen Inhalt zuzuweisen. Beispiel: Dieser Ausdruck ruft den Root-Teil einer Nachricht als Text aus ab MediationMessage Schnittstelle: <pre>message.rootPartAsText</pre> Beispiel: Dieser Ausdruck ruft ab MediationContext Variablen: <pre>message.variables</pre>

Hilfsobjekt	Zweck
mtable	Diese Domäne bietet Zugriff auf MTableAdapter -Schnittstelle, die die Vorgänge für eine definiert mtable Instanz. Beispiel: Dieser Ausdruck legt Daten in fest mtable: <pre>mtable['my_property'].set(0, 'Data_Column', 'my_test_data')</pre>
parts	Gewährt Zugriff auf die einzelnen Teile der aktuellen Nachricht. Die meisten Nachrichten in Workday-Integrationen haben einen einzigen Teil, den Root-Teil. Verwenden parts[0] , um auf den Root-Teil zuzugreifen. Beispiel: Dieser Ausdruck ruft den Header ab someHeader aus dem Root-Teil: <pre>parts[0].getHeader('someHeader')</pre> Beispiel: Dieser Ausdruck implementiert die xpathB -Methode zur Evaluierung eines booleschen XPath-Ausdrucks: <pre>parts[0].xpathB('/a/b')</pre>
props	Diese Domäne bietet Zugriff auf MediationContext -Eigenschaften Beispiel: Dieser Ausdruck ruft den Wert der Eigenschaft ab myProperty aus dem MediationContext: <pre>props.myProperty</pre> Beispiel: Dieser Ausdruck ruft den Wert der Eigenschaft ab bean.property aus dem MediationContext: <pre>props['bean.property']</pre>
spring	Diese Domäne bietet Zugriff auf die Erstellungs- und Abruf-Beans, die in der Quellkonfiguration einer Baugruppe definiert sind. Beispiel: Dieser Ausdruck ruft die ab name Attribut der Bean Fred im Kontext „Spring“: <pre>spring.getBean('fred').name</pre> Beispiel: Dieser Ausdruck greift auf den Kontext derSpring-Anwendung zu: <pre>spring.parent ApplicationContext</pre>

Hilfsobjekt	Zweck
util	Eine Reihe von Dienstprogrammfunktionen für die Ausführung allgemeiner Vorgänge, z. B. das Escape von Werten für die Einfügung in XML oder das Erstellen einer HTTPS-URL. Beispiel: Dieser Ausdruck prüft, ob die aktuelle Nachricht in der MediationContext ist die letzte Meldung, die von der Split-Komponente generiert wird: <pre>props['is.last.record'] = util.isLastMessageInBatch()</pre> Einige util -Funktionen unterliegen erheblichen Einschränkungen. Beispiel: Die xpathToCommaDelimString Die - Funktion erstellt eine durch Kommas getrennte Liste aus den Zeichenfolgenwerten der Knotensequenz, die von einem XPath-Ausdruck zurückgegeben wird, der auf eine bestimmte XML-Zeichenfolge angewendet wird. Anschließend können Sie die durch Kommas getrennte Liste mithilfe von Java-Funktionen zur Zeichenfolgenmanipulation in Mpunkte aufteilen und bearbeiten. Für die Funktion gelten jedoch folgende Einschränkungen: • Es wird nur XPath 1 unterstützt. • Es können keine CSV-Dateien erstellt werden, da Zeilenumbrüche, Kommas und Anführungszeichen nicht korrekt mit Escapezeichen versehen werden. • Daten, die Kommas enthalten, können nicht zuverlässig getrennt werden. • Die Anforderung von XML als Zeichenfolge fördert nichtskalierbare Methoden, wie z. B. die Konvertierung der aktuellen Kontextnachricht oder Variable in eine Zeichenfolge. Anmerkung: Workday empfiehlt aus Gründen der Sicherheit und Rückwärtskompatibilität dringend die Verwendung expliziter, versionierter Webservice-Anforderungen. Wird verwendet MVELUtilHelper.getLatestWSVersion() in Integrationen ist nicht die Best Practice und kann zu Integrationen mit unerwarteten Fehlern oder Ergebnissen führen.
vars	Diese Domäne bietet Zugriff auf MediationContext Variablen. Beispiel: Dieser Ausdruck ruft die Länge der Variable ab myVariable: <pre>vars['myVariable'].length</pre>

Hilfsobjekt	Zweck
	Beispiel: Dieser Ausdruck ruft die Länge der Variable ab myVariable: <pre>vars['myVariable'].length</pre>
xmldiff	Diese Domäne bietet programmgesteuerten Zugriff auf die Ergebnisse des Dokumentvergleichs mit der Komponente "xmldiff". Beispiel: Dieser Ausdruck prüft, ob es Unterschiede zwischen zwei Dokumenten gibt: <pre>xmldiff.isDifferent() == true</pre> Beispiel: Dieser Ausdruck ruft eine Liste der Unterschiede zwischen zwei Dokumenten ab: <pre>xmldiff.changes</pre>

Referenz: Hilfsobjekt für den Dokumentenzugriff

Hilfreiche Funktionen

Name	Argumente	Art des Rückgabewerts	Anmerkungen
getData	Integer-Index	Quelle	Ruft das Dokument am angegebenen nullbasierten Index ab. Verwenden Sie in Kombination mit <code>message.setMessage</code>-Hilfsfunktion, um die aktuelle Meldung auf speichereffiziente Weise mit den abgerufenen Daten zu füllen. Beispiel: Der folgende Code füllt die aktuelle Nachricht mit einer UTF-8-XML-Nachricht aus, die aus dem ersten Anhang abgerufen wird: <pre>message.setMessage(da.getData('text/xml'))</pre> Gibt einen Fehler zurück, wenn kein Integrationsanhang mit dem angegebenen Index vorhanden ist.

Name	Argumente	Art des Rückgabewerts	Anmerkungen
	Dateiname für Zeichenfolge	Quelle	Ruft das Dokument mit dem angegebenen Dateinamen ab. Verwenden Sie in Kombination mit <code>message.setMessage</code>-Hilfsfunktion, um die aktuelle Meldung auf speichereffiziente Weise mit den abgerufenen Daten zu füllen. Beispiel: Dieser Code fügt in der aktuellen Meldung einen Nachrichtenanhang im UTF-8-XML-Format mit dem Namen "input.xml" hinzu: <pre>message.setMessage(da.getDa 'text/xml')</pre> Fügt einen Fehler hinzu, wenn kein Integrationsanhang mit dem angegebenen Dateinamen vorhanden ist.
getDocumentIndexes	Listen<String>Labels	Liste<Ganzzahl>	Gibt eine Java-Liste mit Ganzzahlen zurück, die die Indizes der Dokumente angibt, die mit allen angegebenen Labels getaggt wurden. Beachten Sie: Wenn eines der Labels "*" lautet, wird jedes Label abgeglichen und die Funktion gibt alle Indizes für Dokumente mit mindestens einem Label zurück.
getDocsMatching	Liste<String>	Liste<Quelle>	Ruft Dokumente ab, die mit allen angegebenen Labels getaggt wurden. Anmerkung: Workday speichert Dokumente, die kleiner als 100 KB sind, im Speicher und größere Dokumente auf der Festplatte.

Name	Argumente	Art des Rückgabewerts	Anmerkungen
			Beachten Sie die Grenzwerte für Laufzeitspeicher und Festplatte.
	Zeichenfolgenlabel	Quelle	Ruft Dokumente ab, die mit dem angegebenen Label getaggt wurden. Anmerkung: Mit Label ist hier das Label für den Integrationsanhang gemeint. Gleichbedeutend mit Dokument-Tag.
	Labels für Zeichenfolge[]	Quelle	Ruft Dokumente ab, die mit allen Labels im Zeichenfolgen-Array getaggt sind.
getFileName	Integer-Index	Zeichenfolge	Ruft den Dateinamen mit dem angegebenen Index ab.
	Zeichenfolgenlabel	Liste<Zeichenfolge>	Ruft eine Java-Liste der Dateinamen aller Dokumente ab, die mit dem angegebenen Label getaggt wurden.
	Liste<Zeichenfolge> Labels	Liste<Zeichenfolge>	Ruft eine Java-Liste der Dateinamen aller Dokumente ab, die mit den angegebenen Labels getaggt wurden.
	Zeichenfolgenlabel, ganzzahliger Index	Zeichenfolge	Ruft den n-ten Dateinamen aller Dokumente ab, die mit dem angegebenen Label getaggt sind. Funktional äquivalent zu diesem Code: <pre>da.getFileNames(label).get(</pre> In den meisten Fällen müssen Sie "" verwenden <pre>da.getFileNames(label)</pre> um die Anzahl übereinstimmender Dateinamen zu ermitteln, bevor Sie einen Index verwenden können. Die einzige

Name	Argumente	Art des Rückgabewerts	Anmerkungen
			Ausnahme ist der begrenzte Fall, in dem Sie den ersten Wert (Nullindex) benötigen.
	Liste<Zeichenfolge> Labels, ganzzahliger Index	Zeichenfolge	Ruft den n-ten Dateinamen aller Dokumente ab, die mit den angegebenen Labels getaggt sind. Funktional äquivalent zu diesem Code: <pre>da.getFileNames(labels).get</pre> In den meisten Fällen müssen Sie "" verwenden da.getFileNames(labels) um die Anzahl übereinstimmender Dateinamen zu ermitteln, bevor Sie einen Index verwenden können. Die einzige Ausnahme ist der begrenzte Fall, in dem Sie den ersten Wert (Nullindex) benötigen.
	Labels für Zeichenfolge[], ganzzahliger Index		Ruft den n-ten Dateinamen aller Dokumente ab, die mit den angegebenen Labels getaggt sind. Funktional äquivalent zu diesem Code: <pre>da.getFileNames(java.util.A</pre> In den meisten Fällen müssen Sie "" verwenden da.getFileNames(labels) um die Anzahl übereinstimmender Dateinamen zu ermitteln, bevor Sie einen Index verwenden können. Die einzige Ausnahme ist der begrenzte Fall, in dem Sie den ersten Wert (Nullindex) benötigen.

Name	Argumente	Art des Rückgabewerts	Anmerkungen
hasFile	Zeichenfolge für Dateinamen	Boolesch	Ermittelt, ob ein Integrationsanhang mit dem angegebenen Dateinamen vorhanden ist.
	Zeichenfolgenlabel, Dateiname für Zeichenfolge	Boolesch	Ermittelt, ob ein Integrationsanhang mit dem angegebenen Dateinamen vorhanden und mit dem angegebenen Label getaggt ist.
	Liste<Zeichenfolge> Labels, Dateiname für Zeichenfolge	Boolesch	Bestimmt, ob es einen Integrationsanhang mit dem angegebenen Dateinamen gibt, der mit allen Labels in der Liste getaggt ist.
sort	Reihenfolge der Zeichenfolgen	Keine	Sortiert die angehängten Dokumente nach Dateiname. Gültige Werte für die Reihenfolge sind: • "FILENAME_ASCENDING" • "FILENAME_DESCENDING" Alle Anhänge ohne Dateinamen werden an die letzte Stelle gesetzt.
size	Keine	Ganzzahl	Gibt die Anzahl der angehängten Dokumente zurück.
	Liste<Zeichenfolge> Labels	Ganzzahl	Gibt die Anzahl der Dokumente zurück, die mit allen angegebenen Labels getaggt wurden.
	Zeichenfolgenlabel	Ganzzahl	Gibt die Anzahl der Dokumente zurück, die mit dem angegebenen Label getaggt wurden.
	Labels für Zeichenfolge[]	Ganzzahl	Gibt die Anzahl der Dokumente zurück, die mit allen Labels im Zeichenfolgen-Array getaggt sind.
toVar	Integer-Index, Zeichenfolge var_name	MessageVariable	Ruft das Dokument am angegebenen

Name	Argumente	Art des Rückgabewerts	Anmerkungen
			Index ab und platziert es speichereffizient in der angegebenen Assembly-Variablen. Beispiel: <pre>da.toVar(0, 'myInputDocument')</pre>
	Zeichenfolge für Dateiname, Zeichenfolge für var_name	MessageVariable	Ruft das Dokument mit dem angegebenen Dateinamen ab und speichert es speichereffizient in der angegebenen Assembly-Variablen. Beispiel: <pre>da.toVar('WorkerData.xml', 'myInputDocument')</pre>

Referenz: Helper-Objekt für Startparameter

Hilfreiche Eigenschaften

Name	Anmerkungen
integrationEventWID	Gibt die WID für das Integrationsereignis zurück, das sich auf den aktuellen Start der Integration bezieht. Hilfreich bei Webservice-Aufrufen, z. B. Get_Event_Documents, Put_Integration_Message, oder Put_Integration_Event. Stellt automatisch einen Standardparameter für die allgemeinen Komponenten wcc://GetEventDocuments, wcc://PutIntegrationMessage und wcc://PutIntegrationEvent bereit.
integrationSystem	Gibt den zurück Integration_System_ID für das aktuelle Integrationssystem. Hilfreich beim Ausfüllen von Webservice-Aufrufen für Ereignisabonnements oder Anforderungen zum Überschreiben von Feldern.
integrationSystemRefWID	Gibt die WID für das aktuelle Integrationssystem zurück. Eine völlig äquivalente Alternative zu integrationSystem Eigenschaft
parentEventWID	Gibt die WID für das übergeordnete Ereignis des aktuellen Starterereignisses zurück.

Name	Anmerkungen
sentOn	Gibt das Datum und die Uhrzeit des Integrationsstarts im Format als Datum und Uhrzeit nach ISO-8601 zurück.
sentOnAsDate	Gibt das Datum und die Uhrzeit des Integrationsstarts in Form von zurück <code>java.util.Date</code> Das Objekt

Hilfreiche Funktionen

Anmerkung:

Einige dieser Funktionen geben einen Startparameter allein über seinen Namen an. Andere geben sowohl den Namen des Startparameters als auch den Namen des Integrationservice an, der ihn bereitstellt. Letztere sind hilfreich, wenn ein Startparameter mit demselben Namen in mehreren Integrationservices vorhanden ist, die im Integrationssystem konfiguriert sind. Die meisten Integrationen müssen keine Startparameternamen berücksichtigen, die in mehreren Services doppelt vorhanden sind, sodass diese Funktionen weniger häufig benötigt werden.

Name	Argumente	Art des Rückgabewerts	Anmerkungen
exists	Name der Zeichenfolge	Boolesch	Gibt „Wahr“ zurück, wenn der benannte Startparameter einen nicht leeren Wert hat.
	Zeichenfolge service_name, Zeichenfolgenname	Boolesch	Gibt „Wahr“ zurück, wenn der benannte Startparameter im angegebenen Integrationservice einen nicht leeren Wert hat. Diese Domäne ist der Version mit einem Argument nur dann vorzuziehen, wenn ein Startparameter mit demselben Namen in mehreren Integrationservices vorhanden ist, die im Integrationssystem aktiviert sind.
getDate	Name der Zeichenfolge	Zeichenfolge	Gibt den als Zeichenfolge formatierten Datumswert des Laufzeitparameters zurück. Das Format ist JJJJ-MM-TTZ oder JJJJ-MM-TT+/-Versatz.

Name	Argumente	Art des Rückgabewerts	Anmerkungen
	Zeichenfolge service_name, Zeichenfolgenname	Zeichenfolge	Gibt den als Zeichenfolge formatierten Datumswert des Laufzeitparameters des angegebenen Services zurück. Das Format ist JJJJ-MM- DTZ oder JJJJ-MM-TT +/-Zeitversatz.
getParameterData	Name der Zeichenfolge	org.w3c.dom.Element	Gibt das XML- Element zurück Integration_Runtime_Paramete Domäne enthält den benannten Startparameter aus dem Launch_Integration_Event Die Meldung, die beim Start an die Integration gesendet wird.
	Zeichenfolge service_name, Name der Zeichenfolge	org.w3c.dom.Element	Gibt das XML- Element zurück Integration_Runtime_Paramete Domäne enthält den benannten Startparameter des angegebenen Service aus dem Launch_Integration_Event Die Meldung, die beim Start an die Integration gesendet wird.
getReferenceData	Name der Zeichenfolge, Zeichenfolgenart	Zeichenfolge	Gibt die Referenz-ID der angegebenen Art für den Startparameter mit dem angegebenen Namen zurück. Beispiel: Dieser Code gibt die Personalnummer für einen Startparameter mit dem Namen „Mitarbeiter“ zurück: <code>lp.getReferenceData('Employ 'Employee_ID')</code>
getReferenceDataList	Name der Zeichenfolge, Zeichenfolgenart	Liste<Zeichenfolge>	Gibt eine Java-Liste zurück, die keine oder mehr Referenz-IDs aus einem Startparameter

Name	Argumente	Art des Rückgabewerts	Anmerkungen
			für mehrere Instanzen enthält.
	Zeichenfolge service_name, Zeichenfolgenname, Zeichenfolgeart	Liste<Zeichenfolge>	Gibt eine Java-Liste mit null oder mehr Referenz-IDs aus einem Multiinstanz-Startparameter eines angegebenen Service zurück.
getSequencedValue	Name der Zeichenfolge	Zeichenfolge	Gibt den aktuellen Wert des Sequenzgenerators mit dem angegebenen Namen zurück.
	Zeichenfolge service_name, Zeichenfolgenname	Zeichenfolge	Gibt den aktuellen Wert des angegebenen Sequenzgenerators zurück, der für den angegebenen Service deklariert wurde.

Referenz: Util-Hilfsobjekt

Hilfreiche Funktionen

Name	Argumente	Art des Rückgabewerts	Anmerkungen
cleanString	Zeichenfolgenwert	Zeichenfolge	Wird verwendet, um Eingabezeichen zu maskieren, die in XML-Dokumenten mit Escape-Zeichen maskiert werden müssen. Beispiel: <pre><Value>@{util.cleanString(p root/row/ a_value')}}</ Value></pre>
getExtForContentType	Zeichenfolge für MIME-Typ	Zeichenfolge	Ermöglicht einer Integration, basierend auf dem angegebenen MIME-Typ eine Dateierweiterung wie .txt oder .xml abzurufen.
getFilenameFromContentDisposition	Zeichenfolge für Content-Disposition	Zeichenfolge	Extrahiert einen Dateinamen aus einer Inhaltsausschlusszeichenfolge. Einige Build-

Name	Argumente	Art des Rückgabewerts	Anmerkungen
			Komponenten, wie z. B. die Dokument-Iterator-Strategie, laden den Nachrichten-Header für den Inhaltsversatz mit Informationen zum Dateinamen. Beispiel: Dieser Code gibt den Wert zurück <code>testfile.pdf</code>: <pre>util.getFilenameFromContent(filename=testfile.pdf')</pre>
readFileToVar	Zeichenfolge Var_Name, Zeichenfolge Dateiname		Ermöglicht das Lesen von Nicht-Text-Dateien in eine Variable. Dies ist die einzige bereitgestellte Möglichkeit, mit der Nicht-Text-Dokumente geladen werden können. Beachten Sie, dass das Attribut „static-file“ des Schritts write nur Textdokumente in eine Nachricht oder Variable laden kann. Die Datei wird relativ zum Verzeichnis WSAR-INF angegeben. Beispiel: Dieser Code liest die Bilddatei „icon-32.png“ aus dem Verzeichnis WSAR-INF in die Kontextvariable v1: <pre>util.readFileToVar('v1', 'icon-32.png')</pre>
readFileToVar	Zeichenfolge var_name, Zeichenfolge Dateiname, Zeichenfolge MIME-Typ		Ermöglicht das Lesen von Nicht-Text-Dateien in eine Variable, deren MIME-Typ Sie angeben. Dies ist die einzige bereitgestellte Möglichkeit, mit der Nicht-Text-Dokumente geladen werden können. Beachten

Name	Argumente	Art des Rückgabewerts	Anmerkungen
			Sie, dass das Attribut „static-file“ des Schritts write nur Textdokumente in eine Nachricht oder Variable laden kann. Die Datei wird relativ zum Verzeichnis WSAR-INF angegeben. Beispiel: Dieser Code liest die Bilddatei icon-32.png aus dem Verzeichnis WSAR-INF in die Kontextvariable v1 mit dem MIME-Typ Bild/png: <pre>util.readFileToVar('v1', 'icon-32.png', 'image/png')</pre>

Referenz: MVEL-Beispielausdrücke in Studio

Variabel	Beispielausdruck	Beschreibung
context	context.getProperty('myProperty')	Ruft den Wert der Eigenschaft ab myProperty aus dem MediationContext.
	context.getProperty('someProperty').equals('someValue')	Ruft die Eigenschaft ab someProperty aus dem MediationContext wenn der Eigenschaftswert auf festgelegt ist someValue.
da	da.toVar('eib_transform_output.txt', 'var1')	Ruft ein Dokument aus einem Integrationsereignis mit dem Dateinamen eib_transform_output.txt ab und fügt eine Variable mit dem Namen hinzu var1.
	da.getDocsMatching('label')	Ruft ein Dokument mit dem angegebenen Label ab.
ec	ec.getAttribute('attributeName')	Ruft einen Attributnamen ab.
	ec.integrationMapLookup('map', 'key')	Ruft einen Integrations-Map-Wert ab.
env	env['cc.hostname']	Ruft die ab cc.hostname Umgebungseigenschaft

Variabel	Beispielausdruck	Beschreibung
		festgelegt als <code>-Dcc.hostname</code> in der Befehlszeile.
	<code>env.containsKey('key')</code>	Wird als "Wahr" angezeigt, wenn Umgebungseigenschaften den angegebenen Schlüssel enthalten.
<code>ftp</code>	<code>ftp.list('URL')</code>	Ruft eine Liste von Dateien für eine URL ab.
<code>intsys</code>	<code>intsys.getAttribute('username')</code>	Ruft ein Integrationsattribut mit dem Namen ab username.
	<code>intsys.isServiceEnabled('name')</code>	Wird als "Wahr" angezeigt, wenn ein Integrationsservice mit dem angegebenen Namen vorhanden ist.
<code>lp</code>	<code>lp.getSimpleData('name')</code>	Ruft den Wert eines Startparameters mit dem angegebenen Namen ab.
	<code>lp.exists('label') == true</code>	Wird als "Wahr" angezeigt, wenn ein Parameter mit dem angegebenen Label vorhanden ist.
<code>message</code>	<code>message.rootPartAsText</code>	Ruft den Root-Teil einer Nachricht als Text aus ab

Variabel	Beispielausdruck	Beschreibung
		MediationMessage - Schnittstelle.
	<code>message.variables</code>	Ruft abMediationContext Variablen.
<code>mtable</code>	<code>mtable['my_property'].set(0, 'Data_Column', 'my_test_data')</code>	Legt Daten in einer fest mtable.
	<code>mtable['my_property'].get(0, 'Data_Column')</code>	Ruft Daten aus ab mtable.
<code>parts</code>	<code>parts[0].getHeader('someHeader')</code>	Ruft den Header ab SomeHeader aus dem Root-Teil.
	<code>parts[0].xpathB('/a/b')</code>	Implementiert xpathB - Methode zur Evaluierung eines booleschen XPath-Ausdrucks.
<code>props</code>	<code>props.myProperty</code>	Ruft den Wert der Eigenschaft ab myProperty aus dem MediationContext.
	<code>props['bean.property']</code>	Ruft den Wert der Eigenschaft ab bean.property aus dem MediationContext.
<code>spring</code>	<code>spring.getBean('fred').name</code>	Ruft die ab name Attribut der Bean Fred im Kontext „Spring“.
	<code>spring.parentApplicationContext</code>	Greift auf den Kontext derSpring-Anwendung zu.
<code>util</code>	<code>props['is.last.record'] = util.isLastMessageInBatch()</code>	Prüft, ob die aktuelle Nachricht in der ist MediationContext ist die letzte Meldung, die von der Split-Komponente generiert wird.
	<code><ns:Message_Summary>@{util:cleanString(props['is.message.summary'])}</ns:Message_Summary></code>	Implementiert cleanString(String) -Funktion, um sicherzustellen, dass eine

Variabel	Beispielausdruck	Beschreibung
		Aggregationskomponente wohlgeformtes XML generiert.
vars	<code>vars['myVariable'].length</code>	Ruft die Länge der Variable ab myVariable.
	<code>vars['myVariable'].mimeType</code>	Ruft den der Variable zugeordneten MIME-Typ ab myVariable.
xmldiff	<code>xmldiff.isDifferent() == true</code>	Workday prüft, ob es Unterschiede zwischen zwei Dokumenten gibt.
	<code>xmldiff.changes</code>	Ruft eine Liste der Differenzen zwischen zwei Dokumenten ab.

Referenz: MVEL-Syntaxbeispiele für automatische Konvertierung in Studio

Funktion	Beispielausdruck	Beschreibung
Automatische Konvertierung einer Zeichenfolge in eine Variable.	<code>vars['fred'] = '<hello>world</hello>'</code>	Erstellt eine neue Variable mit dem Namen fred mit dem Inhalt <hello>world</hello>. Verwenden Sie diese Methode nur, um kleine Nachrichten zu erstellen. Die Verwendung von Ausdrücken zum Erstellen großer Zeichenfolgen kann zu schlechter Performance führen. In den meisten Fällen sollten Sie die verwenden write Schritt zum Erstellen von Inhalten für eine Variable.
Stellt Informationen zum Inhalt und MIME-Typ für eine Variable bereit.	<code>vars['fred'] = ['Hello World', 'text/plain']</code>	Erstellt eine neue Variable mit dem Namen fred mit dem Inhalt Hello World und MIME-Typ text/plain. Verwenden Sie diese Methode nur, um kleine Nachrichten zu erstellen. Die Verwendung von Ausdrücken zum Erstellen großer Zeichenfolgen kann zu schlechter Performance führen. In den meisten Fällen sollten Sie die verwenden write Schritt zum Erstellen von Inhalten für eine Variable und zum Konfigurieren des MIME-Typs.
Aus einem Anhang in eine Variable kopieren.	<code>vars['fred'] = parts[1]</code>	Erstellt eine neue Variable fred aus dem ersten Anhang der aktuellen Nachricht.

Funktion	Beispielausdruck	Beschreibung
		In den meisten Fällen sollten Sie die verwenden copy , um Eingaben aus einem Anhang zu übernehmen.
Kopieren aus einer Meldung in eine Variable.	<code>vars['fred'] = message</code>	Erstellt eine neue Variable fred aus der gesamten Nachricht. In den meisten Fällen sollten Sie die verwenden copy , um die Eingabe aus der aktuellen Meldung zu übernehmen.
Stellen Sie Informationen zum Inhalt und zum MIME-Typ für eine Variable bereit, deren Wert ein Ausdruck ist.	<code>vars['fred'] = [23 + 5, 'text/plain']</code>	Erstellt eine neue Variable mit dem Namen fred mit Inhalt 28 und MIME-Typ text/plain.
Erstellen Sie eine Variable aus einem XPath-Fragment im Root-Teil der Nachricht.	<code>vars['fred'] = parts[0].xpathF('/dink/donk')</code>	Erstellt eine neue Variable mit dem Namen fred mit einem XPath-Fragment aus dem Root-Teil der aktuellen Meldung. Verwenden Sie diese Methode nur, um kleine Nachrichten zu erstellen. Die Verwendung von XPath-Ausdrücken zum Erstellen großer Zeichenfolgen kann zu einer schlechten Performance führen.
Variable auf null setzen	<code>vars['fred'] = null</code>	Entfernt die Variable mit dem Namen fred.
Weisen Sie dem Root-Teil Inhalt zu.	<code>parts[0] = '<hello>world</hello>hello>'</code>	Weist den Inhalt zu <code><hello>world</hello></code> zum Root-Teil. Verwenden Sie diese Methode nur, um kleine Nachrichten zu erstellen. Die Verwendung von Ausdrücken zum Erstellen großer Zeichenfolgen kann zu schlechter Performance führen. In den meisten Fällen sollten Sie die verwenden write Schritt zum Zuweisen von Inhalt zu einem Anhang und zum Konfigurieren des MIME-Typs.
Weisen Sie dem ersten Anhang Inhalt zu.	<code>parts[1] = ['Hello World', 'text/plain']</code>	Weist den Inhalt zu Hello world und MIME-Typ von text/plain an den ersten Anhang. Verwenden Sie diese Methode nur, um kleine Nachrichten zu erstellen. Die Verwendung

Funktion	Beispielausdruck	Beschreibung
		von Ausdrücken zum Erstellen großer Zeichenfolgen kann zu schlechter Performance führen. In den meisten Fällen sollten Sie die verwenden <code>write</code> Schritt zum Zuweisen von Inhalt zu einem Anhang und zum Konfigurieren des MIME-Typs.
Weisen Sie dem Root-Anhang Inhalt zu und geben Sie den MIME-Typ an.	<code>parts[0] = [23 + 5, 'text/plain']</code>	Weist dem Root-Teil Inhalt zu 28 und MIME-Typ <code>text/plain</code> .
Weisen Sie dem Nachrichtenteil einen Inhalt aus einer Variable zu.	<code>parts[0] = vars['fred']</code>	Weist den Inhalt der Variable zu <code>fred</code> zu einem Nachrichtenteil. In den meisten Fällen sollten Sie die verwenden <code>copy</code> - Schritt, um die Eingabe aus einer Variable zu übernehmen und sie einem Nachrichtenteil zuzuweisen.
Entfernen Sie den Root-Teil.	<code>parts[0] = null</code>	Entfernt den Root-Teil.
Sie können einen Anhang kopieren.	<code>parts[2] = parts[1]</code>	Workday kopiert den ersten Anhang in einen zweiten Anhang. In den meisten Fällen sollten Sie die verwenden <code>copy</code> Mit diesem Schritt können Sie Eingaben aus einem Anhang übernehmen und einem anderen Anhang zuweisen.
Erstellen Sie eine neue Nachricht mit dem Standard-MIME-Typ.	<code>context.message = '<hello>world</hello>'</code>	Erstellt eine neue Nachricht <code><hello>world</hello></code> mit dem Standard-MIME-Typ <code>text/xml</code>. In den meisten Fällen sollten Sie die verwenden <code>write</code> Schritt zum Erstellen einer neuen Nachricht.
Erstellen Sie eine neue Nachricht mit dem MIME-Typ <code>text/plain</code> .	<code>context.message = ['Hello world', 'text/plain']</code>	Erstellt eine neue Nachricht <code>Hello world</code> mit dem MIME-Typ <code>text/plain</code>. In den meisten Fällen sollten Sie die verwenden <code>write</code> Schritt zum Erstellen einer neuen Nachricht.
Neue Nachricht aus variablem Inhalt erstellen	<code>context.message = vars['fred']</code>	Erstellt eine neue Nachricht mit dem gleichen Inhalt und

Funktion	Beispielausdruck	Beschreibung
		der gleichen MIME-Art wie die Variable fred. In den meisten Fällen sollten Sie die verwenden copy Schritt zum Erstellen einer neuen Nachricht aus einer Variable.
Erstellen Sie eine neue Nachricht aus dem Root-Teil der aktuellen Nachricht.	<code>context.message = parts[0]</code>	Erstellt eine neue Nachricht aus dem Root-Teil der aktuellen Nachricht. In den meisten Fällen sollten Sie die verwenden copy Schritt zum Erstellen einer neuen Nachricht aus einem Nachrichtenteil.

Referenz: MVEL-Methoden

Methode	Beschreibung	Beispielcode	Beispielausgabe
<code>getClass().getName()</code>	Ruft die Art einer Eigenschaft oder Variable ab.	<code>props['p.myProperty'] props['p.myProperty']</code>	<code>java.lang.Boolean _getClass().getName()</code>
<code>getEndpoint()</code>	Ruft den Endpunkt eines RaaS-Berichts bei Verwendung einer SOAP-Anforderung ab.	<code>@{intsys.reportService Alias'})}</code>	<code>customreport2/ Report <tenant_name>/ <username>/ report_name_in_XML_format</code>
<code>getExtrapath()</code>	Ruft den zusätzlichen Pfad eines RaaS-Berichts ab.	<code>intsys.reportService workday-in Report Service Alias>')</code>	<code>customreport2/ your <tenant_name>/ <username>/ <Custom Report Name></code>
<code>indexOf()</code>	Ruft die Position eines bestimmten Zeichens innerhalb einer Eingabezeichenfolge ab.	<code>props['input.document'] = 'FoundationDataCollection/4d196113' props['index.of.separator'] = props['input.document.reference'].indexOf("/")</code>	<code>24</code>
<code>leftPad()</code>	Fügt Zeichen am Anfang einer Eingabezeichenfolge hinzu.	<code>org.apache.commons. 10, '0')</code>	<code>0000001234 leftPad('1234', 10, '0')</code>
<code>listToCommaDelimitedString()</code>	Konvertiert eine Liste von Array-Werten in eine Zeichenfolge und trennt sie durch Kommas.	<code>props[p.workers] = [1234, 7890] props['p.workers'] = lp.getWIDs('Pay Group Members') props['p.workers.ids']</code>	<code>1234, 7890</code>

Methode	Beschreibung	Beispielcode	Beispielausgabe
		<pre> = props['p.workers'] ! = empty ? util.listToCommaDelimString(props['p.workers']) </pre>	
replaceAll()	Ersetzt alle Instanzen angegebener Zeichen innerhalb einer Eingabezeichenfolge.	<pre> props['p.employee.id'] 12345 = '12345pdf_a.pdf' props['p.current.id'] = props['p.employee.id'].replaceAll('[^0-9]', '') </pre>	
substring()	Ruft einen Teil einer Eingabezeichenfolge ab.	<pre> props['p.test.string'] te = 'test' props['p.test.substring'] = props['p.test.string'].substring(0,2) </pre>	
toUpperCase()	Konvertiert eine Zeichenfolge in Großbuchstaben.	<pre> props['p.test.string'] WORKDAY STUDIO = 'workday Studio' props['p.test.string'].toUpperCase() </pre>	
URLEncoder ()	Ruft eine Zeichenfolge ab, die nur gültige URL-Zeichen enthält.	<pre> props['p.report.parameters'] Full_Name=Logan = 'Full_Name=' %20McNeil # 'Logan McNeil' props['REST.URL'] = java.net.URLEncoder.encode(props['p.report.parameters'], 'UTF-8') </pre>	

Studio Integrationsservices

Attribut- und Map-Services erstellen

Vorbereitungen

Erstellen Sie ein Baugruppenprojekt.

Warum und wann dieser Vorgang ausgeführt wird

Mit dem Attribut- und Map-Service können Sie Attribute und Maps für Ihre Integration definieren. Attribute sind konfigurierbare Werte, die Ihre Integration definieren. Sie können folgende Arten von Attributen erstellen:

- Einfach
- Referenz
- Enumeration

Maps stimmen interne Workday-Werte mit den externen Attributwerten ab, die in Ihrer Integration definiert sind. Sie können folgende Arten von Maps erstellen:

- Einfach

- CRF
- Enumeration

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Ziehen Sie per Drag-and-Drop einen Workday-In-Transport aus der Palette in den Assembly-Editor.
3. Klicken Sie auf dem Register Services der Ansicht Eigenschaften für den Workday-In-Transport auf Attribut- und Map-Service erstellen.
4. Geben Sie im Fenster Servicename einen Namen für Ihren Attribut- und Map-Service ein.
5. Auf dem Register Integrationsattribute ist Folgendes zu beachten:

Option	Bezeichnung
Name	Der Name des Attributs. Muss für alle Integrationssysteme eindeutig sein.
Art	Die Definition der Attributart.
Attribut für einfache Art hinzufügen	Fügt eine Attributart hinzu, die Zeichenfolgen als Literalwerte verwendet.
Attribut für Referenzart hinzufügen	Fügt eine Attributart hinzu, die Referenzen auf Workday Class Report Fields (CRFs) und Webservices verwendet.
Attribut für Enumerationsart hinzufügen	Wählen Sie Art bearbeiten, um die Enumeration zu konfigurieren. Sie können: • Verwenden Sie eine Enumerationsreferenz für eine vorhandene Enumerationsdefinition in Ihrem Mandanten, indem Sie den zugehörigen Namen der Art eingeben. • Erstellen Sie eine neue Enumeration, indem Sie auf dem Register Enumerationsdefinition einen eindeutigen Namen der Art eingeben und die zulässigen Werte angeben. Anmerkung: Wenn Sie den Namen der Art einer vorhandenen Definition auf dem Register Enumerationsdefinition eingeben, überschreibt Workday die Definition.

6. Wählen Sie auf dem Register Integrations-Maps die Option Hinzufügen, um eine Map hinzuzufügen.
7. Bei der Konfiguration von Interne Art und Externe Art ist Folgendes zu beachten:

Option	Bezeichnung
Einfache Art	Sie können die folgenden einfachen Arten auswählen: • <i>Boolesch</i> • <i>Datum</i> • <i>Datum/Zeit</i> • <i>Zahl</i> • <i>Text</i>
CRF	Wählen Sie Durchsuchen nach CRF.

Option	Bezeichnung
Enumeration	Konfigurieren Sie Ihre Enumerationsreferenz oder Enumerationsdefinition.

Ergebnisse

Studio speichert den Attribut-Mapping-Service im Workday-In-Transport Ihrer Integration. Sie können über die MSEL auf die Daten des Service zugreifen im sys Ausdruck.

Nächste Maßnahme

Implementieren Sie Ihre Integration.

[relatedlinks](#)

Konzepte

[Concept: Enumerations](#)

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Attribute und Karten in Benefits Integration einrichten](#)

Übermittlungsservices erstellen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Mit dem Übermittlungsservice können Sie Dokumente von Ihrem Workday-Mandanten aus an einen Zielendpunkt übermitteln.

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Blenden Sie in der Palette das Register In Transports ein.
3. Ziehen Sie per Drag-and-Drop einen workday-in-Transport aus der Palette in den Assembly-Editor.
4. Klicken Sie im Assembly-Editor auf den workday-in-Transport.
5. Klicken Sie auf dem Register Services der Ansicht Eigenschaften für den workday-in-Transport auf Übermittlungsservice erstellen.
6. Geben Sie im Fenster Servicename einen Namen für Ihren Übermittlungsservice ein.
7. Klicken Sie auf Speichern.

Ergebnisse

Studio speichert den Übermittlungsservice im workday-in-Transport Ihrer Integration.

Nächste Maßnahme

Stellen Sie Ihre Integration bereit und konfigurieren Sie den Übermittlungsservice in Ihrem Integrationssystem in Workday.

[relatedlinks](#)

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Integrationslieferung einrichten](#)

Listener-Services erstellen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Mit dem Listener-Service kann die Integration Startmeldungen von Drittanbieter-Clients über Ihren Workday-Mandanten empfangen.

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Blenden Sie in der Palette das Register In Transports ein.
3. Ziehen Sie per Drag-and-Drop einen workday-in-Transport aus der Palette in den Assembly-Editor.
4. Klicken Sie im Assembly-Editor auf den workday-in-Transport.
5. Klicken Sie auf dem Register Services der Ansicht Eigenschaften für den workday-in-Transport auf Create listener-service.
6. Geben Sie im Fenster Servicename einen Namen für den Listener-Service ein.
7. Klicken Sie auf Speichern.

Ergebnisse

Studio speichert den Listener-Service im workday-in-Transport der Integration.

Nächste Maßnahme

Stellen Sie die Integration bereit und konfigurieren Sie den Listener-Service für Ihr Integrationssystem in Workday.

Anmerkung: Sie können Integrationen, die Listener-Services verwenden, nur dann bereitstellen, wenn sie keine Startparameter haben.

relatedlinks

Konzepte

[Konzept: Listener-Service-API](#)

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

Abrufservices erstellen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Mit dem Abrufservice kann Ihre Integration Dokumente von Remote-Clients abrufen und in Ihrem Workday-Mandanten speichern.

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Blenden Sie in der Palette das Register In Transports ein.
3. Ziehen Sie per Drag-and-Drop einen workday-in-Transport aus der Palette in den Assembly-Editor.

4. Klicken Sie im Assembly-Editor auf workday-in-Transport.
5. Klicken Sie auf dem Register Services der Ansicht Eigenschaften für den workday-in-Transport auf Create retrieval-service.
6. Geben Sie im Fenster Servicename einen Namen für Ihren Abrufservice ein.
7. Klicken Sie auf Speichern.

Ergebnisse

Studio speichert den Abrufservice im workday-in-Transport Ihrer Integration.

Nächste Maßnahme

Stellen Sie Ihre Integration bereit und konfigurieren Sie den Abrufservice in Ihrem Integrationssystem in Workday.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Integrationsabruf einrichten \(Englisch\)](#).

Sequenzgeneratorservice erstellen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Mit dem Sequenzgeneratorservice können Sie bei jeder Ausführung Ihrer Integration eindeutige, sequenzierte Dateinamen für Dokumente generieren. In Studio konfigurieren Sie den Sequenzgeneratorservice für einen workday-in-Transport. Bevor Sie Ihre Integration starten können, müssen Sie auch den Sequenzgeneratorservice in Ihrem Workday-Mandanten konfigurieren.

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Klicken Sie im Assembly-Editor auf workday-in-Transport.
3. Klicken Sie auf dem Register Services der Ansicht Eigenschaften für den workday-in-Transport auf Create sequence-generator-service.
4. Geben Sie im Fenster Servicename einen Namen für Ihren Sequenzgeneratorservice ein.
5. Klicken Sie in der Ansicht Eigenschaften für den Sequenzgeneratorservice auf Hinzufügen, um einen Sequenzer hinzuzufügen.
6. Geben Sie in das Feld Sequenzer einen Namen für Ihren Sequenzer ein.
7. Klicken Sie in der Ansicht Project Explorer mit der rechten Maustaste auf Ihr Assembly-Projekt und wählen Sie Deploy to Workday aus.
8. Navigieren Sie in der Ansicht Cloud Explorer zu Ihrem bereitgestellten Assembly-Projekt.
9. Klicken Sie mit der rechten Maustaste auf den Integrationsknoten des von Ihnen bereitgestellten Assembly-Projekts.
10. Wählen Sie Workday: Integrationssystem.
11. Auf dem Register Workday ist Folgendes zu beachten:

Option	Bezeichnung
Benutzername	Der Benutzername der Zugangsdaten für Ihren Workday-Mandanten.

Option	Bezeichnung
Kennwort	Das Kennwort der Zugangsdaten für Ihren Mandanten.

12. Suchen Sie im Workday-Mandanten in der Spalte Integrationsservices des Abschnitts Custom Integration Services den Namen Ihres Sequenzgeneratorservice und wählen Sie Mögliche Aktionen und Vorschau.
13. Wählen Sie Service für Integrationssequenzgeneratoren > Konfigurieren von Generatoren für Integrationssequenzen .
14. Geben Sie im Feld Sequenz-ID erhöhen nach den Wert „1“ ein.
15. Geben Sie im Feld Sequenz-ID-Format den Wert ein `file[Seq].csv`.
16. Klicken Sie auf OK.
17. Klicken Sie auf Fertig.

Ergebnisse

Studio speichert den Sequenzgeneratorservice im workday-in-Transport Ihrer Integration. Workday speichert Ihre Einstellungen für den Sequenzgenerator in Ihrem Integrationssystem.

Nächste Maßnahme

Starten Sie Ihre Integration.

relatedlinks

Aufgaben

[Integrationen aus Studio starten](#) auf Seite 345

[Integrationssequenzgeneratoren einrichten](#)

Transaktionsprotokollservices erstellen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Mit dem Transaktionsprotokollservice können Sie die Protokolle für Transaktionsereignisse anzeigen, die Ihre Integration in Ihrem Workday-Mandanten abonniert hat.

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Blenden Sie in der Palette das Register In Transports ein.
3. Ziehen Sie per Drag-and-Drop einen workday-in-Transport aus der Palette in den Assembly-Editor.
4. Klicken Sie im Assembly-Editor auf workday-in-Transport.
5. Klicken Sie auf dem Register Services der Ansicht Eigenschaften für den workday-in-Transport auf Create transaction-log-service.
6. Geben Sie im Fenster Servicename einen Namen für den Transaktionsprotokollservice ein.
7. Klicken Sie auf Speichern.

Ergebnisse

Studio speichert den Transaktionsprotokollservice im workday-in-Transport Ihrer Integration.

Nächste Maßnahme

Stellen Sie Ihre Integration bereit und konfigurieren Sie den Transaktionsprotokollservice in Ihrem Integrationssystem in Workday.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Schritte: Verzeichnisserviceintegration einrichten](#)

Berichtsservices erstellen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Der Berichtsservice gibt Aliase für alle benutzerdefinierten Berichte an, auf die Sie in Ihrer Integration verweisen.

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Blenden Sie in der Palette das Register In Transports ein.
3. Ziehen Sie per Drag-and-Drop einen workday-in-Transport aus der Palette in den Assembly-Editor.
4. Klicken Sie im Assembly-Editor auf den workday-in-Transport.
5. Klicken Sie auf dem Register Services der Ansicht Eigenschaften für Workday-In-Transport auf Berichtsservice erstellen.
6. Geben Sie im Fenster Servicename einen Namen für Ihren Berichtsservice ein.
7. Klicken Sie in der Ansicht Eigenschaften für den Berichtsservice auf Neuen Eintrag hinzufügen, um einen neuen Alias hinzuzufügen. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Alias	Der Alias Ihres benutzerdefinierten Berichts.
Beschreibung	Eine Textbeschreibung Ihres benutzerdefinierten Berichts.
Berichtsreferenz	Der Name Ihres benutzerdefinierten Berichts, wie er in Ihrem Workday-Mandanten angezeigt wird.

8. Klicken Sie auf RaaS Report auswählen. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Umwelt	Die Workday-Umgebung, die Ihren benutzerdefinierten Bericht enthält.
ReportName (Berichtsname)	Der Name Ihres benutzerdefinierten Berichts.

9. Klicken Sie auf Weiter.
10. Wählen Sie Ihren benutzerdefinierten Bericht aus.
11. Klicken Sie auf Beenden.
12. Klicken Sie auf Speichern.

Ergebnisse

Studio speichert den Berichtsservice im workday-in-Transport Ihrer Integration.

Nächste Maßnahme

Stellen Sie die Integration bereit und konfigurieren Sie den Berichtsservice für Ihr Integrationssystem in Workday.

relatedlinks

Konzepte

[Konzept: Benutzerdefinierte Berichte](#)

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

Services für benutzerdefinierte Objekte erstellen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Der Service für benutzerdefinierte Objekte gibt Aliasse für alle benutzerdefinierten Objekte an, auf die Sie in Ihrer Integration verweisen.

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Blenden Sie in der Palette das Register In Transports ein.
3. Ziehen Sie per Drag-and-Drop einen workday-in-Transport aus der Palette in den Assembly-Editor.
4. Klicken Sie im Assembly-Editor auf den workday-in-Transport.
5. Klicken Sie auf dem Register Services der Ansicht Eigenschaften für Workday-In-Transport auf Service für benutzerdefinierte Objekte erstellen.
6. Geben Sie im Fenster Servicename einen Namen für Ihren Service für benutzerdefinierte Objekte ein.
7. Klicken Sie in der Ansicht Eigenschaften für den Service für benutzerdefinierte Objekte auf Neuen Eintrag hinzufügen, um einen neuen Alias hinzuzufügen. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Alias	Der Alias Ihres benutzerdefinierten Objekts.
Beschreibung	Eine Textbeschreibung Ihres benutzerdefinierten Objekts.
Referenz auf benutzerdefiniertes Objekt	Der Name Ihres benutzerdefinierten Objekts, wie er in Ihrem Workday-Mandanten angezeigt wird.

8. Klicken Sie auf Benutzerdefiniertes Objekt auswählen.
9. Geben Sie im Fenster Benutzerdefiniertes Objekt auswählen Ihre Workday-Mandantenumgebung an und navigieren Sie zu Ihrem benutzerdefinierten Objekt.
10. Klicken Sie auf OK.
11. Klicken Sie auf Speichern.

Ergebnisse

Studio speichert den Service für benutzerdefinierte Objekte im workday-in-Transport Ihrer Integration.

Nächste Maßnahme

Stellen Sie Ihre Integration bereit und konfigurieren Sie den Service für benutzerdefinierte Objekte in Ihrem Integrationssystem in Workday.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Benutzerdefinierte Integrationsobjekte einrichten](#)

Dateninitialisierungsservices erstellen**Vorbereitungen**

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Der Dateninitialisierungsservice ermöglicht Ihnen einen schnelleren und effizienteren Zugriff auf Workday-Daten in Ihren Studio-Integrationen.

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Blenden Sie in der Palette das Register In Transports ein.
3. Ziehen Sie per Drag-and-Drop einen workday-in-Transport aus der Palette in den Assembly-Editor.
4. Klicken Sie im Assembly-Editor auf den workday-in-Transport.
5. Klicken Sie auf dem Register Services der Ansicht Eigenschaften für den workday-in-Transport auf Dateninitialisierungsservice erstellen.
6. Geben Sie im Fenster Servicename einen Namen für Ihren Dateninitialisierungsservice ein.
7. Klicken Sie auf Speichern.

Ergebnisse

Studio speichert den Dateninitialisierungsservice im workday-in-Transport Ihrer Integration.

Nächste Maßnahme

Stellen Sie die Integration bereit und konfigurieren Sie den Dateninitialisierungsservice für Ihr Integrationssystem in Workday.

Servicereferenzen erstellen**Vorbereitungen**

Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Mit Servicereferenzen können Sie auf jeden Integrationsservice verweisen, der für einen anderen Workday-In-Transport innerhalb derselben Sammlung konfiguriert ist.

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Blenden Sie in der Palette das Register In Transports ein.
3. Ziehen Sie per Drag-and-Drop einen workday-in-Transport aus der Palette in den Assembly-Editor.
4. Klicken Sie im Assembly-Editor auf den workday-in-Transport.
5. Klicken Sie auf dem Register Services der Ansicht Eigenschaften für Workday-In-Transport auf Servicereferenzen erstellen.
6. Wählen Sie im Fenster Servicereferenz eine Servicereferenz aus.
7. Klicken Sie auf OK.
8. Klicken Sie auf Speichern.

Ergebnisse

Studio speichert die Servicereferenz im Workday-In-Transport Ihrer Integration.

Nächste Maßnahme

Stellen Sie Ihre Integration bereit.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

Referenzen zu benutzerdefinierten Berichten über Transporte „Workday-Out-Rest“ hinzufügen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt, das einen workday-out-rest-Transport enthält.

Warum und wann dieser Vorgang ausgeführt wird

Mit dem Transport workday-out-rest können Sie REST-Anforderungsnachrichten an eine HTTP-URL senden. Sie können eine Referenz zu einem benutzerdefinierten Bericht als zusätzlichen Pfadwert für einen Transport workday-out-rest hinzufügen.

Prozedur

1. Öffnen Sie die Assembly im Assembly-Editor.
2. Klicken Sie im Assembly-Editor auf den workday-out-rest-Transport.
3. Klicken Sie auf dem Register Allgemein der Ansicht Eigenschaften für den workday-out-rest-Transport auf Einen RaaS-Bericht auswählen.
4. Im Fenster Benutzerdefinierten RaaS-Bericht hinzufügen ist Folgendes zu beachten:

Option	Bezeichnung
Umwelt	Die Workday-Umgebung, die Ihren benutzerdefinierten Bericht enthält.
ReportName (Berichtsname)	Der Name Ihres benutzerdefinierten Berichts.

5. Wählen Sie im Fenster Berichte auswählen Ihren benutzerdefinierten Bericht aus.
6. Im Fenster Berichtsservice-Konfiguration ist Folgendes zu beachten:

Option	Bezeichnung
Alias des Berichts	Der Alias Ihres benutzerdefinierten Berichts.

Option	Bezeichnung
Beschreibung	Eine Textbeschreibung Ihres benutzerdefinierten Berichts.

7. Klicken Sie auf Beenden.

Ergebnisse

Studio fügt eine Referenz zu Ihrem benutzerdefinierten Bericht über den workday-out-rest-Transport Ihrer Integration hinzu.

Nächste Maßnahme

Starten Sie Ihre Integration.

relatedlinks

Aufgaben

[Integrationen aus Studio starten](#) auf Seite 345

Referenzen zu benutzerdefinierten Objekten in Transporten „Workday-Out-Rest“ hinzufügen

Vorbereitungen

Erstellen Sie ein Assembly-Projekt, das einen workday-out-rest-Transport enthält.

Warum und wann dieser Vorgang ausgeführt wird

Mit dem Transport workday-out-rest können Sie REST-Anforderungsnachrichten an eine HTTP-URL senden. Sie können eine Referenz zu einem benutzerdefinierten Objekt als zusätzlichen Pfadwert für einen Transport workday-out-rest hinzufügen.

Prozedur

- Öffnen Sie die Assembly im Assembly-Editor.
- Klicken Sie im Assembly-Editor auf den workday-out-rest-Transport.
- Klicken Sie auf dem Register Allgemein der Ansicht Eigenschaften für den workday-out-rest-Transport auf Benutzerdefiniertes Objekt auswählen.
- Im Fenster Benutzerdefiniertes Objekt hinzufügen sind folgende Informationen zu beachten:

Option	Bezeichnung
Umwelt	Die Workday-Umgebung, die Ihr benutzerdefiniertes Objekt enthält.
HTTP-Methoden	Folgende Optionen sind verfügbar: • POST • PUT • GET • DELETE

- Im Fenster Benutzerdefiniertes Objekt auswählen sind folgende Informationen zu beachten:

Option	Bezeichnung
Vorhandenen Alias für benutzerdefiniertes Objekt auswählen	Zeigt eine Liste der vorhandenen benutzerdefinierten Objekt-Aliasnamen an.

Option	Bezeichnung
Neuen Alias für benutzerdefiniertes Objekt erstellen	Ermöglicht das Erstellen eines neuen benutzerdefinierten Objektalias.

Fahren Sie mit Schritt 7 fort, wenn Sie die HTTP-Methode PUT, GET oder DELETE ausgewählt haben.

6. Wenn Sie die HTTP-Methode POST ausgewählt haben, zeigt Studio das Fenster POST-Abfrageparameter an. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
updateIfExists	Aktualisiert ein vorhandenes benutzerdefiniertes Objekt.
bulk	Erstellt oder aktualisiert eine Sammlung benutzerdefinierter Objekte.

7. Wenn Sie die HTTP-Methode PUT, GET oder DELETE ausgewählt haben, wählen Sie im Fenster Custom Object Id Reference Selection eine ID-Referenz für das benutzerdefinierte Objekt aus.
8. Klicken Sie auf Beenden.

Ergebnisse

Studio fügt Ihrer Integration über den workday-out-rest-Transport eine Referenz zu Ihrem benutzerdefinierten Objekt hinzu.

Nächste Maßnahme

Starten Sie Ihre Integration.

relatedlinks

Aufgaben

[Integrationen aus Studio starten](#) auf Seite 345

CLAR-Dateien

Manifestdateien generieren

Warum und wann dieser Vorgang ausgeführt wird

Mit Manifestdateien können Sie Assembly-Sammlungen als CLAR-Dateien exportieren. Sie müssen Manifestdateien in einem Projektordner in einer Assembly-Sammlung speichern. Manifestdateien enthalten die Metadaten Ihrer Assembly-Sammlungen.

Prozedur

1. Wählen Sie in der Studio-Menüleiste Workday- > Lösung > Workday-Manifest generieren aus.
2. Doppelklicken Sie auf den Zielprojektordner für die Manifestdatei.

Ergebnisse

Studio speichert die Manifestdatei im Projektordner.

Nächste Maßnahme

Exportieren Sie die Assembly-Sammlung als CLAR-Datei.

CLAR-Dateien importieren

Vorbereitungen

Speichern Sie eine CLAR-Datei in Ihrem Dateisystem.

Warum und wann dieser Vorgang ausgeführt wird

Sie können Baugruppen zu Ihrem Arbeitsbereich hinzufügen, indem Sie Clar-Dateien importieren.

Prozedur

1. Wählen Sie in der Menüleiste von „Studio“ die Option > Dateiimport...
2. Wählen Sie den Assistenten für den CLA- Dateiimport aus.
3. Klicken Sie auf Durchsuchen und wählen Sie eine CLAR-Datei für den Import aus.
4. Klicken Sie auf Weiter, um die Sammlungen und Projekte in der CLAR-Datei anzuzeigen.
5. Wählen Sie Sammlungen und Projekte, die aus der CLAR-Datei importiert werden sollen.
Studio wählt automatisch jede Sammlung und jedes Projekt in der CLAR-Datei aus. Ändern Sie Ihre Auswahl mithilfe der Kontrollkästchen. Studio zeigt Decorators für Sammlungen und Projekte mit Namenskonflikten an.
6. Bearbeiten Sie alle Namenskonflikte von Projekten.
 - a) Wählen Sie ein Projekt mit einem Namenskonflikt aus.
 - b) Klicken Sie auf Bearbeiten
 - c) Benennen Sie das Projekt im Feld Projektname um.
7. Bearbeiten Sie alle Namenskonflikte von Sammlungen.
 - a) Wählen Sie eine Sammlung mit einem Namenskonflikt aus.
 - b) Klicken Sie auf Bearbeiten
 - c) Benennen Sie die Sammlung im Feld Collection Name um.
8. Klicken Sie auf Beenden.

Ergebnisse

Studio importiert den Inhalt der CLAR-Datei in Ihren Arbeitsbereich.

CLAR-Dateien exportieren

Vorbereitungen

- Erstellen Sie ein Assembly-Projekt.
- Generieren Sie eine Manifest-Datei.

Warum und wann dieser Vorgang ausgeführt wird

Sie können Assemblys aus Ihrem Arbeitsbereich als CLAR-Dateien exportieren. Sie können CLAR-Dateien verwenden, um Ihre Assemblys mit anderen Studio-Integrationsentwicklern zu teilen. Beispiel: Durch Hochladen auf die Seite „Beigesteuerte Lösungen“ in der Workday Community.

Prozedur

1. Wählen Sie in der Menüleiste für „Studio“ die Option Datelexport > ...
2. Wählen Sie den Assistenten für den Export von Cclar-Dateien .
3. Wählen Sie im Fenster CLAR-Dateispezifikation eine Sammlung aus, die als CLAR-Datei exportiert werden soll.

4. Wählen Sie im Fenster Workday-Manifest-Spezifikation die Option Vorhandenes Manifest aus Sammlung verwenden aus.
5. Geben Sie im Fenster Exportziel den Zielort der CLAR-Datei an.
6. Klicken Sie auf Beenden.

Ergebnisse

Studio exportiert die Assemblys als CLAR-Datei.

[relatedlinks](#)

Referenz

[All Contributed Solutions](#)

Konzept: CLAR-Dateien

CLAR-Dateien

Eine CLAR-Datei (Cloud Archive) ist eine Paketdatei, in der Assembly-Sammlungen, -Projekte und -Metadaten gespeichert werden. Mit CLAR-Dateien können Sie Assemblys über Ihren Arbeitsbereich importieren und exportieren. Sie können CLAR-Dateien verwenden, um Ihre Assemblys mit anderen Studio-Integrationsentwicklern zu teilen.

Manifestdateien

Mit Manifestdateien können Sie Assembly-Sammlungen als CLAR-Dateien exportieren. Sie müssen Manifestdateien in einem Projektordner in einer Assembly-Sammlung speichern. Manifestdateien enthalten die Metadaten Ihrer Assembly-Sammlungen.

Beigesteuerte Lösungen

Die Seite „Contributed Solutions“ ist ein Online-Repository, mit dem Sie auf CLAR-Dateien zugreifen und diese teilen können. Sie können die Seite „Contributed Solutions“ in der Workday Community öffnen.

JAXB-Projekte

JAXB-Projekte erstellen

Vorbereitungen

Speichern Sie eine XML-Schemadatei in Ihrem Dateisystem.

Warum und wann dieser Vorgang ausgeführt wird

JAXB ist ein Bindungs-Framework, mit dem Sie Daten zwischen den Formaten XML und Java konvertieren können. In Studio können Sie XML-Schemadateien in Java-Klassen konvertieren, indem Sie ein JAXB-Projekt erstellen. Sobald Sie ein JAXB-Projekt erstellt haben, können Sie seine Java-Klassen in einer Integration implementieren, indem Sie sie als Abhängigkeit in einem Zusammenstellungsprojekt hinzufügen.

Prozedur

1. Wählen Sie File > New > Workday Schema to Java (JAXB) Project.

2. Im Fenster Details zur Projektkonfiguration sind folgende Informationen zu beachten:

Option	Bezeichnung
Projektname	Der Name Ihres Projekts.
Ziellaufzeit	Ziellaufzeitserver für Ihr Projekt. Wird automatisch als <i>Workday-Laufzeit</i> ausgefüllt.
Konfiguration	Die Facettenkonfiguration Ihres Projekts. Wird automatisch als <i>Projekt: Workday-Schema in Java (JAXB)</i> ausgefüllt.

3. Im Fenster Quellordner sind folgende Informationen zu beachten:

Option	Bezeichnung
Generierungsverzeichnis	Der Verzeichnisordner für Ihre von JAXB generierten Java-Klassen. Wird automatisch als <i>jaxb-gen</i> ausgefüllt.
Generierte Ressourcen als abgeleitet markieren	Ihre von JAXB generierten Java-Klassen werden als abgeleitete Ressourcen markiert. Workday aktiviert dieses Kontrollkästchen automatisch.

4. Im Fenster Einstellungen für JAXB-Schemaauswahl und -generierung sind folgende Informationen zu beachten:

Option	Bezeichnung
Schemadateien	Der Speicherort Ihrer XML-Schemadateien. Studio zeigt folgende Optionen an: - Arbeitsbereich - Extern - URL
Externe JAXB-Bindungsdateien (optional)	Der Speicherort aller Bindungsdateien für Ihre XML-Schemas. Studio zeigt folgende Optionen an: - Arbeitsbereich - Extern
Standard-Bindungsdatei für nicht gemappte Schemas generieren	Erstellt generische Bindungsdateien für Ihre XML-Schemas.
Standort in Projekt	Der Verzeichnisordner für generische Bindungsdateien. Wird automatisch als ausgefüllt <i>jaxb-resources/bindings.xjb</i> .
Wählen Sie, wie Schema- und externe JAXB-verbindliche Dateien verwaltet werden sollen	Wählen Sie Im Projekt Links zu externen Schema-/Bindungsdateien erstellen, um in Ihrem neuen JAXB-Projekt Links zu Ihren Schema- und Bindungsdateien zu erstellen. Workday aktiviert diese Option automatisch. Wählen Sie Schema-/Bindungsdateien jetzt in Projekt kopieren, um Ihre Schema- und Bindungsdateien in Ihr neues JAXB-Projekt zu kopieren.
Compiler-Optionen	Studio zeigt folgende Optionen an:

Option	Bezeichnung
	• Zielpaket angeben (überschreibt alle anderen Paketdefinitionen) • Weniger strenge Validierung auf Schemas anwenden • Anmerkungen auf Paketebene unterdrücken • Automatische Lösung von Namenskonflikten

5. Klicken Sie auf Beenden.

Ergebnisse

Studio zeigt das JAXB-Projekt in der Ansicht Projekt-Explorer an.

Nächste Maßnahme

Fügen Sie das JAXB-Projekt als Abhängigkeit in einem Assembly-Projekt hinzu.

JAXB-Projekte als Abhängigkeiten in Assembly-Projekten hinzufügen

Vorbereitungen

- Erstellen Sie ein Assembly-Projekt.
- Erstellen Sie ein JAXB-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

JAXB ist ein Bindungs-Framework, mit dem Sie Daten zwischen den Formaten XML und Java konvertieren können. In Studio können Sie JAXB-Java-Klassen in Ihren Integrationen implementieren, indem Sie JAXB-Projekte als Abhängigkeiten zu Assembly-Projekten hinzufügen.

Prozedur

1. Klicken Sie in der Ansicht Projekt-Explorer mit der rechten Maustaste auf ein Baugruppenprojekt.
2. Wählen Sie Pfad erstellen > Build-Pfad konfigurieren...
3. Wählen Sie im Fenster Java Build Path das Register Projects .
4. Klicken Sie auf Hinzufügen
5. Wählen Sie ein JAXB-Projekt aus, das Sie zu Ihrem Assembly-Projekt hinzufügen möchten.
6. Klicken Sie auf OK.
7. Klicken Sie auf Anwenden und schließen.

Ergebnisse

Studio fügt das JAXB-Projekt als Abhängigkeit zum Assembly-Projekt hinzu. Sie können nun Ihre JAXB-Java-Klassen in Integrationen implementieren, die in diesem Assembly-Projekt gespeichert sind.

Nächste Maßnahme

Stellen Sie Ihre Integration bereit.

Konzept: JAXB-Unterstützung in Studio

JAXB ist ein Bindungs-Framework, mit dem Sie Daten zwischen den Formaten XML und Java konvertieren können.

JAXB-Projekte

In Studio können Sie XML-Schemadateien in Java-Klassen konvertieren, indem Sie ein JAXB-Projekt erstellen. Sobald Sie ein JAXB-Projekt erstellt haben, können Sie dessen Java-Klassen in einer Integration implementieren, indem Sie das Projekt als Abhängigkeit zu einem Assembly-Projekt hinzufügen.

JAXB-Mediationsschritte

In Ihren Assembly-Projekten können Sie JAXB in der Ansicht Eigenschaften der folgenden Mediationsschritte aufrufen:

- java-to-xml
- xml-to-java

Studio Ivy-Plugin

Ivy-Plugin konfigurieren

Warum und wann dieser Vorgang ausgeführt wird

Workday Studio unterstützt ein Apache Ivy™-Plugin, mit dem Sie Projektabhängigkeiten verwalten können. Sie können das Ivy-Plugin als Einstellung in Studio konfigurieren.

Prozedur

1. Klicken Sie auf **Fenstereinstellungen > für > Workday > Ivy**.
2. Im Fenster **Configure Ivy Engine** sind folgende Informationen zu beachten:

Option	Bezeichnung
URL für Ivy-Einstellungen	Die URL für Ihre Datei „ivysettings.xml“.
Ivy-Eigenschaften	Die URL für Ihre Datei „ivyproperties.xml“.
Datei ivy.properties aus Hauptverzeichnis laden, falls vorhanden	Lädt die Datei „ivy.properties“ automatisch aus Ihrem Hauptverzeichnis.

Ergebnisse

Studio wendet Ihre Ivy-Plugin-Einstellungen auf Ihre Assembly-Projekte an.

Workday Ivy-Arten zu Workday Studio-Projekten hinzufügen

Vorbereitungen

- Konfigurieren Sie das Plugin „Apache Ivy™“ für Workday Studio.
- Erstellen Sie ein Assembly-Projekt.

Warum und wann dieser Vorgang ausgeführt wird

Workday Studio unterstützt das Plug-in „Apache Ivy“, mit dem Sie Projektabhängigkeiten verwalten können, indem Sie Projekte mit der Workday Ivy-Art konfigurieren.

Prozedur

1. Klicken Sie in der Ansicht Projekt-Explorer mit der rechten Maustaste auf ein Baugruppenprojekt.
2. Wählen Sie das Hinzufügen von Workday Ivoryte > konfigurieren.
3. Blenden Sie den Baugruppenprojektordner ein.
4. Öffnen Sie die `ivy.xml` Datei.
5. Blenden Sie den Info- Knoten ein. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Modul	Der Modulname.
Fügen Sie Ihren Inhalt hier hinzu	Fügen Sie ein übergeordnetes Element ein <dependencies> Element, das ein oder mehrere untergeordnete Elemente einschließt <dependency> Elemente

Ergebnisse

Studio fügt im Zusammenhang mit dem Projekt eine Workday Ivory-Art hinzu.

Abhängigkeiten von Projekten mit Ivy-Definitionen auflösen

Vorbereitungen

- Konfigurieren Sie das Plugin „Apache Ivy™“ für Workday Studio.
- Konfigurieren Sie ein Assembly-Projekt mit der Workday Ivy-Art.

Warum und wann dieser Vorgang ausgeführt wird

Workday Studio unterstützt das Plug-in „Apache Ivy“, mit dem Sie Projektabhängigkeiten verwalten können, indem Sie Projekte mit der Workday Ivy-Art konfigurieren. Wenn für Ihr Projekt ein Build-Fehler im Zusammenhang mit fehlenden Abhängigkeiten angezeigt wird, können Sie diese Abhängigkeitsprobleme beheben.

Prozedur

1. Klicken Sie in der Ansicht Project Explorer mit der rechten Maustaste auf ein Assembly-Projekt mit der Workday Ivy-Art.
2. Wählen Sie Lösen.

Ergebnisse

In der Ansicht Konsole wird ein Lösungsbericht angezeigt.

Abhängige JARs zu Projekten mit Ivy-Arten hinzufügen

Vorbereitungen

- Konfigurieren Sie das Plugin „Apache Ivy™“ für Workday Studio.
- Konfigurieren Sie ein Assembly-Projekt mit der Workday Ivy-Art.

Warum und wann dieser Vorgang ausgeführt wird

Workday Studio unterstützt ein Plugin für Workday Ivy, mit dem Sie Projektabhängigkeiten verwalten können, indem Sie Projekte mit der Art „Workday Ivory“ konfigurieren. Wenn ein Studio-Projekt einen

Ivony mit Elementen von Artenabhängigkeiten enthält `ivy.xml`. In der `-Datei` können Sie das Projekt so konfigurieren, dass es alle abhängigen JAR-Dateien enthält.

Prozedur

1. Klicken Sie in der Ansicht Projekt-Explorer mit der rechten Maustaste auf ein Baugruppenprojekt mit der Art Workday Ivy.
2. Wählen Sie Eigenschaften.
3. Klicken Sie im Fenster Java Build Path auf Bibliotheken.
4. Klicken Sie auf Ivy-Workday IDE.
5. Klicken Sie auf JARs hinzufügen.
6. Wählen Sie im Fenster JAR-Auswahl die JAR-Dateien aus, die Sie zu Ihrem Projekt hinzufügen möchten, und klicken Sie auf OK.
7. Klicken Sie im Fenster Java Build Path auf Anwenden und schließen.

Ergebnisse

Studio fügt die abhängigen JAR-Dateien zu Ihrem Projekt hinzu.

Nächste Maßnahme

Stellen Sie Ihr Projekt bereit.

Workday Ivy-Arten aus Studio-Projekten entfernen

Vorbereitungen

- Konfigurieren Sie das Plugin „Apache Ivy™“ für Workday Studio.
- Konfigurieren Sie ein Assembly-Projekt mit der Workday Ivy-Art.

Warum und wann dieser Vorgang ausgeführt wird

Workday Studio unterstützt das Plug-in „Apache Ivy“, mit dem Sie Projektabhängigkeiten verwalten können, indem Sie Projekte mit der Workday Ivy-Art konfigurieren. Sie können die Art „Workday Ivy“ nach Bedarf zu Projekten hinzufügen und entfernen.

Prozedur

1. Wählen Sie in der Ansicht "Projekt-Explorer" einen Baugruppenprojektordner.
2. Wählen Sie Workday > Efeu > Workday Efeu entfernen.

Ergebnisse

Studio entfernt die Workday-Efeu-Art aus Ihrem Projekt.

Nächste Maßnahme

Stellen Sie Ihr Projekt bereit.

Integrations-Deployment

Eine Verbindung hinzufügen

Warum und wann dieser Vorgang ausgeführt wird

In Studio können Sie Verbindungen zu Workday erstellen und verwalten. Sie können diese Verbindungen beim Bereitstellen und Starten von Assemblys verwenden.

Prozedur

1. Wählen Sie Fenster > Systemeinstellungen > Workday > Verbindungen.
2. Klicken Sie auf Hinzufügen
3. Im Fenster Verbindung hinzufügen sind folgende Informationen zu beachten:

Option	Bezeichnung
Name	Der Name der Verbindung.
URL	Die URL der Verbindung.
Gruppe	Fügt die Verbindung zu einer Gruppe hinzu.

4. Geben Sie im Abschnitt Zugangsdaten den Mandanten an und wählen Sie als Authentifizierungsmethode Basisauthentifizierung oder OAuth 2.0.

Anmerkung: Um die OAuth 2.0-Autorisierung verwenden zu können, müssen Sie zuerst einen API-Client registrieren. Benutzer dieses Clients, die für die Multi-Faktor-Authentifizierung von Workday eingeschrieben sind, müssen eine zweite Authentifizierungsmethode verwenden, wenn sie eine Verbindung zu Workday aus Studio herstellen.

Option	Bezeichnung
Basisauthentifizierung	Geben Sie Benutzernamen und das Kennwort für die Verbindung an.
OAuth 2.0	Geben Sie die folgenden Details für die Verbindung an: • Zuteilungsart • Client-ID • Autorisierungsendpunkt • Token-Endpunkt • Zugriffs-Token

5. Klicken Sie auf Verbindung testen.
Studio bestätigt, ob es auf die Verbindung zugreifen kann.

Ergebnisse

Die Verbindung wird auf der Systemeinstellungsseite Verbindungen angezeigt.

Nächste Maßnahme

Erstellen Sie eine Assembly.

relatedlinks

Aufgaben

[Multi-Faktor-Authentifizierung in Workday Studio aktivieren](#) auf Seite 342

Eine Gruppe von Verbindungen hinzufügen

Warum und wann dieser Vorgang ausgeführt wird

In Studio können Sie Verbindungen zu Workday erstellen und verwalten. Sie können diese Verbindungen in Gruppen organisieren. Sie können diese Verbindungen beim Bereitstellen und Starten von Baugruppen verwenden.

Prozedur

1. Wählen Sie Windows > Einstellungen > für Workday > Verbindungen.
2. Klicken Sie auf Gruppe hinzufügen...
3. Im Fenster Gruppe hinzufügen ist Folgendes zu beachten:

Option	Bezeichnung
Name	Der Name der Verbindungsgruppe.
Gefiltert	Filtert die Verbindungsgruppe aus der Ansicht Cloud Explorer und der Seite Deploy to Workday.

Ergebnisse

Die Verbindungsgruppe wird auf der Systemeinstellungsseite Verbindungen angezeigt.

Nächste Maßnahme

Erstellen Sie eine Assembly.

Eine Verbindung bearbeiten

Vorbereitungen

Fügen Sie eine Verbindung zu Workday hinzu.

Warum und wann dieser Vorgang ausgeführt wird

In Studio können Sie Verbindungen zu Workday erstellen und verwalten. Sie können diese Verbindungen beim Bereitstellen und Starten von Baugruppen verwenden.

Prozedur

1. Wählen Sie Windows > für Workday > Einstellungen > Verbindungen.
2. Wählen Sie eine Verbindung aus.
3. Klicken Sie auf Umgebungs-URL bearbeiten. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Host	Die Host-ID der Verbindung.

Option	Bezeichnung
URL	Die URL der Verbindung.

Ergebnisse

Die Verbindung wird auf der Systemeinstellungsseite Verbindungen angezeigt.

Nächste Maßnahme

Erstellen Sie eine Assembly.

Eine Gruppe von Verbindungen bearbeiten

Vorbereitungen

Erstellen Sie eine Gruppe von Verbindungen zu Workday.

Warum und wann dieser Vorgang ausgeführt wird

In Studio können Sie Verbindungen zu Workday erstellen und verwalten. Sie können diese Verbindungen in Gruppen organisieren. Sie können diese Verbindungen beim Bereitstellen und Starten von Baugruppen verwenden.

Prozedur

1. Wählen Sie Windows > für Workday > Einstellungen > Verbindungen.
2. Wählen Sie eine Kontaktgruppe aus.
3. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Name	Der Name der Verbindungsgruppe.
Gefiltert	Filtert die Verbindungsgruppe aus der Ansicht Cloud Explorer und der Seite Deploy to Workday.

Ergebnisse

Die Verbindungsgruppe wird auf der Systemeinstellungsseite Verbindungen angezeigt.

Nächste Maßnahme

Erstellen Sie eine Assembly.

Verbindungen entfernen

Vorbereitungen

Fügen Sie eine Verbindung zu Workday hinzu.

Warum und wann dieser Vorgang ausgeführt wird

In Studio können Sie Verbindungen zu Workday erstellen und verwalten. Sie können diese Verbindungen beim Bereitstellen und Starten von Baugruppen verwenden.

Prozedur

1. Wählen Sie Windows > für Workday > Einstellungen > Verbindungen.
2. Wählen Sie eine Verbindung aus.
3. Klicken Sie auf Entfernen

Ergebnisse

Studio entfernt die Verbindung aus der Systemeinstellungsseite Verbindungen.

Nächste Maßnahme

Erstellen Sie eine Assembly.

Verbindungen importieren

Vorbereitungen

Öffnen Sie eine XML-Datei mit Details zur Workday-Verbindung.

Warum und wann dieser Vorgang ausgeführt wird

In Studio können Sie Verbindungen zu Workday erstellen und verwalten. Sie können diese Verbindungen beim Bereitstellen und Starten von Baugruppen verwenden.

Prozedur

1. Wählen Sie Windows > für Workday > Einstellungen > Verbindungen.
2. Klicken Sie auf Import...
3. Klicken Sie auf Durchsuchen
4. Wählen Sie eine XML-Datei für eine Workday-Verbindung aus.

Ergebnisse

Studio zeigt die Verbindung auf der Systemeinstellungsseite Verbindungen an.

Nächste Maßnahme

Erstellen Sie eine Assembly.

Verbindungen exportieren

Vorbereitungen

Fügen Sie eine Verbindung zu Workday hinzu.

Warum und wann dieser Vorgang ausgeführt wird

In Studio können Sie Verbindungen zu Workday erstellen und verwalten. Sie können diese Verbindungen beim Bereitstellen und Starten von Baugruppen verwenden.

Prozedur

1. Wählen Sie Windows > für Workday > Einstellungen > Verbindungen.
2. Klicken Sie auf Exportieren...

3. Wählen Sie eine Verbindung aus.
4. Klicken Sie auf Durchsuchen ...
5. Wählen Sie einen Zielort für die XML-Datei der Workday-Verbindung aus.

Ergebnisse

Studio exportiert die Verbindung zum Zielort.

Nächste Maßnahme

Erstellen Sie eine Assembly.

Verbindung zu Workday herstellen

Vorbereitungen

Fügen Sie eine Verbindung zu Workday hinzu.

Warum und wann dieser Vorgang ausgeführt wird

In Studio können Sie Verbindungen zu Workday erstellen und verwalten. Sie können diese Verbindungen beim Bereitstellen und Starten von Baugruppen verwenden.

Prozedur

1. Wählen Sie in den Einstellungen für Studio die Option Workday > -Verbindungen.
2. Wählen Sie eine Verbindung aus. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Mandant	Geben Sie den Namen Ihres Mandanten an.
Benutzername	Geben Sie den Benutzernamen der Zugangsdaten für Ihren Mandanten ein.
Kennwort	Geben Sie das Kennwort der Zugangsdaten für Ihren Mandanten ein.

3. Klicken Sie auf Verbindung testen.
Studio bestätigt, ob es Zugriff auf Ihren Mandanten hat.
4. Klicken Sie auf Anwenden und schließen.

Ergebnisse

Studio stellt eine Verbindung zu Ihrem Workday-Mandanten her.

Nächste Maßnahme

Stellen Sie eine Assembly in Workday bereit.

Integrationen in Workday bereitstellen

Vorbereitungen

- Erstellen Sie ein Assembly-Projekt.

- Sicherheit: Domänen *Integration Build* und *Integration Configure* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

In Studio können Sie Ihre Integrationen in Workday bereitstellen. Sobald Sie eine Integration bereitgestellt haben, können Sie sie in Ihrem Workday-Mandanten starten.

Prozedur

1. Klicken Sie in der Ansicht Project Explorer mit der rechten Maustaste auf ein Assembly-Projekt.
2. Wählen Sie Deploy to Workday aus.
3. Im Fenster Deploy to Workday ist Folgendes zu beachten:

Option	Bezeichnung
Workday-Umgebungen	Wählen Sie eine Workday-Zielumgebung für die Bereitstellung Ihrer Integration aus.
Sammlung wird bereitgestellt	Wählen Sie eine Sammlung aus, die in Workday bereitgestellt werden soll.

4. Klicken Sie auf Beenden.

Ergebnisse

Studio stellt Ihre Integration in Workday bereit.

Nächste Maßnahme

Starten Sie Ihre Integration.

Bereitgestellte Integrationen in Workday anzeigen

Vorbereitungen

- Stellen Sie eine Integration in Workday bereit.
- Sicherheit: Domänen *Integration Build* und *Integration Configure* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

In Studio können Sie Ihre Integrationen in Workday bereitstellen. Sie können Ihre bereitgestellten Integrationen anzeigen, wenn Sie sich bei Ihrem Workday-Mandanten anmelden.

Prozedur

1. Wählen Sie in der Ansicht Cloud Explorer einen Cloud-Serverknoten aus und blenden Sie ihn ein.
2. Wählen Sie einen Sammlungsknoten aus und blenden Sie ihn ein.
3. Klicken Sie mit der rechten Maustaste auf einen Integrationsknoten.
4. Wählen Sie Workday: Integrationssystem.
5. Melden Sie sich auf dem Register Workday bei Ihrem Workday-Mandanten an. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Benutzername	Geben Sie den Benutzernamen der Zugangsdaten für Ihren Workday-Mandanten ein.

Option	Bezeichnung
Kennwort	Geben Sie das Kennwort der Zugangsdaten für Ihren Mandanten ein.

Ergebnisse

Ihre bereitgestellte Integration wird auf dem Register Workday in Studio angezeigt.

Nächste Maßnahme

Starten Sie Ihre Integration.

Studio für die Verwendung eines HTTPS-Proxyservers konfigurieren

Warum und wann dieser Vorgang ausgeführt wird

Wenn Sie Workday Studio über einen anderen Port als 443 mit einer Trainings- oder Testumgebung verbinden müssen und die Firewall Ihrer Organisation dies nicht standardmäßig zulässt, können Sie möglicherweise eine Verbindung über einen Proxyserver in Ihrem Netzwerk herstellen.

Prozedur

1. Wählen Sie Windows > -Einstellungen > Allgemein - > Netzwerkverbindungen.
2. Wählen Sie im Dropdown-Menü für Aktive Anbieter die *Option Manuell*.

Anmerkung: Wenn Sie *Manuell* auswählen, werden die Kontrollkästchen für alle drei Schemas in der Tabelle mit Proxy-Einträgen , HTTP, HTTPS und Socks, aktiviert. Sie können diese ausgewählten Werte nicht einzeln deaktivieren.

3. Geben Sie neben dem HTTPS -Schema eine Host -IP-Adresse und eine Portnummer für den Proxyserver ein.
4. Klicken Sie auf Anwenden und schließen.

Konzept: Studio-Verbindungen

In Studio können Sie Verbindungen zu Workday erstellen und verwalten. Sie können diese Verbindungen beim Bereitstellen und Starten von Assemblies verwenden.

Anmerkung: Alle Verbindungsnamen müssen eindeutig sein. Studio stellt Sandbox und Production als Standardverbindungen bereit. Sie können diese Verbindungen nicht umbenennen. Aus Sicherheitsgründen speichert Studio keine Details zu Benutzernamen und Kennwort in Ihrem Arbeitsbereich, nachdem Sie die Studio-Sitzung verlassen haben.

Konzept: Cloud Explorer

In der Cloud-Explorer-Ansicht werden Ihre Cloud-Server-Verbindungen angezeigt. Sie können jeden Cloud-Server-Knoten einblenden, um seine Sammlungen anzuzeigen. Sie können jeden Sammlungsknoten einblenden, um Integrationen anzuzeigen, die zum Start verfügbar sind.

Anmerkung: In der Cloud-Explorer-Ansicht werden maximal 999 Sammlungen angezeigt.

Die Cloud-Explorer-Ansicht verwendet dieselben Einstellungen für die Datumsformatierung wie Ihr lokales Betriebssystem.

Referenz: Verbindungsfehler

Verbindungsfehler	Anmerkungen
Ihre Benutzerzugangsdaten konnten nicht authentifiziert werden.	Prüfen Sie Ihre Zugangsdaten und versuchen Sie es erneut. Beachten Sie, dass bei Mandantennamen die Groß-/Kleinschreibung beachtet werden muss.
Ihr Mandant ist nicht verfügbar.	Verwenden Sie einen Webbrowser, um zu prüfen, ob Ihr Mandant online ist. Geben Sie die Mandanten-URL in folgendem Format ein: <pre>https://host-name/tenant-name/ccx/cc-cloud-repo/collections</pre> Beispiel: <pre>https://wd5.myworkday.com/gms/ccx/cc-cloud-repo/collections</pre> Verwenden Sie dieselben Zugangsdaten wie in Studio, kombinieren Sie jedoch Ihren Benutzer- und Mandantennamen mit einem @-Zeichen. Beispiel: <pre>lmcneil@gms</pre> Wenn die Verbindung erfolgreich ist, zeigt der Browser alle Studio-Integrationssammlungen an, die in Ihrem Mandanten bereitgestellt sind. Prüfen Sie als Nächstes die Systemeinstellungen der Verbindung Ihres Browsers, um festzustellen, ob eine direkte Verbindung hergestellt wird oder die Verbindung über einen Proxy-Server erfolgt. Wenn die Verbindung für die Verwendung eines Proxyservers konfiguriert ist, müssen Sie Studio so konfigurieren, dass es auch diesen Proxy verwendet.
Berechtigung zum Deployment abgelehnt.	Um Integrationen in Workday bereitzustellen, müssen Sie Mitglied einer Sicherheitsgruppe mit der Berechtigung <i>Ändern</i> in der Domäne <i>Integration Build</i> sein. Wenden Sie sich an Ihren Workday-Systemadministrator.
Berechtigung zum Starten abgelehnt.	Um Integrationen aus Studio zu starten, müssen Sie Mitglied einer Sicherheitsgruppe mit der Berechtigung <i>Put</i> in einer der folgenden Domänen sein: • <i>Integration Event</i> • <i>Integration Build</i> • <i>Integration Debug</i> Wenden Sie sich an Ihren Workday-Systemadministrator.

Verbindungsfehler	Anmerkungen
Berechtigung zum Deployment abgelehnt (Integration enthält einen Liefer- oder Abrufservice).	Um eine Integration bereitzustellen, die einen Liefer oder Abrufservice enthält, benötigen Sie eine zusätzliche Berechtigung. Sie müssen Mitglied einer Sicherheitsgruppe sein, die über die Berechtigung <i>Put</i> in einer der folgenden Domänen verfügt • <i>Business Process Administration</i> • <i>Manage: Business Process Definitions</i> • <i>iLoad-Web Services</i> Wenden Sie sich an Ihren Workday-Systemadministrator.
Zugriff auf „Meine Berichte“ verweigert.	Workday steuert den Zugriff auf Integrationsdokumente unter „Meine Berichte“ mithilfe konfigurierbarer Sicherheit. Das Workday-Konto, mit dem Sie auf das Integrationsdokument zugegriffen haben, hat nicht die erforderliche Berechtigung. Wenden Sie sich an Ihren Workday-Systemadministrator, um die erforderliche Berechtigung anzufordern.
SOAP-Fehler: Ein Validierungsfehler ist aufgetreten.	Ihre SOAP-Anforderung für den Webservice-Vorgang ist ungültig. Vergewissern Sie sich, dass Ihre Anforderung gültige Datenelemente und Werte enthält. Verwenden Sie den Assistenten Neuste Workday Webservices hinzufügen im Schema Explorer , um gültige WSDLs in Ihre Workday Studio-Umgebung zu importieren. Verwenden Sie dann die Menüs SOAP Literal erstellen und kopieren oder SOAP-verpackte Anforderung im Webservice-Tester öffnen, um eine gültige SOAP-Anforderung zu erstellen.
Ihre IP-Adresse konnte nicht authentifiziert werden.	Ihre IP-Adresse ist nicht für den Zugriff auf die Workday-Umgebung autorisiert. Wenden Sie sich an den Systemadministrator.

Multi-Faktor-Authentifizierung in Workday Studio

Multi-Faktor-Authentifizierung in Workday Studio aktivieren

Warum und wann dieser Vorgang ausgeführt wird

Sie können Workday Studio so konfigurieren, dass die OAuth 2.0-Autorisierung verwendet wird, wenn eine Verbindung zu einem Mandanten hergestellt wird. Dazu müssen Sie einen API-Client registrieren. Benutzer des Clients, die für die Multi-Faktor-Authentifizierung von Workday in einem Mandanten eingeschrieben sind, müssen eine zweite Authentifizierungsmethode verwenden, wenn sie eine Verbindung zu diesem Mandanten aus Studio herstellen. Sie können keine einfache Authentifizierung verwenden.

Prozedur

1. Richten Sie die Multi-Faktor-Authentifizierung in Ihrem Mandanten ein und geben Sie an, welche Konten für die Verwendung erforderlich sind.
 Siehe [Schritte: Multi-Faktor-Authentifizierung mit der Authentifikator-App einrichten](#).
 Siehe [Schritte: Multi-Faktor-Authentifizierung mit Duo-Sicherheit einrichten](#).
 Siehe [Schritte: Multi-Faktor-Authentifizierung mit einem per E-Mail gesendeten Einmal-Passcode einrichten](#).
 Siehe [Schritte: Multi-Faktor-Authentifizierung mit einem per SMS gesendeten Einmal-Passcode einrichten](#).
2. Registrieren Sie einen OAuth 2.0-fähigen API-Client.
 Konfigurieren Sie folgende Einstellungen:

Option	Wert
Kundenzuteilungsart	Code zur Autorisierungsgewährung
Unterstützung von Nachweisschlüssel für Codeaustausch (PKCE, Proof Key for Code Exchange)	Aktiviert
Art des Zugriffs-Tokens	Inhaber
Umleitungs-URI	Geben Sie eine gültige HTTPS-URL ein.
Umfang	Wählen Sie in der Liste „Benutzerdefinierte Objekte“ den funktionalen Bereich für jeden Webservice aus, auf den Sie zugreifen. Wählen Sie außerdem die folgenden funktionalen Bereiche aus: • Implementierung • Integration • System • Nicht konfigurierbarer Mandant Anmerkung: Die Authentifizierung schlägt fehl, wenn Sie einen relevanten Webservice weglassen.
Workday-eigenen Bereich einschließen	Aktiviert

Anmerkung: Aktivieren Sie nicht das Kontrollkästchen Ablauf des 60-minütigen Zugriffs-Tokens erzwingen. Dadurch wird die OAuth 2.0-Authentifizierung nach diesem Zeitraum deaktiviert. Die OAuth 2.0-Authentifizierung unterliegt außerdem den normalen Workday-Sitzungs-Timeouts, die mandantenweit oder für einzelne Konten konfiguriert sind.

Siehe [API-Clients registrieren](#).

3. Kopieren Sie die Client-ID, den Autorisierungsendpunkt und den Token-Endpunkt, die in der Aufgabe Register API Client bereitgestellt wurden.
4. Navigieren Sie im Menü "Systemeinstellungen für Workday Studio" zu "Workday > -Verbindungen".
5. Fügen Sie eine Mandantenverbindung hinzu und wählen Sie das Register OAuth 2.0-Zugangsdaten.
6. Fügen Sie die Client-ID, den Autorisierungsendpunkt und den Token-Endpunkt ein.
7. Klicken Sie auf Zugriffs-Token anfordern.
 Wenn Sie eine Meldung „Interner Serverfehler“ erhalten, ist die wahrscheinlichste Ursache eine falsche Client-ID.

8. Geben Sie einen Benutzernamen und ein Kennwort für den Mandanten ein.
Wenn Sie für Workday Multi-Faktor-Authentifizierung eingeschrieben sind, werden Sie aufgefordert, eine zweite Authentifizierungsmethode anzugeben.
9. Testen Sie die Verbindung.
10. Klicken Sie auf Anwenden und schließen.

relatedlinks

Referenz

[2021R2 What's New Post: Multifactor Authentication in Workday Studio](#)

Integrationskonfiguration

Konto des Integrationssystembenutzers einer Integration zuweisen

Vorbereitungen

- Stellen Sie eine Integration in Workday bereit.
- Sicherheit: Domäne *Integration Security* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Workday Studio-Integrationen erfordern ein Integrationssystembenutzer-Konto (ISU-Konto) für die Authentifizierung und Webservice-Aufrufe.

Prozedur

1. Wählen Sie in der Ansicht Cloud Explorer einen Cloud-Serverknoten aus und blenden Sie ihn ein.
2. Wählen Sie einen Sammlungsknoten aus und blenden Sie ihn ein.
3. Klicken Sie mit der rechten Maustaste auf einen Integrationsknoten.
4. Wählen Sie Integrationssystembenutzer zuweisen.
5. Klicken Sie auf Integrationssystembenutzer.
6. Wählen Sie im rechten Fensterbereich einen ISU aus.

Ergebnisse

Sie können Ihre Integration als ISU starten.

Nächste Maßnahme

Starten Sie Ihre Integration.

Konzept: Sicherheit des Integrationssystems

In Workday können Sie Sicherheitsgruppen beitreten, mit denen Sie bestimmte Aufgaben in Ihrem Mandanten ausführen können. Wenn Sie in Studio eine Verbindung zu Ihrem Workday-Mandanten herstellen, werden im Fenster Verbindungen testen drei Sicherheitsgruppen angezeigt:

- Bereitstellen
- Starten
- Geschäftsprozess

In Studio müssen Sie mindestens Mitglied von mindestens einer dieser Sicherheitsgruppen sein, bevor Sie eine Integration ausführen können.

relatedlinks

Konzepte

[Konzept: Konfigurierbare Sicherheit](#)

Integration wird gestartet

Integrationen aus Studio starten

Vorbereitungen

- Stellen Sie eine Integration in Workday bereit.
- Sicherheit: Domäne *Integration Event* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Mit Studio können Sie Integrationen in Ihrem Workday-Mandanten starten.

Prozedur

1. Wählen Sie in der Ansicht Cloud Explorer einen Cloud-Serverknoten aus und blenden Sie ihn ein.
2. Wählen Sie einen Sammlungsknoten aus und blenden Sie ihn ein.
3. Klicken Sie mit der rechten Maustaste auf einen Integrationsknoten.
4. Wählen Sie Integration starten aus.
5. Klicken Sie auf Starten.

Ergebnisse

Ihre Integration wird in Ihrem Workday-Mandanten gestartet. In Studio zeigt das Register Prozessmonitor den Status Ihrer Integration an.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

Start-URLs kopieren

Vorbereitungen

Stellen Sie eine Integration in Workday bereit.

Warum und wann dieser Vorgang ausgeführt wird

Mit Studio können Sie die Start-URLs von Integrationen in Ihren Workday-Mandanten kopieren. Sie können diese URLs verwenden, um Integrationen von REST-basierten Drittanbieter-Clients mit Standardauthentifizierungsberechtigungen zu starten.

Prozedur

1. Wählen Sie in der Ansicht Cloud Explorer einen Cloud-Serverknoten aus und blenden Sie ihn ein.

- 2. Wählen Sie einen Sammlungsknoten aus und blenden Sie ihn ein.
- 3. Klicken Sie mit der rechten Maustaste auf einen Integrationsknoten.
- 4. Wählen Sie Start-URL kopieren aus.
- 5. Markieren Sie im Fenster Start-URL die Start-URL und drücken Sie Strg + C.

Ergebnisse

Studio kopiert die Start-URL Ihrer Integration in Ihre Zwischenablage.

Nächste Maßnahme

Verwenden Sie die Start-URL in REST-Anforderungen von REST-basierten Drittanbieter-Clients mit Standardauthentifizierungsberechtigungen.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

Startkonfigurationen erstellen

Vorbereitungen

- Stellen Sie eine Integration in Workday bereit.
- Sicherheit: Domäne *Integration Event* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Mit Studio können Sie Integrationen in Ihrem Workday-Mandanten starten. Sie können Startkonfigurationen für Ihre Integrationen erstellen.

Prozedur

- 1. Wählen Sie Laufkonfigurationen > ausführen.
- 2. Klicken Sie im linken Navigationsbereich mit der rechten Maustaste auf Integration.
- 3. Klicken Sie auf Neu.
- 4. Im rechten Navigationsbereich sind folgende Informationen zu beachten:

Option	Bezeichnung
Name	Der Name Ihrer Startkonfiguration.
Starten in	Die Workday-Zielumgebung für den Start Ihrer Integration.
Sammlung	Die Sammlung, die in Ihrer Integration gestartet werden soll.
Projekt	Das Projekt, das in Ihrer Integration gestartet werden soll.
Workday-In	Der Name des Workday-In-Transports in Ihrer Integration.
Startparameter	Die Startparameter Ihrer Integration.

- 5. Klicken Sie auf Anwenden.
- 6. Klicken Sie auf Ausführen.

Ergebnisse

Studio führt Ihre Integration mit den Details Ihrer neuen Startkonfiguration aus.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

Startkonfigurationen bearbeiten

Vorbereitungen

- Stellen Sie eine Integration in Workday bereit.
- Sicherheit: Domäne *Integration Event* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Mit Studio können Sie Integrationen in Ihrem Workday-Mandanten starten. Sie können die Startkonfigurationen Ihrer Integrationen bearbeiten.

Prozedur

1. Wählen Sie Laufkonfigurationen > ausführen.
2. Blenden Sie im linken Navigationsbereich den Knoten Integration ein.
3. Wählen Sie einen Startkonfigurationsknoten.
4. Im rechten Navigationsbereich sind folgende Informationen zu beachten:

Option	Bezeichnung
Name	Der Name Ihrer Startkonfiguration.
Starten in	Die Workday-Zielumgebung für den Start Ihrer Integration.
Sammlung	Die Sammlung, die in Ihrer Integration gestartet werden soll.
Projekt	Das Projekt, das in Ihrer Integration gestartet werden soll.
Workday-In	Der Name des Workday-In-Transports in Ihrer Integration.
Startparameter	Die Startparameter Ihrer Integration.

5. Klicken Sie auf Anwenden.
6. Klicken Sie auf Ausführen.

Ergebnisse

Studio führt Ihre Integration mit den Details Ihrer neuen Startkonfiguration aus.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

Integrationsmonitoring und Problembehebung

Prozessmonitor

Konzept: Integrationsereignis-IDs

Workday weist allen Integrationsereignissen eine ID zu, mit der Sie Integrationsereignisse vom Start bis zum Abschluss verfolgen können. Wenn Sie eine Integration aus Workday Studio starten, wird die ID des Integrationsereignisses in der Spalte ID der Ansicht Prozessmonitor angezeigt.

Referenz: Symboleleistenschaltflächen für Prozessmonitor

Schaltfläche	Beschreibung
Aktualisieren	Aktualisiert die Ansicht Prozessmonitor.
Prozess löschen	Löscht einen Prozess aus der Ansicht Prozessmonitor.
Prozess-ID in Zwischenablage kopieren	Kopiert eine Integrationsereignis-ID in Ihre Zwischenablage.
Prozess abbrechen	Bricht einen Prozess ab.
Konsolidierten Bericht anzeigen	Öffnet einen konsolidierten Bericht im Betrachter für konsolidierte Berichte.
Protokolldatei anzeigen	Zeigt die Protokolldatei für einen Prozess im Fenster Protokolldatei anzeigen an.
Anforderungsnachricht anzeigen	Zeigt die Anforderungsnachricht für einen Prozess im Fenster Anforderungsnachricht anzeigen an.
Konsolidierten Bericht importieren	Öffnet das Fenster Konsolidierten Bericht importieren.
Protokolldatei importieren	Öffnet das Fenster Protokolldatei importieren.
Anforderungsnachricht importieren	Öffnet das Fenster Anforderungsnachricht importieren.
Automatische Aktualisierung aktivieren	Aktiviert die automatische Aktualisierung für die Ansicht Prozessmonitor.
Workday: Prozessmonitor	Öffnet den Workday-Prozessmonitor auf dem Register Workday.
Workday	Öffnet die Workday-Ansicht und zeigt den ESB-Aufruf des Integrationsservers für Ihre Integration an.
Menü anzeigen	Zeigt die Symboleistenoptionen für die Ansicht Prozessmonitor an.

Betrachter für konsolidierte Berichte

Konsolidierte Berichte für Integrationen laden

Vorbereitungen

- Stellen Sie eine Integration in Workday bereit.
- Sicherheit: Domäne *Integration Event* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

In Studio zeigt das Register Betrachter für konsolidierte Berichte Performance-Informationen zum Timing, zur Speichernutzung und zur Anzahl der Aufrufe pro Schritt in Ihrer Integration an. Mit konsolidierten Berichten können Sie den Prozessablauf Ihrer Integrationen analysieren.

Prozedur

1. Wählen Sie in der Ansicht Cloud Explorer einen Cloud-Serverknoten aus und blenden Sie ihn ein.
2. Wählen Sie einen Sammlungsknoten aus und blenden Sie ihn ein.
3. Klicken Sie mit der rechten Maustaste auf einen Integrationsknoten.
4. Wählen Sie Integration starten aus.
5. Klicken Sie auf Starten.
6. Klicken Sie in der Ansicht Prozessmonitor mit der rechten Maustaste auf Ihre Integration.
7. Wählen Sie Konsolidierten Bericht anzeigen aus.

Ergebnisse

Ihr konsolidierter Bericht wird auf dem Register Betrachter für konsolidierte Berichte angezeigt.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Integrationen aus Studio starten](#) auf Seite 345

Konsolidierte Berichte in Ihrem Arbeitsbereich speichern

Vorbereitungen

- Stellen Sie eine Integration in Workday bereit.
- Sicherheit: Domäne *Integration Event* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Mit konsolidierten Berichten können Sie den Prozessablauf Ihrer Integrationen analysieren. Sie können konsolidierte Berichte in Ihrem Arbeitsbereich speichern.

Prozedur

1. Wählen Sie in der Ansicht Cloud Explorer einen Cloud-Serverknoten aus und blenden Sie ihn ein.
2. Wählen Sie einen Sammlungsknoten aus und blenden Sie ihn ein.
3. Klicken Sie mit der rechten Maustaste auf einen Integrationsknoten.
4. Wählen Sie Integration starten aus.
5. Klicken Sie auf Starten.

6. Klicken Sie in der Ansicht Prozessmonitor mit der rechten Maustaste auf Ihre Integration.
7. Wählen Sie Konsolidierten Bericht anzeigen aus.
8. Klicken Sie auf dem Register Betrachter für konsolidierte Berichte mit der rechten Maustaste auf den konsolidierten Bericht.
9. Wählen Sie Speichern als
10. Wählen Sie im Fenster Konsolidierten Bericht speichern einen übergeordneten Ordner für Ihren konsolidierten Bericht aus.
11. Klicken Sie auf Beenden.

Ergebnisse

Studio speichert Ihren konsolidierten Bericht in Ihrem Arbeitsbereich.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Integrationen aus Studio starten](#) auf Seite 345

Details zum Workday-Integrationsereignis mithilfe konsolidierter Berichte anzeigen

Vorbereitungen

- Stellen Sie eine Integration in Workday bereit.
- Sicherheit: Domäne *Integration Event* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Mit konsolidierten Berichten können Sie den Prozessablauf Ihrer Integrationen analysieren. Sie können die Details Ihres Workday-Integrationsereignisses mithilfe von konsolidierten Berichten in Studio anzeigen.

Prozedur

1. Wählen Sie in der Ansicht Cloud Explorer einen Cloud-Serverknoten aus und blenden Sie ihn ein.
2. Wählen Sie einen Sammlungsknoten aus und blenden Sie ihn ein.
3. Klicken Sie mit der rechten Maustaste auf einen Integrationsknoten.
4. Wählen Sie Integration starten aus.
5. Klicken Sie auf Starten.
6. Klicken Sie in der Ansicht Prozessmonitor mit der rechten Maustaste auf Ihre Integration.
7. Wählen Sie Konsolidierten Bericht anzeigen aus.
8. Wählen Sie auf dem Register Betrachter für konsolidierte Berichte die Option In Workday anzeigen.
9. Melden Sie sich auf dem Register Workday bei Ihrem Workday-Mandanten an. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Benutzername	Der Benutzername der Zugangsdaten für Ihren Workday-Mandanten.
Kennwort	Das Kennwort der Zugangsdaten für Ihren Mandanten.

Ergebnisse

Die Details Ihres Workday-Integrationsereignisses werden auf dem Register Workday in Studio angezeigt.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Integrationen aus Studio starten](#) auf Seite 345

Anforderungsnachrichten für Integrationsereignisse mithilfe konsolidierter Berichte anzeigen

Vorbereitungen

- Stellen Sie eine Integration in Workday bereit.
- Sicherheit: Domäne *Integration Event* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Mit konsolidierten Berichten können Sie den Prozessablauf Ihrer Integrationen analysieren. Sie können die Anforderungsnachrichten für Ihre Integrationsereignisse mithilfe konsolidierter Berichte in Studio anzeigen.

Prozedur

1. Wählen Sie in der Ansicht Cloud Explorer einen Cloud-Serverknoten aus und blenden Sie ihn ein.
2. Wählen Sie einen Sammlungsknoten aus und blenden Sie ihn ein.
3. Klicken Sie mit der rechten Maustaste auf einen Integrationsknoten.
4. Wählen Sie Integration starten aus.
5. Klicken Sie auf Starten.
6. Klicken Sie in der Ansicht Prozessmonitor mit der rechten Maustaste auf Ihre Integration.
7. Wählen Sie Konsolidierten Bericht anzeigen aus.
8. Wählen Sie auf dem Register Betrachter für konsolidierte Berichte die Option Anforderungsnachricht anzeigen.

Ergebnisse

Die Anforderungsnachricht für Ihr Integrationsereignis wird im Fenster Anforderungsnachricht anzeigen angezeigt.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Integrationen aus Studio starten](#) auf Seite 345

Integrationen mit dem Ereignisprotokoll des Betrachters für konsolidierte Berichte überwachen

Vorbereitungen

- Stellen Sie eine Integration in Workday bereit.
- Sicherheit: Domäne *Integration Event* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Mit konsolidierten Berichten können Sie den Prozessablauf Ihrer Integrationen analysieren. Sie können Ihre Integrationen mithilfe des Ereignisprotokolls auf dem Register Betrachter für konsolidierten Bericht in Studio überwachen.

Prozedur

1. Wählen Sie in der Ansicht Cloud Explorer einen Cloud-Serverknoten aus und blenden Sie ihn ein.
2. Wählen Sie einen Sammlungsknoten aus und blenden Sie ihn ein.
3. Klicken Sie mit der rechten Maustaste auf einen Integrationsknoten.
4. Wählen Sie Integration starten aus.
5. Klicken Sie auf Starten.
6. Klicken Sie in der Ansicht Prozessmonitor mit der rechten Maustaste auf Ihre Integration.
7. Wählen Sie Konsolidierten Bericht anzeigen aus.
8. Wählen Sie auf dem Register Betrachter für konsolidierte Berichte die Option Ereignisprotokoll.

Ergebnisse

Das Ereignisprotokoll für Ihre Integration wird auf dem Register Betrachter für konsolidierte Berichte angezeigt.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Integrationen aus Studio starten](#) auf Seite 345

Integrationen mit dem Profilprotokoll des Betrachters für konsolidierte Berichte überwachen

Vorbereitungen

- Stellen Sie eine Integration in Workday bereit.
- Sicherheit: Domäne *Integration Event* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Mit konsolidierten Berichten können Sie den Prozessablauf Ihrer Integrationen analysieren. Sie können Ihre Integrationen mithilfe des Profilprotokolls auf dem Register Betrachter für konsolidierten Bericht in Studio überwachen.

Prozedur

1. Wählen Sie in der Ansicht Cloud Explorer einen Cloud-Serverknoten aus und blenden Sie ihn ein.
2. Wählen Sie einen Sammlungsknoten aus und blenden Sie ihn ein.
3. Klicken Sie mit der rechten Maustaste auf einen Integrationsknoten.
4. Wählen Sie Integration starten aus.
5. Klicken Sie auf Starten.
6. Klicken Sie in der Ansicht Prozessmonitor mit der rechten Maustaste auf Ihre Integration.
7. Wählen Sie Konsolidierten Bericht anzeigen aus.
8. Wählen Sie auf dem Register Betrachter für konsolidierte Berichte die Option Profilprotokoll.

Ergebnisse

Das Profilprotokoll für Ihre Integration wird auf dem Register Betrachter für konsolidierte Berichte angezeigt. Die verwendete Zeitzone ist PST.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Integrationen aus Studio starten](#) auf Seite 345

Assembly-Ausführungen für konsolidierte Berichte erneut wiedergeben

Vorbereitungen

- Stellen Sie eine Integration in Workday bereit.
- Sicherheit: Domäne *Integration Event* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Mit konsolidierten Berichten können Sie den Prozessablauf Ihrer Integrationen analysieren. Mit konsolidierten Berichten in Studio können Sie für Ihre Integrationen Assembly-Ausführungen erneut wiedergeben.

Prozedur

1. Wählen Sie in der Ansicht Cloud Explorer einen Cloud-Serverknoten aus und blenden Sie ihn ein.
2. Wählen Sie einen Sammlungsknoten aus und blenden Sie ihn ein.
3. Klicken Sie mit der rechten Maustaste auf einen Integrationsknoten.
4. Wählen Sie Integration starten aus.
5. Klicken Sie auf Starten.
6. Klicken Sie in der Ansicht Prozessmonitor mit der rechten Maustaste auf Ihre Integration.
7. Wählen Sie Konsolidierten Bericht anzeigen aus.
8. Wählen Sie auf dem Register Betrachter für konsolidierte Berichte die Option Erneut wiedergeben.

Ergebnisse

Die Assembly-Ausführung für Ihre Integration wird im Assembly-Editor angezeigt.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Integrationen aus Studio starten](#) auf Seite 345

Auditereignisse für konsolidierten Bericht mit Assembly-Elementen verknüpfen

Vorbereitungen

- Stellen Sie eine Integration in Workday bereit.
- Sicherheit: Domäne *Integration Event* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Mit konsolidierten Berichten können Sie den Prozessablauf Ihrer Integrationen analysieren. Sie können Auditereignisse für konsolidierte Berichte mit Assembly-Elementen in Studio verknüpfen.

Prozedur

1. Wählen Sie in der Ansicht Cloud Explorer einen Cloud-Serverknoten aus und blenden Sie ihn ein.
2. Wählen Sie einen Sammlungsknoten aus und blenden Sie ihn ein.
3. Klicken Sie mit der rechten Maustaste auf einen Integrationsknoten.
4. Wählen Sie Integration starten aus.
5. Klicken Sie auf Starten.

6. Klicken Sie in der Ansicht Prozessmonitor mit der rechten Maustaste auf Ihre Integration.
7. Wählen Sie Konsolidierten Bericht anzeigen aus.
8. Wählen Sie auf dem Register Betrachter für konsolidierte Berichte die Option Mit Editor verknüpfen.

Ergebnisse

Studio verknüpft Ihre Auditereignisse für konsolidierte Berichte mit Assembly-Elementen in Ihrer Integration. Wenn Sie einen Knoten auf dem Register Betrachter für konsolidierten Bericht auswählen, hebt Studio das entsprechende Assembly-Element in Ihrem Assembly-Diagramm hervor.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Integrationen aus Studio starten](#) auf Seite 345

Anmerkungen zum konsolidierten Bericht in Assembly-Diagrammen anzeigen

Vorbereitungen

- Stellen Sie eine Integration in Workday bereit.
- Sicherheit: Domäne *Integration Event* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Mit konsolidierten Berichten können Sie den Prozessablauf Ihrer Integrationen analysieren. Sie können Anmerkungen zu konsolidierten Berichten in Ihrem Assembly-Diagramm in Studio anzeigen.

Prozedur

1. Wählen Sie in der Ansicht Cloud Explorer einen Cloud-Serverknoten aus und blenden Sie ihn ein.
2. Wählen Sie einen Sammlungsknoten aus und blenden Sie ihn ein.
3. Klicken Sie mit der rechten Maustaste auf einen Integrationsknoten.
4. Wählen Sie Integration starten aus.
5. Klicken Sie auf Starten.
6. Klicken Sie in der Ansicht Prozessmonitor mit der rechten Maustaste auf Ihre Integration.
7. Wählen Sie Konsolidierten Bericht anzeigen aus.
8. Wählen Sie auf dem Register Betrachter für konsolidierte Berichte die Option Anmerkung hinzufügen.

Ergebnisse

Studio zeigt Anmerkungen zu konsolidierten Berichten in Ihrem Assembly-Diagramm im Assembly-Editor an.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Integrationen aus Studio starten](#) auf Seite 345

Konzept: Betrachter für konsolidierte Berichte

In Workday Studio zeigt das Register Betrachter für konsolidierte Berichte Performance-Informationen zum Timing, zur Speichernutzung und zur Anzahl der Aufrufe pro Schritt in Ihrer Integration an. Das Register Betrachter für konsolidierten Bericht fasst außerdem Workday Webservice-Aufrufe zusammen und hebt Integrationsfehler hervor.

Sie können konsolidierte Berichte mit Assemblys verknüpfen. Sie können auch Anmerkungen zu einem konsolidierten Bericht in jedem Assembly-Element in Ihrem Diagramm anzeigen, um Probleme und Fehler in Ihren Integrationen zu erkennen.

Das Register Viewer für konsolidierte Berichte wird als Teil der Workday Studio-Standardansicht angezeigt. Wenn Sie ihn schließen, können Sie ihn erneut öffnen, indem Sie Fenster > Anzeigen - Ansicht von > konsolidierten Berichten wählen.

Sie können konsolidierte Berichte über das Register Anzeige für konsolidierten Bericht oder aus der Ansicht "Prozessmonitor" für einen ausgewählten Integrationsprozess laden.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Integrationen aus Studio starten](#) auf Seite 345

Referenz: Fehlermeldungen im konsolidierten Bericht

Fehlermeldung	Beispiel:	Beschreibung
java.lang.Exception: Gewinn in E2: XSLT-Job fehlgeschlagen	<pre>com.capeclear.mediation.MediationException: Custom step id=waitJobComplete, ref=waitJobCompleteBean - Error while invoking bean, reason: Win on E2: XSLT job failed. The Job with referenceId <referenceid> has failed as it exceeded the maximum number of retry attempts, 1 Root Cause: java.lang.Exception: Win on E2: XSLT job failed.</pre>	Wird angezeigt, wenn Ihr Job die maximale Anzahl an Wiederholungsversuchen überschritten hat.
HttpAuthenticationRequiredException: Unauthorized	<pre>com.capeclear.mediation.MediationException: RetrieveStep id=null encountered a problem sending data to endpoint <endpoint>, reason:HttpAuthenticationRequiredException: Unauthorized.</pre>	Wird angezeigt, wenn Sie keine ausreichenden Berechtigungen für den Zugriff auf einen Anhang haben.
com.workday.esb.blobitory.BlobitoryMandanten-Domänen nicht aktiv	<pre>com.capeclear.mediation.MediationException: application=bpelEnterpriseInterface<version-number> - Failed to store resource in collection eib_xslt_collection</pre>	Ein Mandantenkonfigurationsfehler. Wenden Sie sich an den Support und Technical Operations, um sicherzustellen, dass der Mandant aktiv ist.

Fehlermeldung	Beispiel:	Beschreibung
	in workspace wdoms.	
java.lang.OutOfMemoryError: Java-Heap-Speicherplatz.	com.capeclear.mediation.MediationException: Custom step id=waitJobComplete, ref=waitJobCompleteBean - Error while invoking bean, reason: Win on E2: XSLT job failed. Integration Failed. com.capeclear.mediation.MediationException: Xslt step id=transformEIB failed to apply stylesheet mctx:vars/eib.xslt.	Wird angezeigt, wenn Ihre Integration schlecht geschriebene XSLT enthält. com.capeclear.mediation.MediationException:
java.xml.transform.TransformConfigurationException	com.capeclear.mediation.MediationException: Custom step id=waitJobComplete, ref=waitJobCompleteBean - Error while invoking bean, reason: Win on E2: XSLT job failed. Integration Failed. com.capeclear.mediation.MediationException: Xslt step id=transformEIB stylesheet mctx:vars/ eib.xslt located at mctx:vars/eib.xslt is invalid or missing, reason: javax.xml.transform.TransformerConfigurationException: Failed to compile stylesheet. 1 error detected. Error reported by XML parser org.xml.sax.SAXParseException; systemId: mctx:vars/ eib.xslt; lineNumber: 960; columnNumber: 5; The markup in the document following the root element must be well-formed. Root Cause: javax.xml.transform.TransformerConfigurationException: Failed to compile stylesheet. 1 error detected.	Wird angezeigt, wenn Ihre Integration ungültiges XSLT enthält. com.capeclear.mediation.MediationException: javax.xml.transform.TransformerConfigurationException: Wird angezeigt, wenn Sie einen ungültigen XSLT- Verarbeiter implementieren,
com.workday.services.server.capec	com.capeclear.mediation.MediationException: application=bpelACHTransformOutbound-21.0.06.267 -	Wird angezeigt, wenn Sie einen ungültigen XSLT- Verarbeiter implementieren,

Fehlermeldung	Beispiel:	Beschreibung
	<pre>Custom step id=aggregateElement, ref=DocumentElementProcessor - Error while invoking bean, reason: document processor step error Root Cause: com.workday.services.server.capeclear.bean.dep.ProcessingExc error transforming SOAP request element, reason:error transforming request element, reason:javax.xml.transform.TransformerException: org.w3c.dom.DOMException: HIERARCHY_REQUEST_ERR: An attempt was made to insert a node where it is not permitted.</pre>	um einen zu erstellen SoapElementProcessor Transformationsinstanz.
java.io.FileNotFoundException	<pre>com.capeclear.mediation.MediationException: application=<application- name> - Http Out transport id=retrieveSchemaFromOMS encountered error sending to endpoint <endpoint> Root Cause: java.io.FileNotFoundException: <keystore-file- location> (No such file or directory).</pre>	Wird angezeigt, wenn eine MediationException: Schlüsselspeicher-Datei auf dem Workday ESB-Server fehlt.
org.jaxen.UnresolvableException	<pre>com.capeclear.mediation.MediationException: Error eval step id=ValidatePaygroup expression=props['myprop.empl'] = vars['input.picof'].xpath('/ picof:Employee/ picof:Status/ picof:Pay_Group') failed to evaluate. Reason: unable to resolve property: vars['input.picof'].xpath('/ picof:Employee/ picof:Status/ picof:Pay_Group') Root Cause: org.jaxen.UnresolvableException: Cannot resolve namespace prefix 'picof'.</pre>	Wird angezeigt, wenn die MediationException: Definition des Namespace- Präfixes in Ihrer Build fehlt.
BadRequestException	com.capeclear.mediation.MediationException:	Wird angezeigt, wenn Ihre MediationException: Integration eine ungültige

Fehlermeldung	Beispiel:	Beschreibung
	<pre> Workday Out transport id=CallGetPos encountered problem sending to endpoint 'https://impl- cc.workday.com/ccx/ service/mycompany/ Staffing/v17' with operation 'Get_Positions_Request'. Details: Code: SOAP- ENV:Client.validationError, Reason: Validation error occurred. Invalid ID value. '?' is not a valid ID value for type = 'Position_ID' Root Cause: BadRequestException : Error code : unknown. </pre>	SOAP-Anforderungsmeldung generiert.
java.net.SocketTimeoutException	<pre> com.capeclear.mediation.MediationException: Transport error in email out transport id=emFailureRecords with endpoint=mailto: reason:com.capeclear.capecconnect.transport.TransportException: Transport messaging error: Unable to send message : Exception reading response Root Cause: java.net.SocketTimeoutException: Read timed out. </pre>	Wird angezeigt, wenn Ihre Integration ungültige SMTP-Details enthält.

relatedlinks

Aufgaben

[Integrationen in Workday bereitstellen](#) auf Seite 338

[Integrationen aus Studio starten](#) auf Seite 345

AUnit-Assembly-Tests

Ein AUnit-Test-Skelett für ein Assembly-Projekt erstellen

Warum und wann dieser Vorgang ausgeführt wird

AUnit ist ein Test-Framework von Workday Studio zum Entwickeln und Durchführen von Einheitentests für Assemblys. Mit AUnit-Tests Gehaltserhöhungen können Sie Testfälle für Ihre Baugruppen erstellen.

Prozedur

1. Wählen Sie Datei > Neues > Workday-Zusammenfassungsprojekt.

2. Geben Sie im Feld Projektname einen Namen für Ihr Projekt ein.
Projektnamen dürfen keine Leerzeichen enthalten.
3. Wählen Sie im Feld Ziellaufzeit die Option *Workday Laufzeit*.
4. Wählen Sie im Feld Konfigurationen die Option *Assembly (mit Java-Unterstützung)* aus.
5. Klicken Sie auf Finish.
6. Wählen Sie Datei > Neue > andere > Klasse.
7. Im Menü Neue Java-Klasse sind folgende Informationen zu beachten:

Option	Bezeichnung
Paket	Geben Sie einen Paketnamen ein.
Name	Geben Sie einen Namen für die neue Java-Klasse ein.
Oberklasse	Klicken Sie auf Durchsuchen und wählen Sie eine Option <code>com.workday.aunit.AssemblyTestCase</code> .

8. Klicken Sie auf Finish.
9. Klicken Sie in der Ansicht Projekt-Explorer mit der rechten Maustaste auf den Ordner src in Ihrem Projekt.
10. Wählen Sie JUnit-Bibliothek hinzufügen.

Ergebnisse

Studio zeigt Ihr AUnit-Test-Skelett im Java-Editor an.

Nächste Maßnahme

Führen Sie Einheitentests für Ihre Assemblys aus.

Konzept: AUnit-Tests

AUnit ist ein Test-Framework von Workday Studio, mit dem Sie Einheitentests für Assemblys entwickeln und durchführen können. Sie können AUnit-Tests in Java mithilfe von Testskeletten entwickeln, die Sie in Ihren Assembly-Projekten erstellen. AUnit-Test-Skelette unterstützen folgende Features:

- Aktionen
- Anmerkungen
- Assertionen

Sie können diese Features verwenden, um Testfälle für Ihre Assemblys zu erstellen.

Beispiel:

```
package com.workday.training.aunit;

import java.io.InputStream;

import com.workday.aunit.AssemblyTestCase;
import com.workday.aunit.actions.Action;
import com.workday.aunit.actions.StandardAction;
import com.workday.aunit.annotations.AssemblyTest;
import com.workday.aunit.annotations.AssertAfter;
import com.workday.aunit.annotations.AtComponent;
import com.workday.aunit.annotations.UnitTest;
```

```
@AssemblyTest(project = "AUnitTrainer", displayLabel = "Citizens of the
Cloud")
```

```

public class SimpleMediationTest extends AssemblyTestCase {

    @UnitTest(startComponent = "Simple-Mediation")
    public void testSimpleMediation() throws Exception {
        setMessagePart(0, "test/simple-input.xml", "text/xml");
    }

    // Using the @AssertAfter annotation and the assertTrue assertion.    //
    @AssertAfter(id = "Simple-Mediation", step="transform")
    public void verifySimpleMediation() throws Exception {
        assertTrue(compare(
            getTestResourceInputStream("test/simple-expected.xml"), "text/
xml",
            (InputStream) getMediationContext().getMessage().getMessagePart(0,
InputStream.class),
            "text/xml", Comparator.dom));
    }

    @AtComponent(id="Next-Thing")
    public Action handleAfterMediation() throws Exception {
        return new StandardAction(Action.Type.terminate);
    }
}

```

Anmerkungen

Alle AUnit-Tests müssen eine enthalten @AssemblyTest Anmerkung. Diese Anmerkung identifiziert Ihren AUnit-Test und ermöglicht es Ihnen, den Test Ihrem Baugruppenprojekt zuzuordnen. Beispiel:

```
@AssemblyTest(project = "AUnitTrainer")
```

Anmerkungen zu Testmethoden definieren den Startpunkt für einen Einheitentest in der Verarbeitungskette der Assembly. Beispiel:

```
@UnitTest(startComponent = "Simple-Mediation")
```

Assertionen

Assertionen definieren boolesche Testbedingungen. Wenn eine Assertion fehlschlägt, bricht AUnit die Ausführung der Assembly ab und zeigt das Testergebnis als Fehler an. Wenn eine unerwartete Ausnahme auftritt, bricht AUnit die Batch-Ausführung ab und zeigt das Testergebnis als Fehler an.

Beispiel: Im folgenden Beispiel wird eine verwendet @AssertAfter Anmerkung und eine assertTrue Assertion, um zu prüfen, ob der Root-Nachrichtenteil nach der Ausführung der Vermittlung die erwartete XML-Datei enthält.

```

@AssertAfter(id = "Simple-Mediation", step="transform")
public void verifySimpleMediation() throws Exception {
    assertTrue(compare(
        getTestResourceInputStream("test/simple-expected.xml"), "text/xml",
        (InputStream) getMediationContext().getMessage().getMessagePart(0,
InputStream.class),
        "text/xml", Comparator.dom));
}

```

Aktionen

Aktionen definieren das Verhalten von AUnit-Tests, wenn die Ausführung der Assembly eine Komponente oder einen Ausstiegspunkt erreicht. Beispiel:

```

@AtComponent(id="Next-Thing")
public Action
    handleAfterMediation() throws Exception {

```

```
return new StandardAction(Action.Type.terminate); }
```

Referenz: Anmerkungen zum AUnit-Test

Annotation	Beschreibung	Beispiel:
@AssemblyTest	Identifiziert einen Test als AUnit-Test und ordnet ihn einer Baugruppe zu.	<pre>@AssemblyTest(project = "AUnitTrainer", displayLabel = "Citizens of the Cloud")</pre>
@UnitTest	Identifiziert einen einzelnen Testfall.	<pre>@UnitTest(startComponent = "Simple-Mediation") public void testSimpleMediation() throws Exception { setMessagePart(0, "test/simple- input.xml", "text/xml"); }</pre>
@AssertBefore	Führt Assertionen aus, bevor eine benannte Baugruppenkomponente oder ein benannter Schritt ausgeführt wird.	<pre>@AssertBefore(id="Store-Output") public void verifySummaryDoc() throws Exception { assertEquals("10", evaluateExpression("parts[0].xpath summary/ wd:WorkerSummary')")); }</pre>
@AssertAfter	Führt Assertionen nach der Ausführung einer benannten Build-Komponente oder eines Schritts aus.	<pre>@AssertAfter(id = "Simple-Mediation", step="transform") public void verifySimpleMediation() throws Exception { assertTrue(compare(getTestResource(simple-expected.xml"), "text/xml", (InputStream) getMediationContext().getMessage(), InputStream.class), "text/xml", Comparator.dom)); } @AssertAfter(id="Process", step="transform") public void verifySummary() throws Exception { assertEquals("Ace Developer", evaluateExpression("parts[0].xpath wd:WorkerSummary/ wd:Name')")); assertEquals("2010-07-27-07:00", evaluateExpression("parts[0].xpath wd:WorkerSummary/ wd:HireDate')")); }</pre>

Annotation	Beschreibung	Beispiel:
		}
@AssertAfterCompletion	Führt Assertionen nach Abschluss der Build aus.	<pre>@AssertAfterCompletion public void verifyErrorHandling() throws Exception { assertEquals(3, evaluateExpression("props['error.c</pre>
@AtComponent	Definiert, wo der Test die Baugruppe abschneidet, um die Verifizierung durchzuführen, und definiert das Testverhalten für die benannte Komponente.	<pre>@AtComponent(id="Get- workers") public Action handleGetWorkersInvocation() throws Exception { setMessagePart(0, "test/fromworkdayin/ workdayin_workers_response.xml", "text/xml"); return new StandardAction(Action.Type.mock); }</pre>

Referenz: Aktionen des AUnit-Tests

Aktionen	Beschreibung	Beispiel:
continue_to_next	Setzt den AUnit-Test für das nächste Element in Ihrer Baugruppe fort.	<pre>return new StandardAction(Action.Type.continu</pre>
invoke	Ruft den Endpunkt auf, der durch den ausgehenden Transport Ihrer Baugruppe angegeben wurde.	<pre>return new StandardAction(Action.Type.invoke)</pre>
mock	Modelliert eine vom ausgehenden Transport zurückgegebene Antwortnachricht, indem eine Eingabedatei angegeben wird, die die Antwortnachricht enthält.	<pre>return new StandardAction(Action.Type.mock);</pre>
mock_exception	Simuliert eine Ausnahme, die vom ausgehenden Transport ausgelöst wird, indem eine Instanz der folgenden Klasse angegeben wird: com.Capeclear.CapeConnect.transport.SchlechtRequestException(msg)	<pre>new StandardAction(Action.Type.mock_ex new BadRequestException("Include message here..."));</pre>
terminate	Beendet den Test.	<pre>return new StandardAction(Action.Type.termina</pre>

Webservice-Tests

Webservices mit einer WSDL-Datei testen

Warum und wann dieser Vorgang ausgeführt wird

Sie können den Webservice-Tester von Workday Studio verwenden, um Beispielanforderungsnachrichten mithilfe eines beliebigen in einer WSDL-Datei beschriebenen Vorgangs zu senden.

Prozedur

1. Klicken Sie in der Studio-Symboleiste auf Webservice im Tester testen.
2. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Datei	Zeigt den Speicherort der ausgewählten WSDL-Datei an.
Arbeitsbereich	Wählen Sie eine WSDL-Datei aus Ihrem Arbeitsbereich.
Extern	Wählen Sie eine WSDL-Datei an einem externen Speicherort.
URL	Wählen Sie eine WSDL-Datei aus einer URL.
Antwortendpunkt und Authentifizierungsdetails für Sandbox-Umgebung initialisieren	Gibt an, ob Folgendes zutrifft: • Das Feld Standort des Testers ist mit dem Webservice-Endpunkt vorausgefüllt. • Das Benutzername-Token für WS-Security ist mit Ihren Sandbox-Benutzer-Anmeldedaten vorausgefüllt. Diese Option ist standardmäßig aktiviert.
Ende	Klicken Sie auf diese Option, wenn Sie eine WSDL-Datei ausgewählt haben. Alternativ können Sie auch darauf klicken, ohne eine Datei anzugeben, um den Webservice-Tester zu öffnen und den Speicherort und die Nachricht Ihrer Wahl einzugeben.

3. Wählen Sie eine WSDL-Datei aus Ihrem Arbeitsbereich, einem externen Speicherort oder einer URL. Klicken Sie auf Beenden.
4. Studio öffnet den Webservice-Tester. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Standort	Der Webservice-Endpunkt.
PortType	Die vom Webservice offengelegten WSDL-Vorgänge und die zugehörige Bindung.
Vorgang	Vorgänge aus der WSDL-Datei. Wenn Sie einen auswählen, generiert Studio eine Beispielnachricht im Bereich Anforderung.

Option	Bezeichnung
SOAP action	Der Header „SOAPAction“ in der HTTP-Anforderung.
Anforderung	Die Anforderungsnachricht, die Studio an den Webservice sendet.
Antwort	Die Antwortnachricht, die Studio vom Webservice erhält.

5. Ändern Sie die XML-Datei im Bereich Anforderung nach Bedarf. Um einen Anhang hinzuzufügen, klicken Sie auf Anhang hinzufügen und dann auf Bearbeiten, um den Speicherort der Datei sowie Name und MIME-Typ des Anhangs anzugeben.
6. Klicken Sie auf Send Request, um die Anforderungsnachricht an den Webservice unter der URL zu senden, die im Feld Standort angegeben ist. Studio zeigt die Antwort im Bereich Antwort an.

Webservices mit einer XSD-Datei testen

Warum und wann dieser Vorgang ausgeführt wird

Mit dem Webservice-Tester von Workday Studio können Sie Beispielanforderungsnachrichten senden, indem Sie jedes Element verwenden, das in einer XSD-Datei beschrieben ist.

Prozedur

1. Klicken Sie in der Studio-Symbolleiste auf Webservice im Tester testen.
2. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Datei	Zeigt den Speicherort der ausgewählten XSD-Datei an.
Arbeitsbereich	Wählen Sie eine XSD-Datei aus Ihrem Arbeitsbereich aus.
Extern	Wählen Sie eine XSD-Datei von einem externen Speicherort aus.
URL	Wählen Sie eine XSD-Datei aus einer URL aus.
Antwortendpunkt und Authentifizierungsdetails für Sandbox-Umgebung initialisieren	Gibt an, ob Folgendes zutrifft: • Das Feld Standort des Testers ist mit dem Webservice-Endpunkt vorausgefüllt. • Das Benutzername-Token für WS-Security ist mit Ihren Sandbox-Benutzer-Anmeldedaten vorausgefüllt. Diese Option ist standardmäßig aktiviert.
Ende	Klicken Sie auf diese Option, wenn Sie eine XSD-Datei ausgewählt haben. Alternativ können Sie auch darauf klicken, ohne eine Datei anzugeben, um den Webservice-Tester zu öffnen und den Speicherort und die Nachricht Ihrer Wahl einzugeben.

3. Wählen Sie eine XSD-Datei aus Ihrem Arbeitsbereich, einen externen Speicherort oder eine URL aus. Klicken Sie auf Beenden.
4. Studio öffnet den Webservice-Tester. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Standort	Der Webservice-Endpunkt. Da der Standortwert in einer XSD nicht verfügbar ist, müssen Sie den Standort manuell eingeben.
Elemente	Elementdefinitionen aus der XSD-Datei. Wenn Sie eine auswählen, generiert Studio eine Beispielnachricht im Bereich Anforderung.

5. Ändern Sie die XML-Datei im Bereich Anforderung nach Bedarf. Um einen Anhang hinzuzufügen, klicken Sie auf Anhang hinzufügen und dann auf Bearbeiten, um den Speicherort der Datei sowie Name und MIME-Typ des Anhangs anzugeben.
6. Klicken Sie auf Send Request, um die Anforderungsnachricht an den Webservice unter der URL zu senden, die im Feld Standort angegeben ist. Studio zeigt die Antwort im Bereich Antwort an.

Einzelne Webservices mit dem Schema Explorer testen

Vorbereitungen

Zeigen Sie Workday-Webservice-WSDLs oder -XSDs im Schema Explorer an.

Warum und wann dieser Vorgang ausgeführt wird

Sie können einzelne Webservices testen, indem Sie mit dem Webservice-Tester von Workday Studio beispielhafte SOAP- oder XML-Anforderungsnachrichten über den Schema Explorer erstellen.

Prozedur

1. Um eine Beispielanforderungsnachricht für einen bestimmten Webservice zu generieren, klicken Sie mit der rechten Maustaste auf ein WSDL- oder XSD-Element in der Ansicht Schema Explorer. Wählen Sie entweder Open request in Web Service Tester oder Open SOAP wrapped request in Web Service Tester aus.
2. Studio öffnet die Ansicht WS Tester. Im Abschnitt Ziel wird der Standort des Webservice-Endpunkts angezeigt.
3. Wählen Sie in der Dropdown-Liste Content-Type die Medienart und die Medienunterart der Nachricht aus:

Option	Bezeichnung
text/plain; charset=utf-8	Gibt Klartext ohne Formatierungsbefehle oder -anweisungen an.
text/xml; charset=utf-8	Gibt XML-Text an.
application/octet-stream	Gibt eine nicht erkannte Art an. Beispiel: Wählen Sie diese Option aus, wenn Sie eine Binärdatei an die Nachricht anhängen.

4. Ändern Sie die XML-Datei im Bereich Anforderung nach Bedarf. Um einen Anhang hinzuzufügen, klicken Sie auf Anhang hinzufügen und dann auf Bearbeiten, um den Speicherort der Datei sowie Name und MIME-Typ des Anhangs anzugeben.
5. Klicken Sie auf Send Request, um die Anforderungsnachricht an den Webservice unter der URL zu senden, die im Feld Standort angegeben ist. Studio zeigt die Antwort im Bereich Antwort an.

relatedlinks

Aufgaben

[Workday Webservice-WSDLs anzeigen](#) auf Seite 288

Get-Vorgänge und Anforderungsnachrichten mit dem Anforderungs-BUILDER-Assistenten testen

Vorbereitungen

Zeigen Sie Workday-Webservice-WSDLs oder -XSDs im Schema Explorer an.

Warum und wann dieser Vorgang ausgeführt wird

Sie können den Workday Web Service Request Builder Wizard verwenden, um das Testen vieler Get-Vorgänge und Anforderungsnachrichten zu vereinfachen. Auswählbare Vorgänge und Anforderungsnachrichten haben als Kennzeichnung einen Pfeil in der oberen linken Ecke ihres Symbols.

Prozedur

1. Klicken Sie mit der rechten Maustaste auf den betreffenden Get-Knoten in der Ansicht Schema Explorer und wählen Sie dann den Workday Web Service Builder Wizard aus.
2. Wählen Sie eine der folgenden Optionen aus:
 - Eine Referenz-ID verwenden, um bestimmte Daten abzurufen.
 - Alle verfügbaren Datensätze aus dem Webservice abrufen.
3. Wenn Sie sich für die Verwendung einer Referenz-ID entschieden haben, geben Sie diese ein.
4. Wählen Sie die Antwortgruppen aus, die Sie einschließen möchten.
5. Wählen Sie eine der folgenden Optionen aus:
 - Die Beispielanforderung in die Zwischenablage kopieren.
 - Die Beispielanforderung in einer neuen WS-Tester-Ansicht öffnen.
 - Den Inhalt der aktuellen WS-Tester-Ansicht ersetzen.
6. Klicken Sie auf Beenden.

Authentifizierung für Webservice-Tester konfigurieren

Vorbereitungen

Erstellen Sie einen Webservice-Test.

Warum und wann dieser Vorgang ausgeführt wird

Sie können eine Reihe von Sicherheitsoptionen im Webservice tester von Workday Studio konfigurieren. Bei den Authentifizierungsoptionen können Sie angeben, ob WS-Sicherheit oder Basisauthentifizierung auf SOAP-Anforderungsmeldungen angewendet oder die interne Authentifizierung aufgerufen werden soll cc-authenticate Service

Prozedur

1. Klicken Sie in der Ansicht "WS-Test" auf Sendeoptionen.
2. Aktivieren Sie das Kontrollkästchen Authentifizierung. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
WS-Security „UsernameToken“	Ermöglicht die Angabe eines Benutzernamens und Kennworts zur Verwendung mit WS-

Option	Bezeichnung
	Security, ohne dass Sie eine WS-Security-Konfigurationsdatei auswählen müssen. Studio fügt ein XML-Fragment mit dem von Ihnen angegebenen Benutzernamen und Kennwort in den Abschnitt „SOAP-Header“ der generierten Anforderungsnachrichten ein.
Authentication Service	Der Webservicetester ruft die interne Instanz auf cc-authenticate-Service mit dem von Ihnen angegebenen Benutzernamen und Kennwort. Wenn der Benutzername und das Kennwort auf dem Server gültig sind, gibt Workday ein Sitzungsticket zurück, das der Webservicetester beim Senden der Anforderung zum SOAP-Header hinzufügt. Die Serververarbeitung umfasst einen Interceptor, der prüft, ob das bereitgestellte Sitzungsticket gültig ist. Standardmäßig besteht eine Verbindung zu cc-authenticate Der Service "" wird einmal durchgeführt und wiederverwendet. Wenn Sie Server stoppen und starten, wählen Sie vor jedem Senden Re-connect to the cc-authenticate service, um für jede Anforderung eine neue Verbindung zu erstellen.
Einfache Authentifizierung (präemptiv)	Ermöglicht die Angabe eines Benutzernamens und eines Kennworts für die präemptive einfache HTTP-Authentifizierung.

X.509-Authentifizierung für Webservice-Tester konfigurieren

Vorbereitungen

Erstellen Sie einen Webservice-Test. Konfigurieren Sie einen Integrationssystembenutzer mit einem Benutzernamen, der dem Aliasnamen eines Schlüsselspeichers entspricht, jedoch ohne @tenant-name Suffix. Weisen Sie dem Integrationssystembenutzer die entsprechende benutzerbasierte Gruppe für den Webservice zu.

Warum und wann dieser Vorgang ausgeführt wird

Sie können eine Reihe von Sicherheitsoptionen im Webservice-Tester von Workday Studio konfigurieren. Mit den Optionen für X.509-Authentifizierung können Sie einen Alias für den Schlüsselspeicher auswählen, der einem Integrationssystembenutzer und dem zugehörigen privaten und öffentlichen Schlüsselpaar entspricht.

Prozedur

1. Klicken Sie in der Ansicht WS-Tester auf Sendeoptionen.
2. Aktivieren Sie das Kontrollkästchen X.509-Authentifizierung.
3. Wählen Sie einen Benutzernamen aus der Dropdown-Liste Benutzername aus. Jedem Benutzernamen ist ein Schlüsselspeicher-Alias zugeordnet.

4. (Optional) Klicken Sie auf Schlüssel verwalten, um vorhandene X.509-Schlüssel zu verwalten oder ein neues Schlüsselpaar zu generieren. Studio fordert einmal pro Sitzung ein Kennwort für den Schlüsselspeicher an. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
X.509-Schlüsselpaar generieren	Hiermit können Sie ein X.509-Schlüsselpaar mit einem zugehörigen digitalen Zertifikat generieren.
Schlüssel aus Schlüsselspeicher entfernen	Entfernt das ausgewählte Schlüsselpaar.
Öffentlichen Schlüssel (Zertifikat) anzeigen	Zeigt den öffentlichen Schlüssel für das ausgewählte Schlüsselpaar an.
Schlüsselpaar importieren	Sie haben folgende Möglichkeiten: • Importieren Sie einen privaten Schlüssel und ein X.509-Zertifikat aus einem Java-Schlüsselspeicher. • Importieren Sie einen privaten Schlüssel und ein X.509-Zertifikat im OpenSSL-PEM-Format.
Alias umbenennen	Ermöglicht das Umbenennen des Schlüsselspeicher-Alias für das ausgewählte Schlüsselpaar.

Webservice-Tester konfigurieren – Erweiterte WS-Sicherheit

Vorbereitungen

Erstellen Sie einen Webservice-Test.

Warum und wann dieser Vorgang ausgeführt wird

Sie können eine Reihe von Sicherheitsoptionen im Webservice-Tester von Workday Studio konfigurieren. Mit den Optionen für Erweiterte WS-Sicherheitskonfiguration kann der Webservice-Tester eine Sicherheitskonfigurationsdatei lesen, die eine Abfolge von Sicherheitsschritten für die Anforderung und eine weitere Abfolge von Sicherheitsschritten für die Antwort angibt. Die Sicherheitsschritte werden auf die ausgehende Anforderungsnachricht angewendet und die Sicherheitscodierung aus der eingehenden Antwortnachricht entfernt. Der Server entfernt die Sicherheitscodierung beim Empfang und fügt die Sicherheitscodierung bei der Antwort hinzu.

Prozedur

1. Klicken Sie in der Ansicht WS-Tester auf Sendeoptionen.
2. Aktivieren Sie das Kontrollkästchen Erweiterte WS-Sicherheitskonfiguration.
3. Geben Sie eine Konfigurationsdatei-URL aus einem Arbeitsbereich oder einem externen Speicherort an.
4. (Optional) Wenn die Sicherheitskonfigurationsdatei relative Standorte für Werte wie den Standort des Schlüsselspeichers enthält, geben Sie eine Konfigurationsbasis-URL an. Studio stellt diese URL den relativen Standorten voran.
5. (Optional) Aktivieren Sie das Kontrollkästchen Informationen zur Sicherheitskonfiguration vor jedem Senden neu laden, um die Datei für jede Anforderung neu zu laden. Diese Option ist hilfreich, wenn Sie Änderungen an der Konfigurationsdatei vornehmen.

Konzept: Webservice-Tests

Mit dem Webservice-Tester von Workday Studio können Sie SOAP- oder XML-Anforderungsnachrichten basierend auf der Webservicebeschreibung (WSDL) oder einer Schemadefinition (XSD) erstellen. Das Tool zeigt auch alle Antwortnachrichten an, die der Webservice zurückgibt. Es unterstützt HTTP und HTTPS.

In dieser Tabelle sind die drei wichtigsten Möglichkeiten zum Starten des Webservice-Testers zusammengefasst:

Methode	Anmerkungen
Über die Studio-Menüleiste.	Ermöglicht die Auswahl einer WSDL- oder XSD-Datei zum Testen aus Ihrem Arbeitsbereich, einer URL oder einem externen Dateispeicherort. Sie können diese Option auch verwenden, um den Webservice-Tester ohne Bezug auf eine bestimmte Datei zu öffnen.
In der Ansicht Schema Explorer.	Mit diesem Webservice können Sie XML-Anforderungen für einen bestimmten WSDL-Vorgang oder ein bestimmtes Schemaelement entweder in einem Nicht-SOAP-Format oder in einem SOAP-Umschlagformat generieren.
Mit dem Assistenten zum Webserviceanforderungs-Builder von Workday.	Vereinfacht das Testen bestimmter Get-Vorgänge und Get-Anforderungsnachrichten.

Referenz: Schaltflächen der Symbolleiste des Webservice-Tester

Allgemein

Schaltfläche	Beschreibung
Anforderung senden	Ruft den Webservice auf. Die Antwortnachricht wird im Bereich Antwort angezeigt.
Sendeoptionen	Öffnet das Dialogfeld Sendeoptionen, in dem Sie Sicherheitsoptionen konfigurieren können.
Anforderung abbrechen	Die aktuelle Anforderung wird abgebrochen.
WSDL neu laden	Lädt die WSDL, die den Webservicetester steuert, neu. Dies ist hilfreich, wenn sich der Webservice geändert hat.
XML-Schrift in den Systemeinstellungen auswählen	Ermöglicht das Festlegen von Schrifteigenschaften für die XML-Datei, die in den Bereichen Anforderung und Antwort angezeigt wird.
Anforderungs-/Antworthistorie	Bietet Zugriff auf eine Liste der zuvor gesendeten Anforderungsnachrichten, sodass Sie diese auswählen und erneut senden können.
Nur Anforderung	Zeigt nur den Bereich Anforderung an.
Nur Antworten	Zeigt nur den Bereich Antwort an.
Horizontales Layout	Das Layout wird horizontal angeordnet.

Schaltfläche	Beschreibung
Vertikales Layout	Das Layout wird vertikal angeordnet.
Hilfe	Zeigt kontextbezogene Hilfe für den Webservice-Tester an.

Bereich „Anforderungen“

Schaltfläche	Beschreibung
Anforderung speichern	Die Anforderungsnachricht wird als XML-Datei gespeichert.
Anforderung laden	Die Anforderungsnachricht wird geladen.
Anforderung formatieren	Standardeinzug wird auf die Anforderungsnachricht angewendet. Der Text kann die Breite des Panels überschreiten.
Ursprüngliche oder zuletzt gesendete Anforderung wiederherstellen	Bei nicht gesendeten Nachrichten wird der Text in den ursprünglichen, unformatierten und unverpackten Zustand zurückgesetzt. Überschreibt alle manuellen Bearbeitungen. Bei gesendeten Nachrichten wird der Text auf den Status vor der letzten Sendeaktion zurückgesetzt.
Anhang hinzufügen	Mit diesem Webservice können Sie einen Dateianhang hinzufügen oder entfernen. Um der Anforderungsnachricht einen Anhang hinzuzufügen, klicken Sie auf Bearbeiten, wählen Sie eine Datei im Arbeitsbereich oder an einem externen Dateispeicherort aus und geben Sie Name und MIME-Typ an. Um einen Anhang aus einer Anforderungsnachricht zu entfernen, klicken Sie auf Entfernen. Um einen in einer Antwortnachricht enthaltenen Anhang zu speichern, klicken Sie unter dem Bereich Antwort auf Öffnen.
Schriftgröße erhöhen	Erhöht die Textgröße im Bereich Anforderung.
Schriftgröße verringern	Verringert die Textgröße im Bereich Anforderung .

Bereich „Antworten“

Schaltfläche	Beschreibung
Antwort speichern	Die Antwortnachricht wird als XML-Datei gespeichert.
Antwort laden	Die Antwortnachricht wird geladen.
Antwort formatieren	Standardeinzug wird auf die Antwortnachricht angewendet. Der Text kann die Breite des Panels überschreiten.

Schaltfläche	Beschreibung
Antwort verpacken	Wendet den Einzug an, formatiert den Text jedoch auch innerhalb der Grenzen der Breite des Panels.
Antwort wiederherstellen	Bei nicht gesendeten Antwortnachrichten wird der Text in den ursprünglichen, unformatierten und unverpackten Zustand zurückgesetzt. Bei Beispielantworten wird die geladene Datei aus dem Textbereich entfernt. Es werden alle manuellen Bearbeitungen rückgängig gemacht.
Beispielantwort	Es wird eine Beispielantwort für eine SOAP-Anforderungsnachricht generiert, die aus einer WSDL generiert wurde.
Schriftgröße erhöhen	Erhöht die Textgröße im Bereich Antwort.
Schriftgröße verringern	Verringert die Textgröße im Bereich Antwort.

Assembly-Debugging

Debug einer Assembly durchführen

Vorbereitungen

- Erstellen Sie ein Assembly-Projekt.
- Sicherheit: Domäne *Integration Debug* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Mit Workday Studio können Sie Fehler in Ihren Assemblys erkennen und debuggen. Sie können ein Debug für Assemblys durchführen, die Sie in Ihrer Workday-Serverumgebung bereitgestellt haben. Wenn eine Assembly Komponenten oder Schritte enthält, die auf eine statische Datei verweisen, stellen Sie vor dem Debuggen sicher, dass der referenzierte Dateipfad keine Leerzeichen enthält.

Prozedur

1. Klicken Sie in der Ansicht Project Explorer mit der rechten Maustaste auf ein Assembly-Projekt.
2. Wählen Sie Deploy to Workday aus.
3. Im Fenster Deploy to Workday ist Folgendes zu beachten:

Option	Bezeichnung
Workday-Umgebungen	Wählen Sie eine Workday-Zielumgebung für die Bereitstellung Ihrer Integration aus.
Sammlung wird bereitgestellt	Wählen Sie eine Sammlung aus.

4. Klicken Sie auf Beenden.
5. Wählen Sie in der Ansicht Cloud-Explorer den Cloud-Serverknoten der Workday-Zielumgebung aus, in der Sie Ihr Assembly-Projekt bereitgestellt haben, und blenden Sie ihn ein.
6. Wählen Sie die Sammlung aus, die Sie in Workday bereitgestellt haben, und blenden Sie sie ein.
7. Klicken Sie mit der rechten Maustaste auf einen Integrationsknoten in Ihrer Sammlung.

8. Wählen Sie Debug für Integration durchführen. Im Fenster Debug für Integration durchführen sind folgende Informationen zu beachten:

Option	Bezeichnung
Starten in	Die Workday-Zielumgebung für den Start Ihrer Debug-Sitzung.
Sammlung	Der Name der Sammlung, für die Sie ein Debugging durchführen.
Projekt	Der Name des Projekts, für das Sie ein Debug durchführen.
Workday-In	Der Name des Workday-In-Transports in dem Projekt, für das Sie ein Debug durchführen.
Startparameter	Die Startparameter des Projekts, für das Sie ein Debug durchführen. Um ein Debug mit alternativen Startparametern durchzuführen, klicken Sie auf Literalwerte auswählen und wählen Sie eine der Workday-IDs aus, die das CRF zurückgibt. Wenn Sie eine bestimmte Workday-ID kennen, die Sie als Startparameter verwenden möchten, klicken Sie auf Literalwert bearbeiten und geben Sie sie in das Feld Wert ein.

9. Klicken Sie auf Debug.

Anmerkung: Wenn zu Beginn der Debug-Laufzeit Probleme auftreten, sollten Sie die Werte Sitzungs-Wiederverbindungsversuche und Maximale Anzahl Iterationen für Startup-Wartestatus anpassen.

„Sitzung erneut verbinden“ bestimmt, wie oft Studio versucht, eine Debug-Sitzung mit der Debug-Laufzeit zu verbinden. Maximale Anzahl Iterationen für Wartestatus bei Start bestimmt, wie oft Studio prüft, ob die Debug-Laufzeit für den Empfang von Anforderungen bereit ist. Versuchen Sie, sie in 10er-Schritten zu erhöhen.

Sie können diese Werte für eine vorhandene Startkonfiguration im Dialogfeld Debug-Konfigurationen anpassen.

Um sie für zukünftige Startkonfigurationen anzupassen, wählen Sie Fenstereinstellungen > Workday > Build > Debug > Debug Runtime.

Ergebnisse

Die Ergebnisse Ihrer Debug-Sitzung werden in der Ansicht „Konsole“ angezeigt.

Konzept: Assembly-Debugger

Der Assembly-Debugger von Workday Studio ist ein Tool zum Debuggen von Assemblys in der Workday-Serverumgebung. Der Assembly-Debugger bietet Ihnen folgende Möglichkeiten:

- Den Laufzeitkontext der Assembly-Nachricht untersuchen
- Assembly-Threads aussetzen
- MVEL-Ausdrücke testen.

Wenn Ihre Assembly Komponenten oder Schritte enthält, die auf eine statische Datei verweisen, stellen Sie vor dem Debuggen sicher, dass der referenzierte Dateipfad keine Leerzeichen enthält.

Anmerkung: Sie können in einem Produktionsmandanten kein Debugging durchführen. Verwenden Sie zum Debuggen eine Nicht-Produktionsumgebung.

Haltepunkte festlegen

Mit Haltepunkten können Sie den Debugging-Prozess an bestimmten Stellen in Ihren Assemblys anhalten. Um Haltepunkte für Assembly-Elemente festzulegen, klicken Sie mit der rechten Maustaste auf ein Element und wählen Sie Toggle Assembly Breakpoint. In der Hauptansicht des Assembly-Editors wird in der oberen rechten Ecke des Elements ein blauer Kreis angezeigt, der einen neuen Haltepunkt anzeigt.

Management von Studio

Systemeinstellungen für Studio

Konzept: Systemeinstellungen des Assembly-Editors

Sie können auf der Seite mit den Systemeinstellungen des Builder-Editors Präferenzen für Ihre Baugruppendiagramme angeben. Um zur Seite "Einstellungen" zu navigieren, wählen Sie "Fenstereinstellungen > " für > Workday > Composite Editor aus. Wählen Sie einen der folgenden Knoten, um die zugehörige Einstellung für den Baugruppeneditor anzugeben:

- Farbe
- Schriften
- Migration
- Module

Farbe

Auf der Seite Systemeinstellungen für Diagrammfarben können Sie Farbpräferenzen für Komponenten in Ihrem Assembly-Diagramm angeben.

Einstellung	Beschreibung
Hintergrundfarbe	Gibt die Hintergrundfarbe für Assembly-Komponenten an.
Linienfarbe	Gibt die Linienfarbe für Assembly-Komponenten an.
Textfarbe	Gibt die Schriftfarbe für Assembly-Komponenten an.
Hintergrundfarbe Schritt	Gibt die Hintergrundfarbe für Assembly-Schritte an.
Header-Hintergrundfarbe	Gibt die Hintergrundfarbe für Header von Assembly-Komponenten an.

Schriften

Auf der Seite Systemeinstellungen für Diagrammschriften können Sie Schrifteinstellungen für Komponenten in Ihrem Assembly-Diagramm angeben.

Einstellung	Beschreibung
Standard	Gibt die Standardschriftart, den Stil und den Schriftgrad für die ID-Werte der Komponenten Ihres Assembly-Diagramms an.
Schritt	Gibt die Standardschriftart, den Stil und den Schriftgrad für die ID-Werte Ihrer Assembly-Schritte an.
Bedingter QuickInfo-Header	Gibt die Standardschriftart, den Stil und den Schriftgrad für QuickInfo-Header-Text an, der in Ihrem Assembly-Diagramm angezeigt wird.
Bedingter QuickInfo-Inhalt	Gibt die Standardschriftart, den Stil und den Schriftgrad für QuickInfo-Inhalt an, der in Ihrem Assembly-Diagramm angezeigt wird.

Migration

Auf der Seite Migration können Sie die Migration für Legacy-Assembly-Projekte, die mit früheren Versionen von Workday Studio erstellt wurden, aktivieren oder deaktivieren.

Wenn die Migration aktiviert ist, migriert Workday Studio die Assembly-Konfiguration für eine assembly.xml-Datei automatisch in die neueste Version von Studio, wenn Sie die Datei öffnen.

Um die automatische Migration zu deaktivieren, deaktivieren Sie das Kontrollkästchen Migration von Legacy-Assemblys im Assembly-Editor aktivieren, bevor Sie eine assembly.xml-Datei öffnen oder speichern.

Module

Bevor Sie mit der Entwicklung von Baugruppenmodulvorlagen beginnen, erstellen Sie einen Ordner, um Ihre Baugruppenmodule entweder in Ihrem Arbeitsbereich oder an einem externen Speicherort, z. B. einem Ordner in Ihrer Workday Studio-Installation, zu speichern.

Die Modulbibliothek in der Palette der Bausatz-Tools des Build-Editors zeigt nur die Build-module an, die zum Laden aus dem Modulordner ausgewählt sind, wie auf der -Seite für Moduleinstellungen im > Workday > - > Builder-Editor > für Fensterfestgelegt. Klicken Sie auf Arbeitsbereich durchsuchen oder Dateisystem durchsuchen, um den Modulordner zu suchen, und geben Sie den Pfad zu diesem Ordner an. Studio zeigt alle XMI-Dateien (XML Metadata Interchange) für das Assembly-Modul an, die sich im angegebenen Ordner in der Tabelle Module befinden. Aktivieren Sie das Kontrollkästchen neben jedem Assembly-Modul, das Sie in die Modulbibliothek laden möchten. Studio führt eine Vorlagenvalidierung durch und lädt nur gültige XMI-Dateien.

Zum Zeitpunkt des Designs fordert Sie der Assembly-Modul-Assistent auf, die XMI-Datei des Moduls in einem Ordner in Ihrem Arbeitsbereich zu speichern. Bei Bedarf können Sie die XMI-Dateien des Assembly-Moduls später aus Ihrem Arbeitsbereichsordner in den Modulordner kopieren, den Sie hier angeben, falls dieser sich vom Arbeitsbereichsordner unterscheidet.

Konzept: Systemeinstellungen der Verbindungen

Auf der Seite mit den Verbindungseinstellungen können Sie Ihre Zugangsdaten für die Integration-Cloud-Verbindung festlegen. Um zur Seite "Verbindungseinstellungen" zu navigieren, wählen Sie Fenstereinstellungen > für > Workday > -Verbindungen. Sie können Zugangsdaten sowohl für Sandbox- als auch für Produktionsumgebungen eingeben. Sie können auch Verbindungen zu anderen Umgebungen sowie Gruppen zu Ihren Verbindungen hinzufügen.

Konzept: Deployment-Systemeinstellungen

Auf der Seite Deployment-Systemeinstellungen können Sie angeben, ob beim Deployment in einer Workday-Umgebung der Projektquellcode in die CLAR aufgenommen werden soll. Aktivieren Sie dazu das Kontrollkästchen Quellcode in bereitgestellte CLAR einschließen.

Um den Quellcode aus der CLAR auszuschließen, deaktivieren Sie das Kontrollkästchen.

Wenn Sie die Option Quellcode in bereitgestellte CLAR einschließen aktivieren, können Sie die Quelle in ein Projekt in Ihrem Arbeitsbereich importieren. Klicken Sie dazu im Cloud Explorer mit der rechten Maustaste auf die Sammlung und wählen Sie Quelle importieren aus.

Konzept: HTTPS-Systemeinstellungen

Im Bereich HTTPS-Systemeinstellungen können Sie angeben, ob die Verwaltung von Truststore-Zertifikaten in Studio aktiviert werden soll. Wenn Sie das Kontrollkästchen Verwaltung von Truststore-Zertifikaten in Studio aktivieren und Studio versucht, eine Verbindung zu Workday über HTTPS herzustellen, wird eine Meldung angezeigt, in der Sie aufgefordert werden, dem Serverzertifikat für die Sandbox- oder Produktionsumgebung zu vertrauen. Die Zertifikate, denen Sie vertrauen, werden im Bereich Truststellen-Zertifikate angezeigt. Hier können Sie neu laden, alle entfernen oder ausgewählte Zertifikate löschen. Um den Bereich Truststore-Zertifikate zu öffnen, erweitern Sie den Link Workday > HTTPS im linken Navigationsbereich.

Konzept: Ivy-Systemeinstellungen

Workday Studio stellt ein Ivy-Plugin bereit, mit dem Sie Projektabhängigkeiten mit Apache Ivy verwalten können. Damit das Ivy-Plugin Ressourcen auflösen kann, können Sie die Einstellungs- und Eigenschaftendatei angeben, die in Ihrem Projekt verwendet werden soll.

Sie können folgende Einstellungen auf der Seite mit den Workday > Ivy -Systemeinstellungen konfigurieren:

- URL für Ivy-Einstellungen: Klicken Sie auf „Durchsuchen...“, wählen Sie dann die Datei `ivySettings.xml` oder geben Sie eine URL an. Die Einstellungsdatei definiert Variablen, die die Ivy-Engine laden kann, um Ressourcen aufzulösen.
- Ivy-Eigenschaftsdatei: Klicken Sie auf Durchsuchen ... und wählen Sie die `.properties`-Datei aus, auf die die Einstellungsdatei verweisen kann. Die Eigenschaftendatei enthält Eigenschaften, die Ivy als Variablen in folgendem Format lädt:

`name=value`

Aktivieren Sie das Kontrollkästchen `ivy.properties`-Datei aus Basisverzeichnis laden, falls vorhanden, um die `ivy.properties`-Datei automatisch aus Ihrem Basisverzeichnis zu laden.

Konzept: Systemeinstellungen verknüpfen

Auf der Seite Verknüpfungseinstellungen geben Sie an, ob Pfadvariablen mit absoluten Pfaden für verknüpfte Ressourcen in Ihrem Arbeitsbereich verwendet werden sollen. Um zur Seite mit den Verknüpfungseinstellungen zu navigieren, wählen Sie Fenstereinstellungen > für > Workday > - Verknüpfung. Die Einstellung PATH-Variablen für absolute Pfade verknüpfter Ressourcen verwenden wird automatisch aktiviert. Deaktivieren Sie das Kontrollkästchen, um diese Einstellung zu deaktivieren.

Konzept: Systemeinstellungen für MVEL-Validierung

Auf der Einstellungsseite MVEL-Validierung können Sie die Validierung für MVEL-Code konfigurieren. Die Einstellung MVEL-Validierung aktivieren wird automatisch aktiviert. Deaktivieren Sie das Kontrollkästchen, um diese Einstellung zu deaktivieren.

Konzept: Systemeinstellungen für Integrationspakete

Die Seite Systemeinstellungen für Integrationspakete ist nur für interne Studio-Entwickler zur Bereitstellung von Integrationspaketen vorgesehen. Interne Studio-Entwickler: Weitere Informationen zur Unterstützung von Integrationspaketen finden Sie in Ihrem internen Wiki.

Konzept: Systemeinstellungen „Laufzeit“

Workday Studio sucht automatisch nach Aktualisierungen der Workday-Laufzeit. Wenn Updates verfügbar sind, lädt Studio diese im Hintergrund herunter und benachrichtigt Sie bei der Installation. Sie können Ihre Download-, Installations- und Benachrichtigungseinstellungen auf der Seite **Fenstereinstellungen > Workday > Laufzeit > konfigurieren**.

Konzept: Server-Systemeinstellungen

Auf der Seite **Server-Systemeinstellungen** können Sie die Einstellung **Workday-Sammlungen** nicht mehr bereitstellen, wenn sie vom Server entfernt werden individuell anpassen. Diese Einstellung wird automatisch deaktiviert. Wenn also eine Sammlung bereits in der ausgewählten Workday-Umgebung bereitgestellt wird und Sie die Sammlung aus der Liste **Konfiguriert** entfernen, hebt Studio die Bereitstellung der Sammlung nicht auf. Aktivieren Sie das Kontrollkästchen, um diese Einstellung zu aktivieren. Wenn Sie im Assistenten für **Deploy to Workday** auf **Beenden** klicken, stellt Studio alle Sammlungen wieder bereit, die sich in Ihrem Arbeitsbereich befinden, aber nicht mehr in der Liste **Konfiguriert** aufgeführt sind.

Konzept: Anmeldeeinstellungen für den Begrüßungsbildschirm

Beim Start zeigt Workday Studio einen optionalen Begrüßungsbildschirm an, in dem Sie Verbindungsdetails für Ihre Workday-Umgebung eingeben können. Wenn Sie möchten, können Sie Ihre Details später unter **Verbindungsdetails** eingeben. Um den Begrüßungsbildschirm beim Start auszuschalten, aktivieren Sie das Kontrollkästchen **Don't show the login on startup** und klicken Sie dann auf **Anwenden**, um Ihre Änderungen zu speichern.

Konzept: Systemeinstellungen der Statusleiste

Auf der Seite der Systemeinstellungen der Statusleiste können Sie die maximale Anzahl von Statusleistenelementen angeben, die in der Verlaufsliste der Statusleiste gespeichert werden sollen, die unten rechts im Arbeitsbereich von Studio angezeigt wird. Geben Sie in das Feld **Maximale Anzahl historischer Elemente in der Statusleiste** die maximale Anzahl von Elementen ein, die Sie speichern möchten.

Konzept: Systemeinstellungen aktualisieren

Workday Studio unterstützt Aktualisierungen für Integrationen in Bereichen mit geringer Bandbreite über das Workday Content Delivery Network (CLN). Sie können den Zugriff auf das Workday CDN auf der **Workday > -Installations-/Aktualisierungsseite > für > Fenstereinstellungen** aktivieren.

Anmerkung: Die CDN-Nutzung erfordert uneingeschränkten ausgehenden Zugriff auf Port 443.

Konzept: Systemeinstellungen des XPath-Explorer

Auf der Systemeinstellungsseite **XPath-Explorer** können Sie Namespace-Präfix-Mappings angeben, die beim Generieren von XML-Fragmenten für die Option **XPath-zu-XML-Fragment** evaluieren verwendet werden sollen.

Um ein Mapping hinzuzufügen, klicken Sie auf **Neu** Das Feld **Namespace-Präfix-Mapping** hinzufügen wird angezeigt, in das Sie Werte für Präfix und Namespace für die erforderlichen Mappings eingeben können.

Um ein unerwünschtes Mapping zu entfernen, wählen Sie es aus und klicken Sie auf Entfernen. Um die Reihenfolge der Mappings neu anzuordnen, wählen Sie ein Mapping aus und klicken Sie je nach Bedarf auf Aufwärts oder Abwärts.

Referenz: Systemeinstellungen für Debug-Laufzeit für Assembly

Einstellung	Beschreibung
Hostname	Der Hostname für das Starten von Debug-Sitzungen.
HTTP Port	Die Portnummer, die Debug-Sitzungen mit Workday verbindet.
Neue Sitzungsverbindungsversuche	Gibt an, wie oft Studio versucht hat, Debug-Sitzungen mit Workday zu verbinden.
Maximale Warteiterationen bei Systemstart	Wie oft Studio prüft, ob Workday für den Empfang von Anforderungen bereit ist.
Alle Zertifikate akzeptieren	Gibt an, ob Studio nicht unterzeichnete Zertifikate akzeptiert.
HTTPS-Proxyeinstellungen für Debugging aktivieren	Aktiviert HTTPS-Proxyservereinstellungen für Debug-Sitzungen.
Proxyeinstellungen aus Eclipse kopieren	Gibt an, ob die Proxy-Einstellungen für Debug-Sitzungen aus Eclipse kopiert werden sollen.
Proxy-Host	Der Hostname des Proxyservers.
Proxy-Port	Der Port, den der Proxyserver überwacht.
Proxybenutzername	Der Benutzername für den Proxyserver.
Proxykennwort	Das Kennwort für den Proxyserver.
Timeout für Warten auf vorherige Geschäftsprozess-Schritte, z. B. Abruf (Sekunden)	Die Timeout-Periode, die Studio auf die zugehörigen Geschäftsprozessintegrationen alloziert, bevor Debug-Sitzungen wiederaufgenommen werden.

Referenz: Systemeinstellungen für Debug-Ansicht für Assembly

Einstellung	Beschreibung
Root-Teil von XML automatisch formatieren	Aktivieren Sie dieses Kontrollkästchen, wenn Studio automatisch die angezeigte XML-Datei für den Root-Teil der Nachricht formatieren soll.
Bring the Assembly Debug View to the front when a thread suspends	Aktivieren Sie dieses Kontrollkästchen, wenn Studio die Ansicht Assembly Debug in den Vordergrund bringen soll, wenn ein Thread während des Debug-Prozesses ausgesetzt wird.
Das Register „Ausnahme“ in den Vordergrund bringen, wenn ein Thread aufgrund einer Fehlerbehandlung ausgesetzt wird	Aktivieren Sie dieses Kontrollkästchen, wenn Studio das Register Ausnahme in der Ansicht Assembly Debug in den Vordergrund bringen soll, wenn ein Thread während des Debug-Prozesses ausgesetzt wird.

Einstellung	Beschreibung
Root-Teile mit weniger als (KB) automatisch abrufen	Gibt den Schwellenwert für das automatische Herunterladen an, d. h. die maximale Dateigröße für Root-Teile von Nachrichten, die in der Ansicht Assembly Debug automatisch angezeigt werden können. Um Root-Teile von Nachrichten anzuzeigen, die den Schwellenwert für das automatische Herunterladen überschreiten, müssen Sie die Nachricht herunterladen oder als Dokument im Arbeitsbereich oder Dateisystem speichern. Verwenden Sie dazu die Schaltflächen in der Symbolleiste der Ansicht „Assembly Debug“.
Truncate messages downloaded to the Assembly Debug View when lengths exceed (KB)	Nachrichten, die die angegebene Dateigröße überschreiten, werden abgeschnitten, wenn sie nach dem Herunterladen in die Ansicht Assembly Debug angezeigt werden.

Referenz: Systemeinstellungen des Cloud Explorer

Einstellung	Beschreibung
Beim Löschen einer Sammlung um Bestätigung bitten	Wenn dieses Feature aktiviert ist, werden Sie aufgefordert, Ihre Auswahl zu bestätigen, bevor Sie eine Sammlung aus der Cloud löschen können.
Anmerkungen im Cloud Explorer anzeigen	Wenn dieses Feature aktiviert ist, werden Anmerkungen zu Knoten im Cloud Explorer angezeigt.
Workday Connector-Inhalt anzeigen	Wenn dieses Feature aktiviert ist, zeigt der Cloud Explorer Inhalte für den workday - in-Transport wie Startparameter, Attribute und Maps an.

Referenz: Systemeinstellungen des Betrachters für konsolidierte Berichte

Spalte „Profilprotokoll“	Beschreibung
Aufrufe	Die Anzahl der Aufrufe, die die Komponente während des Integrationsereignisses erhält.
Durchschnittliche Zeit	Die durchschnittliche Verarbeitungszeit der Komponente.
Maximale Zeit	Die maximale Verarbeitungszeit der Komponente.
Minimale Zeit	Die minimale Verarbeitungszeit der Komponente.
Standardabweichungszeit	Die Standardabweichung von der durchschnittlichen Verarbeitungszeit der Komponente.
Zeit gesamt	Die Gesamtverarbeitungszeit der Komponente.
Durchschnittliche Anforderungsgröße	Die durchschnittliche Anforderungsgröße der Komponente.

Spalte „Profilprotokoll“	Beschreibung
Maximale Anforderungsgröße	Die maximale Anforderungsgröße der Komponente.
Minimale Anforderungsgröße	Die minimale Anforderungsgröße der Komponente.
Standardabweichung der Anforderungsgröße	Die Standardabweichung von der durchschnittlichen Anforderungsgröße der Komponente.
Anforderungsgröße gesamt	Die Gesamtanforderungsgröße der Komponente.
Durchschnittliche Antwortgröße	Die durchschnittliche Antwortgröße der Komponente.
Maximale Antwortgröße	Die maximale Antwortgröße der Komponente.
Minimale Antwortgröße	Die minimale Antwortgröße der Komponente.
Standardabweichung der Antwortgröße	Die Standardabweichung von der durchschnittlichen Antwortgröße der Komponente.
Antwortgröße gesamt	Die Gesamtantwortgröße der Komponente.
Maximale Speichernutzung	Die maximale Speichernutzung der Komponente.
Minimale Speichernutzung	Die minimale Speichernutzung der Komponente.
Durchschnittliche Speichernutzung	Die durchschnittliche Speichernutzung der Komponente.
Verfügbare Speichernutzung	Die verfügbare Speichernutzung der Komponente.

Referenz: Präferenzen für Java-Projekteinstellungen

Einstellung	Beschreibung
Enforce compatibility with only the current workbench default JRE	Standardmäßig ist jeder von Studio generierte Java-Code mit jeder installierten JRE kompatibel. Wenn Sie dieses Kontrollkästchen aktivieren, können Sie jedoch erzwingen, dass Java-Webservice-Projekte nur mit einer bestimmten JRE, d. h. der Workbench-Standard-JRE, kompatibel sind.
Respect export settings for classpath entries in referenced projects	Aktivieren Sie dieses Kontrollkästchen, um nur exportierte Ressourcen aus referenzierten Projekten einzuschließen, wenn Sie eine WSAR-Datei exportieren oder ein Webservice-Projekt bereitstellen.

Referenz: Benachrichtigungseinstellungen

Einstellung	Beschreibung
Benachrichtigungen aktivieren	Studio aktiviert diese Einstellung automatisch. Deaktivieren Sie dieses Kontrollkästchen, um die Anzeige von Benachrichtigungen auf dem Studio-Bildschirm zu deaktivieren.

Einstellung	Beschreibung
Ausblendverzögerung	Geben Sie einen Wert in Millisekunden ein, um anzugeben, wie lange Benachrichtigungen angezeigt werden sollen, bevor sie ausgeblendet werden.

Referenz: Systemeinstellungen des Prozessmonitors

Einstellung	Beschreibung
Automatische Aktualisierung aktivieren	Die automatische Aktualisierung der Ansicht Prozessmonitor ist standardmäßig aktiviert. Deaktivieren Sie das Kontrollkästchen, um dieses Feature zu deaktivieren.
Aktualisierungsintervall in Sekunden	Geben Sie einen Wert (in Sekunden) ein, um die Zeitspanne zwischen automatischen Aktualisierungen festzulegen. Verwenden Sie die Bildlaufschaltflächen auf der rechten Seite, um den Wert zu erhöhen oder zu verringern.
Anzahl der letzten aktiven Prozesse, die aktualisiert werden sollen	Geben Sie die Anzahl der aktiven Prozesse ein, die Sie aktualisieren möchten. Die aktualisierten Prozesse sind die aktuellen und vorherigen Prozesse in umgekehrter chronologischer Reihenfolge.
Fehlerdialogfeld anzeigen, wenn automatische Aktualisierung fehlschlägt	Aktivieren Sie die Anzeige einer Fehlermeldung, wenn das Feature der automatischen Aktualisierung fehlschlägt. Studio aktiviert dieses Feature automatisch. Deaktivieren Sie das Kontrollkästchen, um dieses Feature zu deaktivieren.
Ansicht nach dem Start einer Integration anzeigen	Workday zeigt den Prozessmonitor im Vordergrund an, wenn Sie eine Integration starten. Deaktivieren Sie das Kontrollkästchen, um dieses Feature zu deaktivieren.

Referenz: Systemeinstellungen des Webservice-Testers

Einstellung	Beschreibung
Request and response XML display	Gibt Schriftgrad und -stil zur Anzeige der XML-Daten für Anforderung und Antwort an. Der geöffnete Editor wird sofort aktualisiert, wenn Sie die Schriftdetails ändern.
Do not display messages larger than:	Gibt die maximale Größe von Nachrichten an, die im Bereich Antwort des Webservice-Testers angezeigt werden sollen. Der Standardwert ist 1024 KB. Jede Nachricht, die größer ist als der angegebene Wert, wird als Anhang gespeichert. Sie können den Anhang lokal speichern.

Studio-Aktualisierungen

Studio-Aktualisierung

Warum und wann dieser Vorgang ausgeführt wird

So wie es nur eine Version der Workday-Anwendung gibt, die von allen Kunden verwendet wird, gibt es immer nur eine gültige Version von Workday Studio. Ihre Studio-Installation muss aktuell sein, um mit Ihrer Sandbox- oder Produktionsumgebung zu interagieren.

Prozedur

1. Wenn eine Aktualisierung verfügbar ist, benachrichtigt Studio Sie, wenn Sie Ihre Mandanten-Zugangsdaten eingeben – entweder beim Start oder später. Bestätigen Sie, dass Sie eine Aktualisierung durchführen möchten, und führen Sie dann die Schritte im Assistenten Workday Studio aktualisieren aus.

Anmerkung: Aktualisieren Sie die zugrunde liegende Eclipse-Plattform nicht unabhängig vom Aktualisierungszeitplan von Studio. Dies könnte dazu führen, dass Ihre Integrationen fehlschlagen.

2. Wenn Sie die Aktualisierung nicht sofort durchführen möchten, zeigt Studio als Erinnerung einen roten Pfeil in der Statusleiste an. Sie können Assemblys auch dann noch lokal entwickeln, wenn eine Aktualisierung nicht installiert ist. Sie können jedoch keine Verbindung zu Ihren Sandbox- oder Produktionsmandanten herstellen. Um später zu aktualisieren, wählen Sie Workday > - Workday Studio aktualisieren aus.

Anmerkung: Studio enthält eine vereinfachte Version der Workday-Laufzeit, mit der Sie Ihre Integrationen testen und debuggen können. Um sicherzustellen, dass Sie immer Zugriff auf dieselbe Laufzeit haben, die in Ihrem Produktionsmandanten verfügbar ist, sucht Studio beim Start nach einer Aktualisierung der Laufzeit. Wenn ein Element verfügbar ist, wird es im Hintergrund heruntergeladen. Um dieses Verhalten anzupassen, wählen Sie Fenster > - Systemeinstellungen > Laufzeit > - Automatische Aktualisierungen.

Laufzeitänderungen für Workday-Assembly

Konzept: Assembly-Versionen

Mithilfe des Versionsmanagements für Assemblys können der ESB-Server (Workday Enterprise Service Bus) und die Studio-Entwicklungsteams im Laufe der Zeit Assemblys innerhalb des Workday Runtime-Programmiermodells weiterentwickeln, ohne vorhandene Benutzer zu beeinträchtigen. Mit dem Versionsmanagement für Assemblys kann Workday beispielsweise Folgendes ausführen:

- Features als veraltet markieren
- Neue Features vorstellen
- Ändern Sie die Semantik vorhandener Features

Alle Baugruppen haben eine `version` Attribut. Wenn Sie ein neues Baugruppenprojekt erstellen, legt Workday Studio standardmäßig fest `version` Attributwert für `assembly` -Element in der Datei `Montag.xml` auf die aktuellste Version der Workday-Laufzeit. Das Format ist `year . drop - number`, wobei Dropdown-Nummer der Anzahl der Woche entspricht. Beispiel: `2016 . 11`.

Studio zeigt diese Zahl neben dem Ordner „Zusammenfassung“ im Projekt-Explorer an.

Wenn Änderungen an vorhandenen Runtime-Features vorgenommen werden, stellt Workday sicher, dass diese mit den vorhandenen Features abwärtskompatibel sind. Dies bedeutet, dass alle vorhandenen Assemblys gemäß ihrer Versionsnummer wie bisher funktionieren. Neue Assemblys haben jedoch standardmäßig das Verhalten, das der aktuellen Versionsnummer entspricht. Wenn Workday eine

Änderung einführt, wird in der Regel ein neues Attribut für einen Assembly-Schritt oder eine Komponente definiert. Ältere Assemblys verfügen nicht über dieses Attribut. Daher wird eine Regel eingeführt, die auf der Versionsnummer basiert, um den Wert für ältere Assemblys als Standard vorzugeben. Außerdem wird eine Regel eingeführt, die für neuere Assemblys einen anderen Wert vorgibt. Wenn Sie als Entwickler neue Features in ältere Assemblys übernehmen möchten, haben Sie mehrere Optionen. Sie können:

- Die ältere Version beibehalten und die von der Workday Runtime zugewiesene Standardvorgabe akzeptieren.
- Die ältere Version beibehalten und je nach Ihren Anforderungen einen expliziten Wert für das neue Attribut festlegen.
- Zu einer neueren Version migrieren und die neue Standardvorgabe akzeptieren.
- Zu einer neueren Version migrieren und einen expliziten Wert für das neue Attribut festlegen.

Referenz: Versionsänderungen auf Schemaebene

Diese Tabelle fasst die Änderungen auf Schemaebene zusammen, die für jede Versionsaktualisierung vorgenommen wurden.

Version	Komponente	Attribut/Eigenschaft	Beschreibung der Änderung
8.0	Komponenten „async-mediation“ und „sync-mediation“	treat-downStream-errors	Standard geändert von true bis false Ein Wert von false bedeutet, dass lokale Fehlerhandler in dieser Komponente nur Fehler behandeln, die in den Schritten dieser Komponente auftreten. Ein Wert von true bedeutet, dass lokale Fehlerhandler in dieser Komponente nachgelagerte Fehler verarbeiten können.
9.0	write	output-mime-type	Standard geändert von text/plain bis text/xml, in Übereinstimmung mit anderen Schritten.
11,0	xmldiff-Schritt	copy-to-output	Standard wurde in geändert true. Ein Wert von true bedeutet, dass der Schritt die Ausgabe in die Vermittlungsnachricht kopiert. Ein Wert von false bedeutet, dass der Schritt die Ausgabe in eine Variable kopiert.

Version	Komponente	Attribut/Eigenschaft	Beschreibung der Änderung
12,0	Transport „http-in“	sichern	Standard geändert von false bis null, was bedeutet, dass der Endpunkt über HTTP und HTTPS verfügbar ist, sofern nicht anders angegeben.
	Http-out-Transport	Fehler als Antwort	Standard geändert von false bis true. Ein Wert von true bedeutet, dass, wenn die Laufzeit einen Fehler beim Transport findet, der Fehler in der aktuellen Meldung festgelegt wird. Bisher war der Fehler nur über verfügbar BadRequestException auf der MediationContext.
	Transporte local-out, http-out, ftp-out, ftps-out, sftp-out, xmpp-out, custom-out, email-out	ignore-dynamic-endpoints	Standard geändert von false bis true. Ein Wert von true bedeutet, dass der Transport den WSA-Wert (Dynamische WS-Adressierung) im Header der eingehenden Nachricht ignoriert. Ein Wert von false bedeutet, dass der Transport den WSA-Wert verwendet, um den Wert in seiner Eigenschaft "Endpoint" zu überschreiben.
13,0	store	schema	Standard geändert von urn:com.workday/bsvc/blob bis [http://www.w3.org/2005/Atom]. Die Unterdomäne PutIntegrationMessage kann den Atom direkt nutzen, wenn Dokumente

Version	Komponente	Attribut/Eigenschaft	Beschreibung der Änderung
			zum aktuellen Integrationsereignis hinzugefügt werden.
	Komponenten „async-mediation“ und „sync-mediation“	fortsetzen nach Fehler	Standard geändert von recover bis rewind. Ein Wert von rewind bedeutet, dass die Verarbeitung von der vorherigen Komponente in der Baugruppe aus fortgesetzt wird, wenn ein Fehler behoben wird. Der vorherige Standard recover, ist jetzt veraltet.
	Lokal - Ausgangsverkehr	unset-properties	Standard geändert von false bis true. Ein Wert von true bedeutet, dass die Eigenschaften des Transports zurückgesetzt werden, nachdem er eine Untergruppe aufgerufen hat.
	Komponente "on-error" für Fehlerhandler	rethrow-error	Standard geändert von true bis false. Ein Wert von false bedeutet, dass die Laufzeit den Fehler löscht, wenn der Fehlerhandler ausgeführt wird. Ein Wert von true bedeutet, dass der Fehlerhandler den Fehler selbst löschen muss.
14,0	workday-out-soap	replace-with-soap-fault	Standard geändert von false bis true. Ein Wert von true bedeutet, dass der Transport die aktuelle Meldung durch den SOAP-Fehler ersetzt,

Version	Komponente	Attribut/Eigenschaft	Beschreibung der Änderung
			wenn eine Meldung zurückgegeben wird.
	local-in	Zugriff auf	Standard geändert von public bis private. Ein Wert von private bedeutet nur a local-out in derselben Baugruppe kann den Transport aufrufen. Ein Wert von public bedeutet, dass jede Build den Transport aufrufen kann.
	Die message-content-collater-Strategie der Aggregatorkomponente.	output-mimetype	Standard geändert von application/octet-stream bis text/plain. Benutzerdefinierte Sortierer, die Folgendes implementieren AggregationCollater Für die Schnittstelle ist nach wie vor Standard vorgegeben application/octet-stream.
18,0	store	unveränderlich	Neue Eigenschaft, die angibt, ob die Binärdaten, auf die der Blob verweist, direkt geändert werden können. Standardvorgabe: true. Ein Wert von true bedeutet, dass der BLOB unveränderlich ist und nicht geändert werden kann. Ein Wert von false bedeutet, dass der Blob veränderbar ist und geändert werden kann. Anmerkung: In Workday 18 migriert die Serverlaufzeit alle Speicherschritte

Version	Komponente	Attribut/Eigenschaft	Beschreibung der Änderung
			in vorhandenen Builds mit alten Versionsnummern, um diese neue Standardeinstellung zu verwenden, die Blobs unveränderlich macht. Wenn Ihre Baugruppe einen Blob aktualisieren muss, stellen Sie sicher, dass Sie „unveränderbar“ auf festlegen false.
2016.45 Anmerkung: Nach Version 27 hat Workday ein neues Namensschema im Formular eingeführt year . week	PagedGet	is.paged.get.version	Standard geändert von util.assemblyVersionAsWSVer bis null.
2019.30	PagedGet	As_Of_Effective_Date As_Of_Entry_DateTime	Wenn für diese Eigenschaften keine Werte angegeben werden, werden standardmäßig das aktuelle Datum und die aktuelle Uhrzeit (PST) verwendet.
2020.09	ftp-out, ftps-out, sftp-out		Workday verwendet jetzt die JSCAPE-Version 9.3, die einen breiteren Satz von Verschlüsselungscodes unterstützt.
2020.09	PrismAnalytics	wd.prism.component.api.hostname	Studio nutzt jetzt v2 der REST-API von Prism Analytics auf. Der Komponentename wurde von PrismConnector in PrismAnalytics geändert.

[relatedlinks](#)

Referenz

[FAQ: Verschlüsselung, Zertifikate und Chiffren für Integrationen](#)

Referenz: Versionsänderungen auf XML-Ebene

Die folgende Tabelle fasst die Änderungen auf XML-Ebene zusammen, die für jede Versionsaktualisierung vorgenommen wurden.

Version	Vorheriges XML-Element/-Attribut	Ersetzt durch	Bedeutung/Grund der Änderung
11.0 (Diese Änderungen betreffen alle Änderungen an Integrationssystemdefinitionen in der Workday-in-Transportkomponente.)	simple-type-reference	simple-type	Die Art des Postens wird deklariert, kein Bezug.
	type-reference	class-report-field	Der neue Elementname gibt die Bedeutung besser wieder.
	service	service-reference	Das Element "" wird nicht definiert, es wird auf das Element verwiesen. Entfernt defined-by Attribut, das nicht mehr verwendet wird.
	integration-system/@template	Entfernt	Nicht mehr zutreffend.
	param/@required	param/@option	Booleschen Wert geändert required Attribut zu einem option Attribut mit dem Wert required, wobei wahr ist.
	cloud/name	cloud/@name	Weniger ausführlich.
	integration/name	in-connector/@id	Weniger ausführlich.
	in-connector/id	private-key	private-key
12,0	key-file	private-key	Hat Auswirkungen sftp-out und ftps-out Transporte. Das Attribut wurde umbenannt, um die Bedeutung besser wiederzugeben.
	public-key-properties	private-key-properties	Umbenannt, um die Bedeutung besser wiederzugeben.
	aggregator/@spawn-aggregated-processing	Entfernt	Mit diesem Feature konnte ein Aggregator eine neue Aufgabe oder einen neuen Thread erstellen,

Version	Vorheriges XML-Element/-Attribut	Ersetzt durch	Bedeutung/Grund der Änderung
			um eine aggregierte Nachricht zu bearbeiten. Diese Funktionalität wurde in Workday-Integrationen nicht verwendet und war für das öffentliche Workday Studio nicht geeignet. Daher wurde sie entfernt.
	store/@compress store/@encrypt	Entfernt	Zuvor war eine detaillierte Steuerung über den Schritt „Speichern“ möglich. Die Workday-Richtlinie für Integrationen lautet jedoch, immer zu werden und zu verschlüsseln. Daher wurden sie entfernt.
	ftp-out/@tmp-directory sftp-out/@tmp-directory ftps-out/@tmp-directory	Entfernt	Zuvor verwalteten diese Transporte ihre eigenen temporären Dateien, wurden jedoch umgestaltet, um die allgemeine FBDM-Funktion (FileBackedManagedData) zu verwenden, sodass diese Optionen entfernt wurden.
14,0	delivery-service/ child::*	Entfernt	Unterelemente des Lieferservices definierten die zulässigen Transporte. Solche Beschränkungen sollten nicht zur Designzeit definiert werden. Da es sich hierbei um eine Laufzeitkonfigurationsaufgabe handelt, wurden sie aus der Entwicklungsphase zur Designzeit entfernt.

Referenz: Änderungen des Saxon-XSLT-Prozessors

In der folgenden Tabelle ist die Workday-Unterstützung für verschiedene Versionen von Saxon und XSLT zusammengefasst. Beachten Sie, dass Workday nach Version 27 ein neues Namensschema im Formular eingeführt hat `year . week`.

Workday-Version	Unterstützte Saxon-Version	Unterstützte XSLT-Version	Anmerkungen
<20	9.1	1.0, (2.0)	Eingeschränkte Unterstützung für XSLT-Version 2.0.
20-2017.05	9.4	2.0, (3.0)	Eingeschränkte Unterstützung für XSLT-Version 3.0.
2017.05-	9.7	3.0	Unterstützung für XSLT-Streaming.

Anmerkung: Studio zeigt eine Warnmeldung an, wenn Sie eine Sammlung erstellen, die Assemblys mit unterschiedlichen Versionsnummern enthält. Dies liegt daran, dass Assemblys unterschiedlicher Versionen widersprüchliche Versionen von Saxon erfordern können, was zur Laufzeit zu Fehlern beim Laden von Klassen führen kann. Best Practice ist es, für alle Integrationen die neuesten Assembly-Versionen zu verwenden.

Bitte beachten Sie Folgendes:

- Die XPath-Methoden, die in MVE für den Zugriff auf Nachrichtenteile und Variablen verfügbar sind, unterstützen XPath 3.0.
- Die `mctx` Das URL-Protokoll ermöglicht es XSL-Entwicklern, in Nachrichtenteile und Variablen zu schreiben, z. B.:

```
<xsl:result-document href="mctx:vars/var1">...</xsl:result-document>
```

- Obwohl es für XQuery keine Unterstützung für direktes Build gibt, können Sie benutzerdefinierte Schritte zum Aufrufen von XQuery-Skripten hinzufügen.
- In Baugruppen mit Version 2022.46 oder früher wird Sachsen nur dann als Standardtransformator festgelegt, wenn `use.saxon.transformer`-Eigenschaft ist auf festgelegt `true` im Kontext der Vermittlung. In Baugruppen mit Version 2022.47 oder höher wird Sachsen als Standard festgelegt, wenn diese Eigenschaft auf festgelegt ist `true` oder nicht vorhanden.

Anmerkung: XSLT 2.0 und XSLT 3.0 sind stark modelliert. Beispielsweise erlauben sie nicht `xs:string`, die in XPath-Ausdrücken auf Gleichheit evaluiert werden soll. Der folgende Ausdruck, in `which stringParam ""` ist ein XSL-Parameter und führt zu einem Sachsen-Laufzeitfehler:

```
$stringParam = true()
```

Der XSL-Compiler wird das Problem zum Zeitpunkt der Bereitstellung nicht feststellen, da er nicht im Voraus zulassen kann, welchen Typ der Parameter haben wird.

Ihre vorhandenen XSL-Dokumente und Workday Studio-Integrationen sind von diesem Problem mit schwerwiegenden Fehlern nicht betroffen, es sei denn, Sie aktivieren XSLT 3.0, indem Sie den Versionsattributwert in Ihren XSL-Dokumenten in ändern 3.0. Wenn diese Laufzeitfehler auftreten, die aus der mit Ihrer Integration bereitgestellten XSLT resultieren, müssen Sie Maßnahmen ergreifen, um sie zu beheben.

Anmerkung: Folgender Fehler im Zusammenhang mit dem Sachsen-Upgrade wurde in der Legacy-XSL für den Druck von Geschäftsformularen beobachtet:

```
Failed to compile stylesheet.Xerrors detected. Invalid value for @case-order. Value must be one of (lower-first|upper-first).
```

Wenn dieser Fehler auftritt, prüfen Sie, ob Ihre XSL die folgende Attributeinstellung für ein Sortierelement enthält:

```
case-order="#default"
```

Dies ist optional case-order. Das Attribut "" bestimmt, ob bei der Sortierung Groß- oder Kleinbuchstaben zuerst aufgelistet werden. Standardmäßig wird Großbuchstaben zuerst aufgelistet. Der Sachsen-Parser akzeptiert nur folgende Werte für die Reihenfolge der Groß-/Kleinschreibung:

- upper-first
- lower-first

Um diesen Fehler zu beheben, haben Sie folgende Möglichkeiten:

- Entfernen Sie das vorhandene Attribut für die Groß-/Kleinschreibung aus dem Sortierelement, da dies eine optionale Einstellung ist.
- Legen Sie den Wert für die Fallreihenfolge explizit auf fest upper-first.
- Legen Sie den Wert für die Fallreihenfolge explizit auf fest lower-first.

Beispielkonfigurationen

Beispiel: Beispiel-Assembly „Hello Cloud“ ausführen

Warum und wann dieser Vorgang ausgeführt wird

Mit Workday Studio können Sie Integrationen in Workday bereitstellen und starten. Dieses Beispiel zeigt, wie Sie eine einfache Beispielintegration ausführen können.

Vorbereitungen

Sicherheit: Folgende Domänen im funktionalen Bereich „Integration“:

- *Integration Build*
- *Integration Configure*
- *Integration Event*

Prozedur

1. Wählen Sie in der Symbolleiste für Studio die Option Hilfe > - > Willkommensbeispiele > für Hallo Cloud.
2. Klicken Sie im Fenster Projektnamen auf Fertig stellen.
3. Blenden Sie in der Ansicht Project Explorer den Ordner HelloCloud ein und doppelklicken Sie auf den Ordner Assembly.
4. Klicken Sie in der Symbolleiste des Assembly-Editors auf Anmerkungen anzeigen.
5. Klicken Sie in der Ansicht Cloud Explorer auf Verbindungsdetails.
6. Wählen Sie die Sandbox-Umgebung aus und geben Sie die folgenden Werte in das Fenster Verbindungen ein:

Option	Bezeichnung
Mandant	Der Name Ihres Mandanten.
Benutzername	Der Benutzername der Zugangsdaten für Ihren Mandanten.

Option	Bezeichnung
Kennwort	Das Kennwort der Zugangsdaten für Ihren Mandanten.

7. Klicken Sie auf Anwenden und schließen.
8. Klicken Sie in der Ansicht Project Explorer mit der rechten Maustaste auf den Ordner HelloCloud und wählen Sie Deploy to Workday aus.
9. Wählen Sie im Fenster Deploy to Workday die Sandbox-Umgebung aus. Klicken Sie auf Finish.
10. Blenden Sie in der Cloud-Explorer- Ansicht die folgenden Knoten ein: Sandbox > HalloCloudSammlung > HalloCloud.
11. Klicken Sie mit der rechten Maustaste auf den Integrationsknoten StartHere und wählen Sie Integration starten aus.
12. Klicken Sie auf Starten.

Ergebnisse

Die Integration wird in Ihrem Workday-Mandanten gestartet. Die Ansicht Prozessmonitor in Studio zeigt den Status der Integration an.

Nächste Maßnahme

Melden Sie sich bei Workday an und öffnen Sie die Aufgabe Prozessmonitor, um die Details des Integrationsereignisses für die Beispiel-Assembly anzuzeigen.

Beispiel: Beispiel-Assembly „Hello Workday Web Services“ ausführen

Warum und wann dieser Vorgang ausgeführt wird

Mit Workday Studio können Sie eine Verbindung zu Workday-Webservices herstellen. Dieses Beispiel zeigt die SOAP-Webservice-Anforderungsmeldung Get_Workers_Request. Zur Verarbeitung der Anforderung wird XSLT 3.0-Streaming verwendet.

Vorbereitungen

Sicherheit: Folgende Domänen im funktionalen Bereich „Integration“:

- *Integration Build*
- *Integration Configure*
- *Integration Event*

Prozedur

1. Wählen Sie in der Studio-Symbolleiste die Optionen Hilfe > Willkommen > Beispiele > Hello Workday Webservices.
2. Klicken Sie im Fenster Projektnamen auf Beenden.
3. Blenden Sie in der Ansicht Project Explorer den Ordner HelloWWS ein und doppelklicken Sie auf den Ordner Assembly.
4. Klicken Sie in der Symbolleiste des Assembly-Editors auf Anmerkungen anzeigen.
5. Klicken Sie in der Ansicht Cloud Explorer auf Verbindungsdetails.

6. Wählen Sie die Sandbox-Umgebung aus und geben Sie die folgenden Werte in das Fenster Verbindungen ein:

Option	Bezeichnung
Mandant	Der Name Ihres Mandanten.
Benutzername	Der Benutzername der Zugangsdaten für Ihren Mandanten.
Kennwort	Das Kennwort der Zugangsdaten für Ihren Mandanten.

7. Klicken Sie auf Anwenden und schließen.
8. Klicken Sie in der Ansicht Project Explorer mit der rechten Maustaste auf den Ordner HelloWWS und wählen Sie Deploy to Workday aus.
9. Wählen Sie im Fenster Deploy to Workday die Sandbox-Umgebung aus und klicken Sie auf Beenden.
10. Blenden Sie diese Knoten in der Ansicht Cloud Explorer ein: Sandbox > HelloWWSCollection > HelloWWS.
11. Klicken Sie mit der rechten Maustaste auf den Integrationsknoten HelloWWS_EntryPoint und wählen Sie Integration starten aus.
12. Klicken Sie auf Starten.

Ergebnisse

Die Integration wird in Ihrem Workday-Mandanten gestartet. Die Ansicht Prozessmonitor in Studio zeigt den Status der Integration an.

Nächste Maßnahme

Melden Sie sich bei Workday an und öffnen Sie die Aufgabe Prozessmonitor, um die Details des Integrationsereignisses für die Beispiel-Assembly anzuzeigen.

Beispiel: Beispiel-Assembly „Common Usage Patterns“ ausführen

Warum und wann dieser Vorgang ausgeführt wird

Dieses Beispiel zeigt einige gängige Integrationsdesignmuster. Es enthält mehrere separate Verarbeitungsketten, von denen jede ein anderes Muster veranschaulicht.

Vorbereitungen

Sicherheit: Folgende Domänen im funktionalen Bereich „Integration“:

- *Integration Build*
- *Integration Configure*
- *Integration Event*

Prozedur

1. Wählen Sie in der Studio-Symbolleiste Hilfe > Willkommen > Beispiele > Common Usage Patterns aus.
2. Klicken Sie im Fenster Projektnamen auf Beenden.
3. Blenden Sie in der Ansicht Project Explorer den Ordner CommonUsagePatterns ein und doppelklicken Sie auf den Ordner Assembly.

4. Klicken Sie in der Symbolleiste des Assembly-Editors auf Anmerkungen anzeigen.
5. Klicken Sie in der Ansicht Cloud Explorer auf Verbindungsdetails.
6. Wählen Sie die Sandbox-Umgebung aus und geben Sie die folgenden Werte in das Fenster Verbindungen ein:

Option	Bezeichnung
Mandant	Der Name Ihres Mandanten.
Benutzername	Der Benutzername der Zugangsdaten für Ihren Mandanten.
Kennwort	Das Kennwort der Zugangsdaten für Ihren Mandanten.

7. Klicken Sie auf Anwenden und schließen.
8. Klicken Sie in der Ansicht Project Explorer mit der rechten Maustaste auf den Ordner CommonUsagePatterns und wählen Sie Deploy to Workday.
9. Wählen Sie im Fenster In Workday bereitstellen die Sandbox-Umgebung aus und klicken Sie auf Beenden.
10. Erweitern Sie in der Ansicht Cloud Explorer die folgenden Knoten: Sandbox > CommonUsagePatterns > CommonUsagePatterns.
11. Klicken Sie mit der rechten Maustaste auf den Integrationsknoten Conditional-Processing und wählen Sie Integration starten aus.
12. Klicken Sie auf Starten.
13. Starten Sie jeden der folgenden Integrationsknoten:
 - Enriching
 - Global-Error-Handling
 - Local-Error-Handling
 - Looping
 - Request-Response-Processing-Chain
 - Response-Chaining
 - Routing
 - Scalable-Message-Processing
 - Simple-Processing-Chain
 - SubRoutines
 - Variablen

Die Integrationen Request-Response-Processing-Chain und Response-Chaining haben einen zusätzlichen Startparameter: Employee_ID. Klicken Sie auf Literalwert auswählen, um ihn zu konfigurieren.

Ergebnisse

Die Integration wird in Ihrem Workday-Mandanten gestartet. Die Ansicht Prozessmonitor in Studio zeigt den Status der Integration an.

Nächste Maßnahme

Melden Sie sich bei Workday an und öffnen Sie die Aufgabe Prozessmonitor, um die Details des Integrationsereignisses für die Beispiel-Assembly anzuzeigen.

Beispiel: Beispiel-Assembly „Debugging“ ausführen

Warum und wann dieser Vorgang ausgeführt wird

Mit Workday Studio können Sie Fehler in Ihren Integrationen erkennen und debuggen. Dieses Beispiel zeigt, wie Sie eine Beispielintegration debuggen.

Vorbereitungen

Sicherheit: Folgende Domänen im funktionalen Bereich „Integration“:

- *Integration Build*
- *Integration Configure*
- *Integration Debug*
- *Integration Event*

Prozedur

1. Wählen Sie in der Studio-Symbolleiste die Optionen Hilfe > Willkommen > Samples > Assembly Debugger.
2. Klicken Sie im Fenster Projektnamen auf Beenden.
3. Erweitern Sie in der Ansicht Project Explorer den Ordner AssemblyDebug und doppelklicken Sie auf den Ordner Assembly.
4. Klicken Sie im Assembly-Editor mit der rechten Maustaste auf jedes der folgenden Assembly-Elemente und wählen Sie Toggle Assembly Breakpoint:
 - create-get-workers-request
 - store-summary-doc
5. Klicken Sie in der Ansicht Cloud Explorer auf Verbindungsdetails.
6. Wählen Sie die Sandbox-Umgebung aus und geben Sie die folgenden Werte in das Fenster Verbindungen ein:

Option	Bezeichnung
Mandant	Der Name Ihres Mandanten.
Benutzername	Der Benutzername der Zugangsdaten für Ihren Mandanten.
Kennwort	Das Kennwort der Zugangsdaten für Ihren Mandanten.

7. Klicken Sie auf Anwenden und schließen.
8. Klicken Sie in der Ansicht Project Explorer mit der rechten Maustaste auf den Ordner AssemblyDebug und wählen Sie Deploy to Workday.
9. Wählen Sie im Fenster Deploy to Workday die Sandbox-Umgebung aus und klicken Sie auf Beenden.
10. Erweitern Sie in der Ansicht Cloud Explorer die folgenden Knoten: Sandbox > AssemblyDebug > AssemblyDebug.
11. Klicken Sie mit der rechten Maustaste auf den Integrationsknoten Start-Here und wählen Sie Debug Integration.
12. Klicken Sie auf Debug.
13. Klicken Sie im Fenster Confirm Perspective Switch auf Ja.
14. Wählen Sie in der Ansicht Assembly Debug das Register Message Root Part und klicken Sie auf Save the message or attachment to a document in the workspace.

15. Geben Sie im Fenster Speichern folgende Werte ein:

Option	Bezeichnung
Übergeordneter Ordner	AssemblyDebug
Dateiname	rootpart.xml

16. Klicken Sie in der Ansicht Debug mit der rechten Maustaste auf den Breakpoint Create-Request - create-get-workers-request und wählen Sie Fortsetzen.

17. Wählen Sie in der Ansicht Assembly Debug das Register Message Root Part und klicken Sie auf Save the message or attachment to a document in the workspace.

18. Geben Sie im Fenster Speichern folgende Werte ein:

Option	Bezeichnung
Übergeordneter Ordner	AssemblyDebug
Dateiname	rootpart1.xml

19. Klicken Sie in der Ansicht Debug mit der rechten Maustaste auf den Breakpoint Store - store-summary-doc und wählen Sie Beenden.

Ergebnisse

Studio speichert die XML-Breakpoint-Dateien im Projektordner AssemblyDebug. Sie können diese Dateien verwenden, um Fehler in der Assembly zu beheben.

Nächste Maßnahme

Öffnen Sie die XML-Breakpoint-Dateien für die Assembly.

Beispiel: Beispiel-Assembly „Workday-In Features“ ausführen

Warum und wann dieser Vorgang ausgeführt wird

Die Transportkomponente workday-in wird in Workday als Integrationssystem dargestellt. Dieses Beispiel zeigt, wie Sie MVEL-Ausdrücke in einer Workday Studio-Assembly verwenden können, um auf die Services und Startparameter des Integrationssystems zu verweisen. Es enthält drei separate Verarbeitungsketten.

Vorbereitungen

Sicherheit: Folgende Domänen im funktionalen Bereich „Integration“:

- *Integration Build*
- *Integration Configure*
- *Integration Event*

Prozedur

1. Wählen Sie in der Symbolleiste von Studio die Option Help > Willkommen bei > Samples > in Workday.
2. Klicken Sie im Fenster Projektnamen auf Fertig stellen.
3. Blenden Sie in der Ansicht Project Explorer den Ordner WorkdayInFeatures ein und doppelklicken Sie auf den Ordner Assembly.
4. Klicken Sie in der Symbolleiste des Assembly-Editors auf Anmerkungen anzeigen.
5. Klicken Sie in der Ansicht Cloud Explorer auf Verbindungsdetails.

6. Wählen Sie die Sandbox-Umgebung aus und geben Sie die folgenden Werte in das Fenster Verbindungen ein:

Option	Bezeichnung
Mandant	Der Name Ihres Mandanten.
Benutzername	Der Benutzername der Zugangsdaten für Ihren Mandanten.
Kennwort	Das Kennwort der Zugangsdaten für Ihren Mandanten.

7. Klicken Sie auf Anwenden und schließen.
8. Klicken Sie in der Ansicht Projekt Explorer mit der rechten Maustaste auf den Ordner WorkdayInFeatures und wählen Sie In Workday bereitstellen aus.
9. Wählen Sie im Fenster Deploy to Workday die Sandbox-Umgebung aus und klicken Sie auf Beenden.
10. Für jede der einzelnen Integrationen sind folgende Informationen zu beachten:

Integration	Anmerkung
AttributesAndMaps	Nach der Bereitstellung können Sie die Attribut- und Mapping-Werte in Workday anpassen. - Öffnen Sie den Bericht View Integration System und wählen Sie <i>WorkdayInFeatures-attr-maps</i>. - In den möglichen Aktionen für den Integrationsservice ExampleAttrMapService wählen Sie Integration Generic Service > Configure Integration Attributes oder Integration Generic Service > Configure Integration Maps. - Passen Sie die Attribut- und Map-Werte nach Bedarf an.
LaunchParameters	Erweitern Sie in der Cloud-Explorer -Ansicht in Studio die folgenden Knoten: Sandbox > WorkdayInFeatures > WorkdayInFeatures. - Klicken Sie im Feld ReferenceParam auf Literalwerte auswählen und wählen Sie einen Mitarbeiter aus. - Geben Sie ein password im Feld passwordParam .
SequenceGenerator	Nach der Bereitstellung können Sie den Sequenzgenerator-Service in Workday wie folgt konfigurieren: - Öffnen Sie den Bericht View Integration System und wählen Sie <i>WorkdayInFeatures-sequences</i>. - In den möglichen Aktionen für den Integrationsservice ExampleSeqGenService wählen Sie

Integration	Anmerkung
	Integration Sequence Generator Service > Configure Integration Sequence Generators. c. Geben Sie im Feld Erhöhen um Folgendes ein: 1. d. Geben Sie im Feld Format/Syntax Folgendes ein: <code>file[Seq].csv</code>.

11. Erweitern Sie in der Ansicht „Cloud Explorer“ die folgenden Knoten: Sandbox > WorkdayInFeatures > WorkdayInFeatures.

12. Klicken Sie mit der rechten Maustaste auf einen Workday-In-Transportknoten und wählen Sie Integration starten. Beispiel: AttributesAndMaps.

13. Klicken Sie auf Starten.

Ergebnisse

Die Integration wird in Ihrem Workday-Mandanten gestartet. Die Ansicht Prozessmonitor in Studio zeigt den Status der Integration an.

Nächste Maßnahme

Melden Sie sich bei Workday an und öffnen Sie die Aufgabe Prozessmonitor, um die Details des Integrationsereignisses für die Beispiel-Assembly anzuzeigen.

Beispiel: Beispiel-Assembly „Error Handling“ ausführen

Warum und wann dieser Vorgang ausgeführt wird

Dieses Beispiel zeigt verschiedene Ansätze zur Fehlerbehandlung in Workday Studio. Es enthält sechs verschiedene Verarbeitungsketten, von denen jede ein anderes Muster veranschaulicht.

Vorbereitungen

Sicherheit: Folgende Domänen im funktionalen Bereich „Integration“:

- *Integration Build*
- *Integration Configure*
- *Integration Event*

Prozedur

1. Wählen Sie in der Studio-Symbolleiste Hilfe > Willkommen > Beispiele > Fehlerbehandlung aus.
2. Klicken Sie im Fenster Projektnamen auf Beenden.
3. Blenden Sie in der Ansicht Project Explorer den Ordner ErrorHandler101 ein und klicken Sie doppelt auf den Ordner Assembly.
4. Klicken Sie in der Symbolleiste des Assembly-Editors auf Anmerkungen anzeigen.
5. Klicken Sie in der Ansicht Cloud Explorer auf Verbindungsdetails.
6. Wählen Sie die Sandbox-Umgebung aus und geben Sie die folgenden Werte in das Fenster Verbindungen ein:

Option	Bezeichnung
Mandant	Der Name Ihres Mandanten.

Option	Bezeichnung
Benutzername	Der Benutzername der Zugangsdaten für Ihren Mandanten.
Kennwort	Das Kennwort der Zugangsdaten für Ihren Mandanten.

7. Klicken Sie auf Anwenden und schließen.
8. Klicken Sie in der Ansicht Project Explorer mit der rechten Maustaste auf den Ordner ErrorHandler und wählen Sie Deploy to Workday aus.
9. Wählen Sie im Fenster Deploy to Workday die Sandbox-Umgebung aus und klicken Sie auf Beenden.
10. Blenden Sie diese Knoten in der Ansicht Cloud Explorer ein: Sandbox > -ErrorHandler > ErrorHandler.
11. Klicken Sie mit der rechten Maustaste auf einen Transportknoten von Workday-in und wählen Sie Integration starten aus. Beispiel: All-Or-Nothing.
12. Klicken Sie auf Starten.

Ergebnisse

Die Integration wird in Ihrem Workday-Mandanten gestartet. Die Ansicht Prozessmonitor in Studio zeigt den Status der Integration an.

Nächste Maßnahme

Melden Sie sich bei Workday an und öffnen Sie die Aufgabe Prozessmonitor, um die Details des Integrationsereignisses für die Beispiel-Assembly anzuzeigen.

Beispiel: Beispiel-Tests „AUnit“ ausführen

Warum und wann dieser Vorgang ausgeführt wird

AUnit ist ein Test-Framework von Workday Studio, mit dem Sie Einheitentests für Ihre Assemblys durchführen können. Dieses Beispiel zeigt, wie AUnit-Tests ausgeführt werden. Es enthält 8 separate Verarbeitungsketten.

Prozedur

1. Wählen Sie in der Symbolleiste von Studio die Option Help > Willkommen bei > Samples > AUnit Tests.
2. Klicken Sie im Fenster Projektnamen auf Fertig stellen.
3. Blenden Sie in der Ansicht Project Explorer den Ordner AUnit 101 ein und klicken Sie doppelt auf den Ordner Assembly.
4. Klicken Sie in der Symbolleiste des Assembly-Editors auf Anmerkungen anzeigen.
5. Führen Sie die Beispiel-Tests AUnit 101 aus.
 - a) Blenden Sie in der Ansicht Project Explorer den Ordner src ein.
 - b) Klicken Sie mit der rechten Maustaste auf den Ordner com.workday.example.aunit .
 - c) Wählen Sie Ausführen als > Einheitentest.

Ergebnisse

Studio zeigt die Ergebnisse der Beispieltests in der Ansicht „Konsole “ an.

Nächste Maßnahme

Navigieren Sie zur Ansicht JUnit, um eine detaillierte Zusammenfassung der Ergebnisse anzuzeigen.

Beispiel: Beispiel-Assembly „FOPStep“ ausführen

Warum und wann dieser Vorgang ausgeführt wird

Mit Workday Studio können Sie PDF-Dateien in der Ausgabe Ihrer Integrationen generieren. Dieses Beispiel zeigt, wie einfache PDF-Dokumente generiert werden. Es enthält drei separate Verarbeitungsketten.

Vorbereitungen

Sicherheit: Folgende Domänen im funktionalen Bereich „Integration“:

- *Integration Build*
- *Integration Configure*
- *Integration Event*

Prozedur

1. Wählen Sie in der Symbolleiste von Studio die Option Hilfe > Willkommensbeispiele > FOP > - Schritt.
2. Klicken Sie im Fenster Projektnamen auf Fertig stellen.
3. Blenden Sie in der Ansicht Project Explorer den Ordner FOPStep ein und doppelklicken Sie auf den Ordner Assembly.
4. Klicken Sie in der Ansicht Cloud Explorer auf Verbindungsdetails.
5. Wählen Sie die Sandbox-Umgebung aus und geben Sie die folgenden Werte in das Fenster Verbindungen ein:

Option	Bezeichnung
Mandant	Der Name Ihres Mandanten.
Benutzername	Der Benutzername der Zugangsdaten für Ihren Mandanten.
Kennwort	Das Kennwort der Zugangsdaten für Ihren Mandanten.

6. Klicken Sie auf Anwenden und schließen.
7. Klicken Sie in der Ansicht Projekt Explorer mit der rechten Maustaste auf den Ordner FOPStep und wählen Sie In Workday bereitstellen aus.
8. Wählen Sie im Fenster In Workday bereitstellen die Sandbox-Umgebung aus. Klicken Sie auf Finish.
9. Blenden Sie in der Ansicht „Cloud Explorer“ die folgenden Knoten ein: Sandbox > FOPStep > FOPStep.
10. Klicken Sie mit der rechten Maustaste auf den Integrationsknoten Simple-Start und wählen Sie Integration starten aus.
11. Klicken Sie auf Starten.
12. Klicken Sie in der Ansicht Prozessmonitor mit der rechten Maustaste auf die Integration und wählen Sie Workday: Aufruf Integration ESB.

13. Geben Sie in der Ansicht Workday folgende Werte ein:

Option	Bezeichnung
Benutzername	Der Benutzername der Zugangsdaten für Ihren Mandanten.
Kennwort	Das Kennwort der Zugangsdaten für Ihren Mandanten.

14. Wählen Sie im Fenster Hintergrundprozess anzeigen das Register Ausgabedateien.

15. Doppelklicken Sie in der Spalte Datei der Tabelle Berichte und andere Ausgabedateien auf die PDF-Datei.

16. Starten Sie die Integrationsknoten Invoice-Start und Kanji-Start.

Ergebnisse

Workday zeigt die neu generierten PDF-Dateien an.

Nächste Maßnahme

Laden Sie die PDF-Dateien herunter und speichern Sie sie.

Beispiel: Beispiel-Assembly „Organisationsdaten für einen Mitarbeiter extrahieren“ ausführen

Warum und wann dieser Vorgang ausgeführt wird

Mit Workday können Sie Mitarbeiter mithilfe von Organisationsarten in Gruppen kategorisieren. Dieses Beispiel zeigt, wie eine Integration ausgeführt wird, bei der die Organisationsart und die Werte für die Mitarbeiter-ID eines Mitarbeiters aus Workday extrahiert werden.

Vorbereitungen

Sicherheit: Folgende Domänen im funktionalen Bereich „Integration“:

- *Integration Build*
- *Integration Configure*
- *Integration Event*

Prozedur

1. Wählen Sie in der Symbolleiste von Studio die Option Hilfe > Willkommensprobe > Extrahieren > von Organisationsdaten für einen Mitarbeiter.
2. Klicken Sie im Fenster Projektnamen auf Fertig stellen.
3. Blenden Sie in der Ansicht Project Explorer den Ordner OrgDataForWorkerForOrgType ein und klicken Sie doppelt auf den Ordner Assembly.
4. Klicken Sie in der Symbolleiste des Assembly-Editors auf Anmerkungen anzeigen.
5. Klicken Sie in der Ansicht Cloud Explorer auf Verbindungsdetails.
6. Wählen Sie die Sandbox-Umgebung aus und geben Sie die folgenden Werte in das Fenster Verbindungen ein:

Option	Bezeichnung
Mandant	Der Name Ihres Mandanten.

Option	Bezeichnung
Benutzername	Der Benutzername der Zugangsdaten für Ihren Mandanten.
Kennwort	Das Kennwort der Zugangsdaten für Ihren Mandanten.

7. Klicken Sie auf Anwenden und schließen.
8. Klicken Sie in der Ansicht Project Explorer mit der rechten Maustaste auf den Ordner OrgDataForWorkerForOrgType und wählen Sie Deploy to Workday aus.
9. Wählen Sie im Fenster Deploy to Workday die Sandbox-Umgebung aus. Klicken Sie auf Finish.
10. Blenden Sie in der Cloud-Explorer- Ansicht die folgenden Knoten ein: Sandbox > OrgDataForWorker > OrgDataForWorkerForOrgType.
11. Klicken Sie mit der rechten Maustaste auf den Integrationsknoten StartGetOrgForOrgType und wählen Sie Integration starten aus.
12. Navigieren Sie im Fenster Integration starten zum Abschnitt Startparameter.
13. Klicken Sie im Feld OrgType auf Literalwerte auswählen.
14. Klicken Sie im Fenster CRF auswählen auf Weiter.
15. Wählen Sie im Fenster CRF eine Organisationsart aus. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Name	Der Name der Organisationsart.
WID	Die Workday-ID der Organisationsart.

16. Klicken Sie auf Beenden.
17. Klicken Sie im Feld Personalnummer auf Literalwerte auswählen.
18. Klicken Sie im Fenster CRF auswählen auf Weiter.
19. Wählen Sie im Fenster CRF einen Mitarbeiter aus. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Name	Der Name des Mitarbeiters.
WID	Die Workday-ID des Mitarbeiters.

20. Klicken Sie auf Beenden.
21. Klicken Sie auf Starten.
22. Klicken Sie in der Ansicht Prozessmonitor mit der rechten Maustaste auf Ihre Integration.
23. Wählen Sie Details anzeigen.
24. Blenden Sie in der Ansicht Prozessdetails die Startparameterfelder OrgType und Personalnummer ein.

Ergebnisse

Studio zeigt in der Ansicht Prozessdetails die Organisationsart und die Personalnummer des Mitarbeiters an.

Beispiel: Beispiel-Assembly „PrismAnalytics“ ausführen

Warum und wann dieser Vorgang ausgeführt wird

Dieses Beispiel zeigt, wie Sie die Komponente PrismAnalytics verwenden können, um CSV-Daten an Workday Prism Analytics zu senden. CSV-Quelldateien werden im Assembly-Projekt bereitgestellt. Für dieses Beispiel kopieren Sie diese Dateien auf einen SFTP-Server, damit sie von der Beispiel-Assembly abgerufen werden können.

Vorbereitungen

Sicherheit: Folgende Domänen im funktionalen Bereich „Integration“:

- *Integration Build*
- *Integration Configure*
- *Integration Event*

Prozedur

1. Registrieren Sie einen Prism-API-Client und kopieren Sie die Werte, die die Komponente PrismAnalytics erfordert.
 - a) Öffnen Sie in Workday die Aufgabe API-Client für Integrationen registrieren.
 - b) Registrieren Sie einen API-Client bei Prism Analytics als Bereich (Funktionaler Bereich). Aktivieren Sie das Kontrollkästchen Workday-eigenen Bereich einschließen.
 - c) Kopieren Sie die Werte für Client-ID und Geheimer Client-Schlüssel.
 - d) In den möglichen Aktionen wählen Sie API-Client > Aktualisierungs-Tokens für Integrationen verwalten.
 - e) Wählen Sie ein Workday-Konto aus, das mit dem Aktualisierungs-Token verknüpft werden soll. Das Konto muss die Berechtigung zum Erstellen von Prism-Tabellen in Workday haben.
 - f) Generieren Sie ein Aktualisierungs-Token und kopieren Sie den Wert.
2. Rufen Sie den Workday-REST-API-Endpunkt ab.
 - a) Öffnen Sie den Bericht API-Clients anzeigen.
 - b) Kopieren Sie den Wert Workday REST API Endpoint . Löschen Sie die `https://` am Anfang und die `/ccx/api/v1/<tenant_name>` am Ende.
3. Wählen Sie in der Symbolleiste von Studio die Option Hilfe > – > Willkommensbeispiele > PrismAnalytics.
4. Klicken Sie im Fenster Projektnamen auf Beenden.
5. Blenden Sie in der Ansicht Projekt-Explorer den Ordner PrismAnalytics ein und doppelklicken Sie auf den Ordner Assembly.
6. Kopieren Sie die bereitgestellten CSV-Beispieldateien auf Ihren SFTP-Server.
 - a) Erweitern Sie im PrismAnalytics-Projektordner `ws > WSAR-INF > PrismAnalytics Sample CSVs`.
 - b) Kopieren Sie folgende Dateien auf Ihren SFTP-Server:
 - `States.csv`
 - `States2.csv`
 - `States3.csv`
 - `States4.csv`
 - `States5.csv`
7. Klicken Sie in der Symbolleiste des Assembly-Editors auf Anmerkungen anzeigen.
8. Klicken Sie in der Ansicht Cloud Explorer auf Verbindungsdetails.

9. Wählen Sie die Sandbox-Umgebung aus und geben Sie die folgenden Werte in das Fenster Verbindungen ein:

Option	Bezeichnung
Mandant	Der Name Ihres Mandanten.
Benutzername	Der Benutzername der Zugangsdaten für Ihren Mandanten.
Kennwort	Das Kennwort der Zugangsdaten für Ihren Mandanten.

10. Klicken Sie auf Anwenden und schließen.
11. Klicken Sie in der Ansicht Projekt-Explorer mit der rechten Maustaste auf den Ordner PrismAnalytics und wählen Sie In Workday bereitstellen.
12. Wählen Sie im Fenster In Workday bereitstellen die Sandbox-Umgebung aus und klicken Sie auf Beenden.
13. Starten Sie in der Ansicht Cloud-Explorer eine der Integrationen in diesem Beispiel.
- Erweitern Sie den Sandbox-Knoten.
 - Erweitern Sie den Sammlungsknoten PrismAnalyticsSammlung.
 - Erweitern Sie den Projektknoten PrismAnalytics.
 - Klicken Sie mit der rechten Maustaste auf einen Transportknoten von Workday-in und wählen Sie Integration starten aus. Beispiel: SingleCSV.
14. Klicken Sie auf Starten.

Ergebnisse

Workday Prism Analytics erstellt eine Tabelle mit den von Ihnen bereitgestellten CSV-Daten.

Beispiel für Schnittstelle für Drittanbieter-Entgeltabrechnung

Schritte: Beispiel der Abrechnungsschnittstelle ausführen

Warum und wann dieser Vorgang ausgeführt wird

In diesem Beispiel extrahiert eine Integration namens PayData Extract Daten, die sich auf Einmalzahlungen für eine Abrechnungsgruppe beziehen. Eine zweite Integration mit Studio durchläuft die Integrationsereignisdokumente und verwendet XSLT 3.0-Streaming, um sie zu verarbeiten.

Prozedur

- [Sicherheitsgruppe der Abrechnungsschnittstelle erstellen](#) auf Seite 404.
Erstellen Sie eine Sicherheitsgruppe mit Zugriff auf die Domänen, die für das Beispiel für die Abrechnungsschnittstelle erforderlich sind.
- Öffnen Sie die Aufgabe Create Integration System User.
Erstellen Sie einen Integrationssystembenutzer für die Integration PayData Extract. Beispiel: PayData Extract-ISU.

In den möglichen Aktionen für den von Ihnen erstellten Systembenutzer wählen Sie Sicherheitsprofil > - Sicherheitsgruppe des Integrationssystems zuweisen aus. Wählen Sie aus der Werteliste Zuzuweisende Sicherheitsgruppe des Integrationssystem die Option *Sicherheit der Abrechnungsschnittstelle* aus.

Sicherheit: Domäne *Integration Security* im funktionalen Bereich „Integration“.

3. [PayData Extract-Integrationssystem erstellen](#) auf Seite 405.
Erstellen Sie eine Integration in Workday, die Einmalzahlungsdaten für eine Abrechnungsgruppe extrahiert.
4. [PayData Extract-Integrationssystem testen](#) auf Seite 406.
5. [Studio PayData-Integration bereitstellen](#) auf Seite 407.
Stellen Sie eine Integration mit Studio bereit, die die Dokumente für das Integrationsereignis PayData Extract durchläuft und sie mithilfe von XSLT 3.0-Streaming verarbeitet.
6. Öffnen Sie die Aufgabe Create Integration System User.
Erstellen Sie einen Integrationssystembenutzer für die PayData-Integration. Beispiel: PayData-ISU.
Sicherheit: Domäne *Integration Security* im funktionalen Bereich „Integration“.
7. [Integrationssystembenutzer der PayData-Integration zuweisen](#) auf Seite 408.
8. [PayData Extract- und PayData-Integrationen verknüpfen](#) auf Seite 408.
9. [Verknüpfte Integrationen starten](#) auf Seite 409.

Sicherheitsgruppe der Abrechnungsschnittstelle erstellen

Vorbereitungen

Sicherheit: Domäne *Security Configuration* im funktionalen Bereich „System“.

Warum und wann dieser Vorgang ausgeführt wird

Bevor Sie das Beispiel für eine Abrechnungsschnittstelle in Studio ausführen können, müssen Sie eine Sicherheitsgruppe in Workday erstellen.

Prozedur

1. Melden Sie sich bei Ihrer Workday-Testumgebung an.
2. Öffnen Sie die Aufgabe Create Security Group. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Type of Tenanted Security Group	Integrationssystem - Sicherheitsgruppe (Uneingeschränkt)
Name	Sicherheit der Abrechnungsschnittstelle

3. Öffnen Sie den Bericht Sicherheitsgruppe anzeigen.
4. Wählen Sie *Sicherheit der Abrechnungsschnittstelle* aus der Werteliste Sicherheitsgruppe.
5. Wählen Sie aus dem Menü der möglichen Aktionen für die Sicherheitsgruppe Sicherheitsgruppe > - Domänenberechtigungen für Sicherheitsgruppe verwalten aus.
6. Wählen Sie die folgenden Domänen aus der Werteliste Domänen-Sicherheitsrichtlinien, die Get-Zugriff zulassen:
 - *Payroll Interface*
 - *Manage: Organization Integration*
 - *Worker Data: Public Worker Reports*
 - *Worker Data: Compensation by Organization*

Workday weist Sie darauf hin, dass Sie Ihre Änderungen der Sicherheitsrichtlinie aktivieren müssen.

7. Öffnen Sie die Aufgabe Anstehende Änderungen für Sicherheitsrichtlinie aktivieren und bestätigen Sie die Änderungen.
8. Aktivieren Sie das Kontrollkästchen Bestätigen.

Ergebnisse

Workday fügt Ihre neue Sicherheitsgruppe zu den Domänen-Sicherheitsrichtlinien jedes relevanten funktionalen Bereichs hinzu.

Nächste Maßnahme

Erstellen Sie einen Integrationssystembenutzer, um die Integration PayData Extract auszuführen.

PayData Extract-Integrationssystem erstellen

Vorbereitungen

Sicherheit: Domänen *Integration Build* und *Integration Configure* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Erstellen Sie eine Integration in Workday, die Einmalzahlungsdaten extrahiert.

Prozedur

1. Melden Sie sich bei Ihrer Workday-Testumgebung an.
2. Öffnen Sie die Aufgabe Create Integration System.
3. Bei der Bearbeitung der Aufgabe Integrationssystem erstellen sind folgende Informationen zu beachten:

Option	Bezeichnung
System Name	PayData Extract
New using Template	Abrechnungsschnittstelle

4. Aktivieren Sie auf der Seite Integrationsservices konfigurieren das Kontrollkästchen Aktiviert für die folgenden Integrationsservices:
 - Payroll Interface/PI Change Detection Launch Parameters
 - Payroll Interface/PI Pay Data Section Fields
 - Payroll Interface/PI Filename

Workday weist Sie darauf hin, dass Sie einigen Integrationsattributen Werte zuweisen und einen Sequenzgenerator für das Integrationssystem konfigurieren müssen.

5. Wählen Sie in den möglichen Aktionen für das PayData Extract-Integrationssystem Workday-Konto > Bearbeiten aus.
6. Geben Sie in der Werteliste Workday-Konto den Namen des Integrationssystembenutzers für das PayData Extract-Integrationssystem ein.
7. Klicken Sie auf der Seite Konto für Integrationssystem bearbeiten auf OK.
8. Wählen Sie in den möglichen Aktionen für das PayData Extract-Integrationssystem Integrationssystem > Integrationsattribute konfigurieren. Bei der Bearbeitung der Aufgabe sind folgende Informationen zu beachten:

Option	Bezeichnung
Version	Fügen Sie eine Zeile hinzu und wählen Sie den Wert 25 aus.
Ausgabedokument-Tags	Fügen Sie eine Zeile hinzu, wählen Sie Dokument-Tag erstellen und geben Sie als Name des Dokument-Tags <i>PICOF</i> ein.
Bevorzugte Integration für Entgeltabrechnung	Fügen Sie eine Zeile hinzu.

Option	Bezeichnung
Entgeltabrechnungsanbieter	Fügen Sie eine Zeile hinzu und wählen Sie einen beliebigen Anbieter aus.

9. Wählen Sie in den möglichen Aktionen für das PayData Extract-Integrationssystem Integrationssystem > Feldattribute der Integration konfigurieren.
10. Aktivieren Sie im Abschnitt Feldkonfiguration das Kontrollkästchen In Ausgabe einschließen für folgende Felder:
 - Code
 - Datum
 - Bezug oder Abzug
 - Betrag

11. Wählen Sie in den möglichen Aktionen für das PayData Extract-Integrationssystem Integrationssystem > Sequenzgenerator der Integration konfigurieren.

12. Auf der Seite Sequenzgeneratoren der Integration konfigurieren sind folgende Informationen zu beachten:

Option	Bezeichnung
Erhöhen um	1
Format/Syntax	Payrollinterface-paydata-[Seq].xml

Ergebnisse

Workday konfiguriert das PayData Extract-Integrationssystem.

Nächste Maßnahme

Testen Sie das PayData Extract-Integrationssystem.

PayData Extract-Integrationssystem testen

Vorbereitungen

Sicherheit: Folgende Domänen im funktionalen Bereich „Integration“:

- *Integration Build*
- *Integration Configure*
- *Integration Event*

Warum und wann dieser Vorgang ausgeführt wird

Bevor Sie das Beispiel für eine Abrechnungsschnittstelle in Studio ausführen können, müssen Sie das Integrationssystem dafür in Workday testen.

Prozedur

1. Melden Sie sich bei Ihrer Workday-Testumgebung an.
2. Öffnen Sie die Aufgabe Integrationssystem anzeigen.
3. Geben Sie in der Werteliste Integrationssystem den Wert PayData Extract ein.
4. Wählen Sie in den möglichen Aktionen für das Integrationssystem PayData Extract Integration > Starten/Planen aus.

5. Geben Sie auf der Seite Integration planen die folgenden Werte in die Spalte Wert ein:

Option	Bezeichnung
Abrechnungsgruppe	Wählen Sie eine beliebige Abrechnungsgruppe mit Einmalzahlungen aus.
Auswahloption für Abrechnungsperiode	Abrechnungsperiode für aktuelles Datum verwenden
Letzter erfolgreicher Lauf	Wählen Sie ein Datum aus, nach dem neue Einmalzahlungen erfolgt sind.

6. Starten Sie die Integration und aktualisieren Sie sie, bis sie abgeschlossen ist.

Ergebnisse

Das Integrationssystem PayData Extract wird in Workday gestartet.

Nächste Maßnahme

Stellen Sie die PayData-Integration aus Studio in Workday bereit.

Studio PayData-Integration bereitstellen

Vorbereitungen

Sicherheit: Folgende Domänen im funktionalen Bereich „Integration“:

- *Integration Build*
- *Integration Configure*

Warum und wann dieser Vorgang ausgeführt wird

Die Studio PayData-Integration durchläuft die Dokumente des Integrationsereignisses PayData Extract und verwendet XSLT 3.0-Streaming, um sie zu verarbeiten. Vor dem Start muss eine Bereitstellung in Workday erfolgen.

Prozedur

1. Wählen Sie in der Studio-Symboleiste die Optionen Hilfe > - Willkommen > - Samples > - Abrechnungsschnittstelle.
2. Klicken Sie im Fenster Projektnamen auf Beenden.
3. Erweitern Sie in der Ansicht Project Explorer den Ordner PayData und doppelklicken Sie auf den Ordner Assembly.
4. Klicken Sie in der Symbolleiste des Assembly-Editors auf Anmerkungen anzeigen.
5. Klicken Sie in der Ansicht Cloud Explorer auf Verbindungsdetails.
6. Wählen Sie die Sandbox-Umgebung aus und geben Sie die folgenden Werte in das Fenster Verbindungen ein:

Option	Bezeichnung
Mandant	Der Name Ihres Mandanten.
Benutzername	Der Benutzername der Zugangsdaten für Ihren Mandanten.
Kennwort	Das Kennwort der Zugangsdaten für Ihren Mandanten.

7. Klicken Sie auf Anwenden und schließen.

8. Klicken Sie in der Ansicht Project Explorer mit der rechten Maustaste auf den Ordner PayData und wählen Sie Deploy to Workday.
9. Wählen Sie im Fenster Deploy to Workday die Sandbox-Umgebung aus. Klicken Sie auf Beenden.

Ergebnisse

Studio stellt die PayData-Beispielintegration in Workday bereit.

Nächste Maßnahme

Erstellen Sie einen Integrationssystembenutzer, um die PayData-Integration auszuführen.

Integrationssystembenutzer der PayData-Integration zuweisen

Vorbereitungen

Sicherheit: Domäne *Integration Configure* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Weisen Sie dem PayData-Integrationssystem in Workday einen Integrationssystembenutzer zu.

Prozedur

1. Blenden Sie diese Knoten in der Ansicht Cloud Explorer ein: Sandbox > - PayData > - PayData.
2. Klicken Sie mit der rechten Maustaste auf den Integrationsknoten StartHere.
3. Wählen Sie Workday: Integrationssystem.
4. In der Ansicht Workday sind folgende Informationen zu beachten:

Option	Bezeichnung
Benutzername	Der Benutzername der Zugangsdaten für Ihren Workday-Mandanten.
Kennwort	Das Kennwort der Zugangsdaten für Ihren Mandanten.

5. Wählen Sie in den möglichen Aktionen für das PayData-Integrationssystem Workday-Konto > - Bearbeiten aus.
6. Geben Sie in der Werteliste Workday-Konto den Namen des Integrationssystembenutzers für das PayData-Integrationssystem ein.
7. Klicken Sie auf der Seite Konto für Integrationssystem bearbeiten auf OK.

Ergebnisse

Workday weist dem PayData-Integrationssystem einen Integrationssystembenutzer zu.

Nächste Maßnahme

Verknüpfen Sie die PayData Extract- und die PayData-Integration.

PayData Extract- und PayData-Integrationen verknüpfen

Vorbereitungen

Sicherheit: Domäne *Integration Configure* im funktionalen Bereich „Integration“.

Warum und wann dieser Vorgang ausgeführt wird

Verknüpfen Sie die PayData Extract- und PayData-Integrationen in Workday.

Prozedur

1. Melden Sie sich bei Ihrer Workday-Testumgebung an.
2. Öffnen Sie die Aufgabe Integrationssystem anzeigen.
3. Geben Sie in der Werteliste Integrationssystem den Wert *PayData Extract* ein.
4. Wählen Sie in den möglichen Aktionen für das PayData Extract-Integrationssystem die Aktion Geschäftsprozess > Definition erstellen, kopieren oder verknüpfen.
5. Klicken Sie auf OK, um eine neue Geschäftsprozessdefinition zu erstellen.
6. Klicken Sie auf dem Register Geschäftsprozessschritte der Seite Edit Business Process Integration Process Event for PayData Extract (TOP LEVEL) auf Zeile hinzufügen. In der neuen Zeile sind folgende Informationen zu beachten:

Option	Bezeichnung
Reihenfolge	c
Art	Integration

Workday warnt Sie, dass die Konfiguration noch nicht abgeschlossen ist.

7. Klicken Sie auf Configure Integration System.
8. Wählen Sie in der Werteliste Integration den Wert PayData.
9. Auf dem Register Integrationsattribute der Seite Integrationsschritt konfigurieren sind folgende Informationen zu beachten:

Option	Bezeichnung
Wertart	<i>Wert zur Laufzeit festlegen</i>
Wert	<i>Deliverable Documents</i>

Ergebnisse

Workday verknüpft die PayData Extract- und PayData-Integrationen miteinander, um die erforderliche Ausgabe bereitzustellen.

Nächste Maßnahme

Starten Sie die verknüpften PayData Extract- und PayData-Integrationen in Workday.

Verknüpfte Integrationen starten

Vorbereitungen

Sicherheit: Folgende Domänen im funktionalen Bereich „Integration“:

- *Integration Build*
- *Integration Configure*
- *Integration Event*

Warum und wann dieser Vorgang ausgeführt wird

Starten Sie die Integration PayData Extract, die jetzt mit der Integration PayData verknüpft ist.

Prozedur

1. Melden Sie sich bei Ihrer Workday-Testumgebung an.
2. Öffnen Sie die Aufgabe Integrationssystem anzeigen.
3. Geben Sie in der Werteliste Integrationssystem den Wert PayData Extract ein.
4. Wählen Sie in den möglichen Aktionen für das Integrationssystem PayData Extract Integration > Starten/Planen aus.

5. Geben Sie auf der Seite Integration planen die folgenden Werte in die Spalte Wert ein:

Option	Bezeichnung
Abrechnungsgruppe	Wählen Sie eine beliebige Abrechnungsgruppe mit Einmalzahlungen aus.
Auswahloption für Abrechnungsperiode	Abrechnungsperiode für aktuelles Datum verwenden
Letzter erfolgreicher Lauf	Wählen Sie ein Datum aus, nach dem neue Einmalzahlungen erfolgt sind.

6. Starten Sie die Integration und aktualisieren Sie sie, bis sie abgeschlossen ist.
7. Notieren Sie sich im Abschnitt Ereignisdokumente die CSV-Datei, die Workday erstellt hat. Beispiel: payrollinterface-paydata-1.csv.