

IBM WebSphere Partner Gateway  
Enterprise und Advanced Edition



# Unternehmensintegration

*Version 6.1.1*



IBM WebSphere Partner Gateway  
Enterprise und Advanced Edition



# Unternehmensintegration

*Version 6.1.1*

**Hinweis**

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen im Abschnitt „Bemerkungen“ auf Seite 263 gelesen werden.

Kommentare zu dieser Dokumentation können an die folgende E-Mail-Adresse gerichtet werden: doc-comments@us.ibm.com.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

**Dritte Ausgabe (März 2008)**

Diese Veröffentlichung ist eine Übersetzung des Handbuchs  
*IBM WebSphere Partner Gateway Enterprise and Advanced Editions Enterprise Integration Guide Version 6.1.1*,  
herausgegeben von International Business Machines Corporation, USA

© Copyright International Business Machines Corporation 2004, 2008  
© Copyright IBM Deutschland GmbH 2008

Informationen, die nur für bestimmte Länder Gültigkeit haben und für Deutschland, Österreich und die Schweiz nicht zutreffen, wurden in dieser Veröffentlichung im Originaltext übernommen.

Möglicherweise sind nicht alle in dieser Übersetzung aufgeführten Produkte in Deutschland angekündigt und verfügbar; vor Entscheidungen empfiehlt sich der Kontakt mit der zuständigen IBM Geschäftsstelle.

Änderung des Textes bleibt vorbehalten.

Herausgegeben von:  
SW TSC Germany  
Kst. 2877  
März 2008

---

# Inhaltsverzeichnis

<b>Zu diesem Handbuch</b> . . . . .	<b>ix</b>
Zielgruppe . . . . .	ix
Typografische Konventionen . . . . .	ix
Referenzliteratur . . . . .	x
Neuerungen in diesem Release . . . . .	x
Neuerungen in Release 6.1.1 . . . . .	xi
Neuerungen in Release 6.1 . . . . .	xi

---

## Teil 1. Einführung in die Back-End-Integration . . . . . 1

### Kapitel 1. Übersicht . . . . . 3

Übersicht über die Dokumentverarbeitung . . . . .	3
Rollen in der Hub-Community . . . . .	4
Prozess der Hubkonfiguration . . . . .	4
Übersicht über die Back-End-Integration . . . . .	7

### Kapitel 2. Back-End-Integration planen . . . . . 9

Zu verwendendes Geschäftsprotokoll . . . . .	9
Web-Services (SOAP) . . . . .	9
cXML . . . . .	10
EDI . . . . .	10
RosettaNet. . . . .	16
ebMS . . . . .	18
XML-Ereignis (XMLEvent) . . . . .	21
Zu verwendender Pakettyp . . . . .	25
None . . . . .	26
Back-End-Integrationspaket . . . . .	26
Welcher Pakettyp eignet sich für Ihre Dokumente? . . . . .	36
Beispiel für ein Back-End-Integrationspaket über HTTP. . . . .	36
Zu verwendendes Nachrichtentransportprotokoll. . . . .	37
HTTP-Transportprotokoll. . . . .	40
JMS-Protokoll. . . . .	41
Dateisystemprotokoll . . . . .	44
Zugriff auf Back-End-Anwendungen . . . . .	45
Nachrichtenbehandlung . . . . .	45
Zustellung über Warteschlangen . . . . .	46
Kommunikationsfehlerbehandlung . . . . .	46
Doppelte Nachrichten . . . . .	47
WebSphere Partner Gateway konfigurieren . . . . .	47
Dokumente an das Back-End-System senden . . . . .	48
Dokumente vom Back-End-System empfangen . . . . .	52

---

## Teil 2. Integration mit WebSphere Process Server . . . . . 57

### Kapitel 3. Einführung zur WebSphere Process Server-Integration . . . . . 59

Übersicht . . . . .	59
Kommunikation zwischen WebSphere Process Server und WebSphere Partner Gateway . . . . .	60
Integration mit WebSphere Process Server planen. . . . .	61
Von WebSphere Partner Gateway unterstützte WebSphere Process Server-Versionen . . . . .	61
Unterstützte Installationsszenarios. . . . .	61
Von WebSphere Process Server unterstützte Nachrichtentransporte . . . . .	62
Unterstützung für WebSphere Process Server-Integration . . . . .	63
Integration von WebSphere Partner Gateway mit WebSphere Process Server - Aufgabenübersicht. . . . .	64
Auf dem WebSphere Partner Gateway-System. . . . .	64

Auf dem WebSphere Process Server-System	65
Paketnachricht der Back-End-Integration verarbeiten	66
JMS-Transport	66
HTTP-Transport	67
Geschäftsobjekt der höchsten Ebene und untergeordnete Geschäftsobjekte	68
Funktionsweise der Datenbindung des Back-End-Integrationspakets	73
Klasse BCGBackEndIntegrationDataBindingUtil	79
DataBinding-Schnittstelle	79
Methoden	80
Mustercode	85
Klasse BCGBackEndIntegrationJMSDataBindingImpl	85
JMSDataBinding-Schnittstelle	85
Methoden	85
Datenbindungsklassen in Komponentenimplementierung einfügen	87

**Kapitel 4. Integration von WebSphere Process Server mit HTTP als Transportprotokoll 89**

Nachrichten an WebSphere Process Server senden	89
Nachrichten von WebSphere Process Server senden	91
Dokumente an WebSphere Process Server senden	93
WebSphere Partner Gateway konfigurieren	93
WebSphere Process Server konfigurieren	94
Dokumente von WebSphere Process Server senden	95
WebSphere Partner Gateway konfigurieren	95
WebSphere Process Server konfigurieren	95

**Kapitel 5. Integration von WebSphere Process Server mit JMS als Transportprotokoll 97**

Dokumentenversand mit dem JMS-Transportprotokoll	97
Dokumentenempfang mit dem JMS-Transportprotokoll	99
WebSphere Process Server-Umgebung in WebSphere Application Server einrichten	100
SCA-Service mit WSDL erstellen	100
JMS-Import- und -Exportbindungen anpassen	100
JMS-Datenbindung implementieren	101
Funktionsselektor anpassen	102
JMS konfigurieren, wenn WebSphere Partner Gateway auf WebSphere Application Server installiert ist	103
Busse, JMS-Warteschlangen und Verbindungsfactorys erstellen und konfigurieren	104
JMS-Empfänger erstellen	106
JMS-Ziel erstellen	107
Zielwarteschlange erstellen	108

**Kapitel 6. Integration von WebSphere Process Server mit SOAP/HTTP. 109**

SOAP-Nachrichten über das HTTP-Protokoll austauschen	109
In WebSphere Process Server enthaltene Web-Services aufrufen	109
In Partnern enthaltene Web-Services aufrufen	110
In WebSphere Process Server enthaltene Web-Services aufrufen	111
WebSphere Partner Gateway konfigurieren	111
WebSphere Process Server konfigurieren	111
In externen Partnern enthaltene Web-Services aufrufen	112
WebSphere Partner Gateway konfigurieren	112
WebSphere Process Server konfigurieren	112

**Kapitel 7. Integration von WebSphere Process Server mit Dateisystem als Transportprotokoll 115**

Dokumentenversand mit dem Dateisystemprotokoll	115
Dokumentenempfang mit dem Dateisystemprotokoll	115
WebSphere Process Server-Umgebung einrichten	115
WebSphere Adapter for Flat Files implementieren und konfigurieren	116
SCA-Komponenten erstellen	117
WebSphere Partner Gateway-Umgebung einrichten	117

<b>Teil 3. Integration mit WebSphere InterChange Server.</b>	<b>119</b>
<b>Kapitel 8. Einführung zur InterChange Server-Integration</b>	<b>121</b>
Integration mit InterChange Server planen	122
Von WebSphere Partner Gateway unterstützte InterChange Server-Versionen	122
Von InterChange Server unterstützte Nachrichtentransporte	123
Unterstützung für die InterChange Server-Integration	124
WebSphere Partner Gateway für die Verwendung mit InterChange Server konfigurieren	125
Unterstützung für ausgehende Dokumente bereitstellen	125
Unterstützung für eingehende Dokumente bereitstellen	126
InterChange Server konfigurieren	128
Geschäftsobjektdefinitionen erstellen	128
Connector erstellen	132
Collaborations erstellen	132
Projekt implementieren	133
Dokumente mit Anhängen verarbeiten	133
Konvertierung mit dem Attachment-Data-Handler	134
Umgebung für den Attachment-Data-Handler einrichten	139
Attachment-Data-Handler konfigurieren	141
Anhangsbezogene Geschäftsobjektdefinitionen erstellen	145
<b>Kapitel 9. Integration von InterChange Server über HTTP</b>	<b>153</b>
HTTP-Transportprotokoll mit ICS verwenden	153
Erforderliche Komponenten zum Dokumentenversand an ICS über den HTTP-Transport	154
Umgebung für den HTTP-Transport mit ICS einrichten	157
Geschäftsobjektdefinitionen für ICS über HTTP erstellen	160
ICS-Artefakte für HTTP erstellen	167
SOAP-Dokumente über HTTP/S senden	169
Erforderliche Komponenten für den Dokumentenversand und -empfang	169
Web-Service-Aufruf durch externe Partner	170
Web-Service-Aufruf durch den internen Partner	171
<b>Kapitel 10. Integration von InterChange Server über JMS</b>	<b>173</b>
Erforderliche Komponenten für den Austausch von Dokumenten über den JMS-Transport	173
Dokumentenversand über das JMS-Transportprotokoll	174
Dokumentenempfang über das JMS-Transportprotokoll	176
Umgebung für den JMS-Transport einrichten	179
JMS-Warteschlangen konfigurieren	179
Adapter für JMS konfigurieren	180
Geschäftsobjektdefinitionen für JMS erstellen	182
Struktur des Geschäftsobjekts für Nutzdaten für JMS erstellen	182
JMS-Headerinformationen erstellen	183
ICS-Artefakte für JMS erstellen	186
JMS-Connectorobjekt erstellen	186
Collaborations zur Kommunikation mit dem Adapter für JMS binden	187
<b>Teil 4. Integration mit anderen Back-End-Systemen.</b>	<b>189</b>
<b>Kapitel 11. Integration mit WebSphere Message Broker</b>	<b>191</b>
Integration mit WebSphere Message Broker planen	192
Von WebSphere Partner Gateway unterstützte WebSphere Message Broker-Versionen	192
Von WebSphere Message Broker unterstützte Nachrichtentransporte	192
Unterstützung für WebSphere Message Broker-Integration	193
WebSphere Partner Gateway für die Verwendung mit WebSphere Message Broker konfigurieren	193
Unterstützung für ausgehende Dokumente bereitstellen	193
Unterstützung für eingehende Dokumente bereitstellen	195
WebSphere Message Broker konfigurieren	196
Nachrichtenfluss erstellen	197
Projekt implementieren	197

HTTP-Transportprotokoll mit WebSphere Message Broker verwenden . . . . .	197
Erforderliche Komponenten für den Austausch von Dokumenten über den HTTP-Transport . . . . .	197
Nachrichtenfluss für den HTTP-Transport erstellen . . . . .	199
SOAP-Dokumente senden . . . . .	201
JMS-Transportprotokoll mit WebSphere Message Broker verwenden . . . . .	203
Erforderliche Komponenten für den Austausch von Dokumenten über den JMS-Transport . . . . .	203
Umgebung für den JMS-Transport einrichten . . . . .	207
Nachrichtenfluss für den JMS-Transport erstellen . . . . .	208
<b>Kapitel 12. Integration mit WebSphere Data Interchange . . . . .</b>	<b>211</b>
Zielgruppe dieses Kapitels . . . . .	211
Für dieses Kapitel zu verwendende Ressourcen . . . . .	211
Einführung . . . . .	212
Dokumentenversand an WebSphere Data Interchange definieren . . . . .	212
Dokumentenempfang von WebSphere Data Interchange definieren . . . . .	213
Beispielszenario für dieses Kapitel . . . . .	214
Integration mit WebSphere Data Interchange planen . . . . .	215
Von WebSphere Partner Gateway unterstützte WebSphere Data Interchange-Versionen . . . . .	215
Umgebung für den Nachrichtenaustausch konfigurieren . . . . .	215
WebSphere MQ-Kommunikation konfigurieren . . . . .	215
WebSphere Data Interchange konfigurieren . . . . .	217
JMS-Umgebung einrichten . . . . .	222
WebSphere Partner Gateway Enterprise Edition konfigurieren . . . . .	223
WebSphere Partner Gateway - Express konfigurieren . . . . .	231
Mein Profil konfigurieren . . . . .	231
Partner für Partner 1 erstellen . . . . .	232
Partner für Partner 1 konfigurieren . . . . .	232
Zusammenfassung . . . . .	233

---

## **Teil 5. Integration mit anderen Produkten . . . . . 235**

### **Kapitel 13. Integration mit WebSphere Transformation Extender . . . . . 237**

Einführung . . . . .	237
Was ist WebSphere Transformation Extender? . . . . .	237
Zusammenarbeit zwischen WebSphere Transformation Extender und WebSphere Partner Gateway . . . . .	237
Integration mit WebSphere Transformation Extender planen . . . . .	238
Integration mit WebSphere Message Broker . . . . .	238
Integration mit einem WebSphere Partner Gateway-Benutzerexit . . . . .	238
Integration von WebSphere Message Broker mit WebSphere Transformation Extender . . . . .	240
Integration mit WebSphere Transformation Extender-Benutzerexit . . . . .	240
WebSphere Partner Gateway-Benutzerexit für WebSphere Transformation Extender erstellen . . . . .	240
WebSphere Partner Gateway für den Benutzerexit konfigurieren . . . . .	240
WebSphere Partner Gateway-Laufzeitkonfiguration für den Benutzerexit . . . . .	241
WebSphere Partner Gateway-Beispielbenutzerexit . . . . .	241
WebSphere Transformation Extender - Konfiguration des RMI-Servers . . . . .	244

### **Kapitel 14. Integration mit WebSphere Transformation Extender Trading Manager . . . 247**

Einführung . . . . .	247
Was ist WebSphere Transformation Extender Trading Manager? . . . . .	247
Zusammenarbeit zwischen WebSphere Transformation Extender Trading Manager und WebSphere Partner Gateway . . . . .	247
Relevante Dokumentation . . . . .	247
Integration mit WebSphere Transformation Extender Trading Manager planen . . . . .	248
Gemeinsame Konfiguration bei WebSphere Partner Gateway und WebSphere Transformation Extender Trading Manager . . . . .	248
WebSphere Partner Gateway konfigurieren . . . . .	249
Hinweise zum internen Partner . . . . .	249
Hinweise zum externen Partner . . . . .	250
Hinweise zur Dokumentdefinition . . . . .	250
Hinweise zu Interaktionen . . . . .	250

Hinweise zu Verbindungen . . . . .	250
Hinweise zum Empfänger . . . . .	250
WebSphere Transformation Extender Trading Manager konfigurieren. . . . .	251

---

**Teil 6. Andere geschäftsprotokollspezifische Informationen . . . . . 253**

**Kapitel 15. EDI-Dokumente weiterleiten . . . . . 255**

Übersicht über das EDI-Routing . . . . .	255
------------------------------------------	-----

**Kapitel 16. Besondere Hinweise zum AS-Paket . . . . . 257**

Routing eingehender Dokumente. . . . .	257
Routing ausgehender Dokumente . . . . .	257
Beide IDs im Partnerprofil festlegen . . . . .	257
AS3-MDN-Verarbeitung . . . . .	258
Weitere AS-Referenzen . . . . .	258

**Kapitel 17. Besondere Hinweise zum RosettaNet-Paket . . . . . 259**

Weitere RosettaNet-Referenzen . . . . .	259
-----------------------------------------	-----

**Kapitel 18. Besondere Hinweise zum ebMS-Paket. . . . . 261**

Weitere ebMS-Referenzen . . . . .	261
-----------------------------------	-----

**Bemerkungen . . . . . 263**

Informationen zu Programmierschnittstellen . . . . .	265
Marken und Servicemarken . . . . .	265

**Index . . . . . 267**



---

## Zu diesem Handbuch

In diesem Handbuch wird die Schnittstelle für die Back-End-Integration beschrieben. Diese Schnittstelle ist der Mechanismus, mit dem Back-End-Systeme und IBM<sup>(R)</sup> WebSphere<sup>(R)</sup> Partner Gateway miteinander kommunizieren. Darüber hinaus wird in diesem Handbuch beschrieben, wie WebSphere Process Server, WebSphere InterChange Server, WebSphere Message Broker und WebSphere Data Interchange mit Hilfe der Schnittstelle für die Back-End-Integration mit WebSphere Partner Gateway integriert werden.

Die Informationen in diesem Handbuch beziehen sich nur auf WebSphere Partner Gateway Enterprise Edition und Advanced Edition.

---

## Zielgruppe

Dieses Buch richtet sich an Personen, die für die Integration von WebSphere Partner Gateway mit Back-End-Systemen verantwortlich sind.

---

## Typografische Konventionen

In diesem Dokument werden die folgenden Konventionen verwendet.

*Tabelle 1. Typografische Konventionen*

<b>Konvention</b>	<b>Beschreibung</b>
Monospaceschrift	In Monospaceschrift dargestellter Text kennzeichnet Elemente, die vom Benutzer eingegeben werden müssen, Werte für Argumente oder Befehlsoptionen, Beispiele und Codebeispiele sowie Informationen, die vom System am Bildschirm ausgegeben werden (Nachrichtentexte oder Systemanfragen).
<b>Fettdruck</b>	In Fettdruck dargestellter Text kennzeichnet Steuerelemente der grafischen Benutzerschnittstelle (z. B. die Namen von Schaltflächen, Menüs oder Menüoptionen) und Spaltenüberschriften in Tabellen und im Fließtext.
<i>Kursivdruck</i>	In Kursivdruck dargestellter Text kennzeichnet Hervorhebungen, Buchtitel, neue Termini und Termini, die im Text definiert werden. Darüber hinaus werden in Kursivdruck Variablennamen und alphabetische Zeichen dargestellt, die als Literalwerte benutzt werden.
<i>Monospaceschrift in Kursivdruck</i>	In kursiv gedruckter Monospaceschrift dargestellter Text kennzeichnet Variablennamen innerhalb von Textsegmenten, die in Monospaceschrift gedruckt sind.
<i>produktverz</i>	Die Angabe <i>produktverz</i> kennzeichnet das Verzeichnis, in dem das Produkt installiert wurde. Alle Pfadnamen in IBM WebSphere Partner Gateway sind relativ und beziehen sich auf das Verzeichnis, in dem IBM WebSphere Partner Gateway auf dem System installiert ist.
<i>%text%</i> und <i>\$text</i>	In Prozentzeichen (%) dargestellter Text gibt den Wert für die Windows <sup>(R)</sup> -Systemvariable oder Benutzervariable <i>text</i> an. Die entsprechende Notation in UNIX <sup>(R)</sup> -Umgebungen lautet <i>\$text</i> und gibt den Wert der UNIX-Umgebungsvariablen <i>text</i> an.

Tabelle 1. Typografische Konventionen (Forts.)

Konvention	Beschreibung
Unterstrichener farbiger Text	Unterstrichener farbiger Text kennzeichnet Querverweise. Wenn Sie auf diesen Text klicken, wird das Objekt aufgerufen, auf das sich der Verweis bezieht.
<i>Text in einem blauen Rahmen</i>	(Nur in PDF-Dateien) Ein blauer Rahmen um ein Textelement kennzeichnet einen Querverweis. Wenn Sie auf den umrandeten Text klicken, wird das Objekt aufgerufen, auf das sich der Verweis bezieht. Diese Konvention in PDF-Dateien entspricht der in der vorliegenden Tabelle bereits erläuterten Textkonvention für den unterstrichenen farbigen Text.
“ ” (Anführungszeichen)	(Nur in PDF-Dateien) Querverweise auf andere Abschnitte des Dokuments stehen in Anführungszeichen.
{ }	In einer Zeile mit Syntaxelementen wird in geschweiften Klammern eine Gruppe von Optionen dargestellt, von der eine Option ausgewählt werden muss.
[ ]	In einer Zeile mit Syntaxelementen wird in eckigen Klammern ein optionaler Parameter dargestellt.
< >	Spitze Klammern werden verwendet, um variable Elemente eines Namens voneinander zu trennen. Beispiel: <servername><connectorname>tmp.log.
/, \	Backslashes (\) werden in Windows-Installationen zur Trennung der einzelnen Elemente eines Verzeichnispfads verwendet. Bei UNIX-Installationen müssen die Backslashes (umgekehrte Schrägstriche) durch normale Schrägstriche (/) ersetzt werden.

---

## Referenzliteratur

Die gesamte, zum vorliegenden Produkt bereitgestellte Dokumentation enthält umfassende Informationen zur Installation, Konfiguration, Verwaltung und Verwendung von WebSphere Partner Gateway Enterprise Edition und Advanced Edition.

Sie können diese Dokumentation von der folgenden Website herunterladen oder direkt dort online lesen: <http://www.ibm.com/software/integration/wspartnergateway/library/>

**Anmerkung:** Wichtige Informationen zum vorliegenden Produkt, die erst nach der Veröffentlichung des vorliegenden Dokuments verfügbar wurden, werden bei Bedarf in technischen Hinweisen (TechNotes) der technischen Unterstützungsfunktion und in Aktualisierungen (Flashes) bereitgestellt. Diese können von der Unterstützungswebsite für WebSphere Business Integration (<http://www.ibm.com/software/integration/wspartnergateway/support>) heruntergeladen werden. Wählen Sie dort den Bereich mit den für Sie relevanten Informationen aus, und durchsuchen Sie dort die Bereiche “Technotes” und “Flashes”.

---

## Neuerungen in diesem Release

In diesem Abschnitt werden die neuen Funktionen von IBM WebSphere Partner Gateway beschrieben.

## Neuerungen in Release 6.1.1

WebSphere Partner Gateway 6.1.1 unterstützt die folgenden neuen Funktionen:

- In früheren Releases war die Unterstützung der Basisauthentifizierung nur für Nachrichten des Typs "Web-Services" verfügbar. Diese Funktion wurde nun auf alle Protokolle erweitert. Für die Basisauthentifizierung wird empfohlen, eine gesicherte HTTP-Verbindung (d. h. HTTPS anstelle von HTTP) zu verwenden.
- Neben Unterzeichnung und Verschlüsselung wird nun auch die Unterstützung für die Komprimierung und Dekomprimierung von RNIF-Nachrichten bereitgestellt.
- Unterstützung für die Validierung des SOAP-Hauptteils (SOAP Body) und des SOAP-Umschlags (SOAP Envelope). Darüber hinaus können Sie den Umschlag eines SOAP-Umschlags entfernen.
- Das maximale Zeitlimit für synchrone Verbindungen und die maximale Anzahl synchroner Verbindungen können für jeden HTTP-Empfänger lokal gesteuert werden.
- Der FTP-Server ist in WebSphere Partner Gateway integriert und unterstützt das AS3-Protokoll, das FTP-Scripting-Ziel, den FTP-Scripting-Empfänger sowie den FTP/FTPS-Empfänger und das FTP/FTPS-Ziel.
- Ein Fehlerdokument kann an den einleitenden Partner, den empfangenden Partner oder beide Partner gesendet werden. Der Fehlerdokumentenfluss kann in WebSphere Partner Gateway Console konfiguriert werden und kann im WebSphere Partner Gateway-Format oder im Web-Services-Format vorliegen.
- Die Leistung der Archivierungsfunktion wurde verbessert.
- Mehrere interne Partner werden unterstützt.
- Sie können mehrere eingehende (inbound) und ausgehende (outbound) Dokumente gleichzeitig senden.
- Der FIPS-Modus wird unterstützt. Das Produkt kann für die Ausführung im FIPS-Modus oder im Standardmodus konfiguriert werden.
- Die Funktion für die Befehle "Delete" (Löschen) und "Whereused" (Verwendet von) wird für das Ziel, die Validierungszuordnungen, Dokumentdefinitionen, Interaktionen und Benutzer bereitgestellt.
- Für AS2- und AS3-Dokumente wird die Unterstützung für die Komprimierung großer Dateien bereitgestellt.
- Unterstützung für die Verschlüsselung und für Signaturen wird nun bereitgestellt.
- Die Konfigurationstypabhängigkeiten für die Migration umfassen auch Ereigniscodes und Alertbenachrichtigungen. Darüber hinaus wurde die Funktionalität der Partnermigration erweitert. Sie unterstützt nun den Import und Export von Definitionen alertfähiger Ereignisse.
- Das Hochladen mehrerer Zertifikate wird unterstützt. Die Konsole enthält nun einen neuen Assistenten für das Hochladen und Konfigurieren von Zertifikaten.
- Das Produkt unterstützt nun AIX 6.1, RHEL 5 (32- und 64-Bit), SLES 10 (64-Bit) und Windows Server 2003 (64-Bit).

## Neuerungen in Release 6.1

WebSphere Partner Gateway V6.1 unterstützt die folgenden neuen Funktionen:

- Neue Geschäftsprotokolle: Unterstützung für AS3, SOAP with attachments, CIDX und ebXML Message Service (ebMS) 2.0.

- Verbesserte Unterstützung für angepasste XML-Dokumente, d. h. bessere Organisation, vollständige Unterstützung für XPath-Ausdrücke, Suchfelder, benutzerdefinierte Attribute und synchrone Unterstützung.
- Neue IPV6-Unterstützung (Internet Protocol Version 6) sowie erweitertes FTP-Scripting zur Unterstützung von AS3.
- Reorganisation von Dokumentdefinitionsattributen.
- Neue Dokumentdefinitionsattribute zur Verwendung mit Benutzerexits.
- Unbestreitbarkeit konfigurierbar nach Dokumenttyp und auf der Ebene der Handelspartner.
- Zusätzliche benutzerdefinierte Suchfelder in der Dokumentanzeige.
- Verbesserte Unterstützung für die AS-Anzeige auf der Basis des MDN-Rückgabestatus.
- EDI-Konfigurationsassistent und EIF-Importassistent (diese wurden zuvor im Support-Pack GA02 bereitgestellt).
- Neuer Alertbenachrichtigungsmodus zum Senden von Benachrichtigungen an alle beteiligten Parteien (Quellen- und Zielpartner) bzw. an alle subskribierten Kontakte. Dadurch reduziert sich der Aufwand für die Alertkonfiguration.
- Berechtigungen zum erneuten Senden und für das Gateway stehen nicht nur dem Administrator "Hubadmin", sondern auch anderen Benutzern zur Verfügung.
- Neue Benutzergruppe, damit mehrere Benutzer die Funktion des Hubadministrators übernehmen können.
- LDAP-Unterstützung für die Anmeldeauthentifizierung.
- Verwendung der Protokollierungs- und Tracefunktion von WebSphere Application Server für WebSphere Partner Gateway-Komponenten.
- Konfigurationsdaten der Merkmaldatei befinden sich jetzt an zentraler Position und werden über die WebSphere Partner Gateway-Konsole verwaltet.
- WebSphere MQ ist kein vorausgesetztes Produkt mehr; die interne Kommunikation erfolgt nun über die Unterstützung für WebSphere Platform Messaging.
- Auf der Basis des Partners und des Dokumenttyps auswählbares Archiv.
- Migration der WebSphere Partner Gateway-Konfiguration durch den Export und Import von Definitionen aus einer WebSphere Partner Gateway-Instanz in eine andere Instanz.
- Option für die vereinfachte Installation auf einer einzelnen Maschine (einfacher Modus).
- Verwendung von WebSphere Application Server Network Deployment zur Implementierung auf mehreren Maschinen, um Clustering und zentrales Infrastrukturmanagement zu ermöglichen.
- Unterstützung für die Verwendung von WebSphere Process Server Version 6.1 als Back-End-Integrationssystem.

**Anmerkungen:**

1. Die XML-basierte Administrator-API wird in Version 6.1 nicht weiter unterstützt.
2. WebSphere Partner Gateway Version 6.1 bietet keine Unterstützung für den RC5-Algorithmus.

---

## **Teil 1. Einführung in die Back-End-Integration**

Die folgenden Kapitel enthalten eine Übersicht über die Integration von WebSphere Partner Gateway mit einem Back-End-System. Darüber hinaus enthalten sie Richtlinien für die Planung der Integration mit einem Back-End-System.



---

# Kapitel 1. Übersicht

Dieses Kapitel enthält die folgenden allgemeinen Informationen zur Back-End-Integration:

- „Übersicht über die Dokumentverarbeitung“
- „Übersicht über die Back-End-Integration“ auf Seite 7

---

## Übersicht über die Dokumentverarbeitung

Mit Hilfe von WebSphere Partner Gateway tauschen Sie Geschäftsdokumente mit Ihren externen Partnern aus. Diese Dokumente werden ausgetauscht, um Informationen weiterzugeben. Hierbei werden in der Regel Daten verarbeitet und ein Ergebnis zurückgegeben. Wenn Sie Daten von einem externen Partner empfangen, findet die Verarbeitung dieser Daten im Allgemeinen im Back-End-System Ihres Unternehmens statt. WebSphere Partner Gateway ist die Stelle in der Hub-Community, über die Nachrichten an das Unternehmen und aus dem Unternehmen weitergeleitet werden.

Der Zugriff auf das Unternehmen erfolgt über ein Back-End-System, zu dem WebSphere Partner Gateway eine Verbindung herstellt.

In Abb. 1 wird der Dokumentenfluss durch WebSphere Partner Gateway Enterprise Edition und Advanced Edition dargestellt. Ein Partner sendet ein Dokument an WebSphere Partner Gateway (den Hub). WebSphere Partner Gateway empfängt das Dokument und führt zuvor definierte Maßnahmen (wie eine Validierung oder Transformation des Dokuments) durch. Anschließend sendet WebSphere Partner Gateway das Dokument an eine Back-End-Anwendung, die das Dokument verarbeitet.

**Anmerkung:** Wie in der Abbildung zu erkennen ist, findet auch ein Informationsfluss in entgegengesetzter Richtung statt. Die Back-End-Anwendung kann ein Dokument generieren und es an den Hub senden. Dieser verarbeitet es und sendet es dann an den Partner.

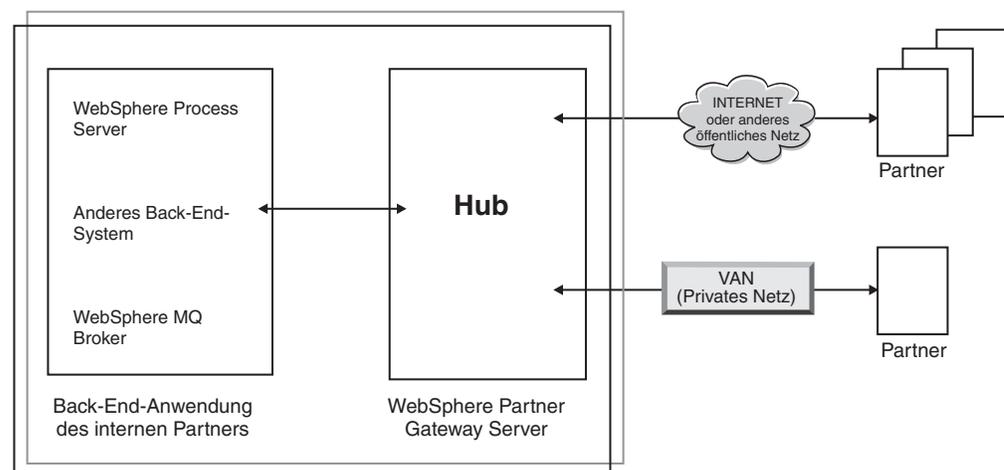


Abbildung 1. Vollständiger Verlauf des Dokumentenflusses

Im vorliegenden Handbuch wird schwerpunktmäßig die Integration zwischen dem Hub und der Back-End-Anwendung behandelt.

**Anmerkung:** Die im vorliegenden Dokument enthaltenen Informationen gelten sowohl für WebSphere Partner Gateway Enterprise Edition als auch für WebSphere Partner Gateway Advanced Edition. Bei WebSphere Partner Gateway - Express handelt es sich um ein schlankes, benutzerfreundliches B2B-Konnektivitätstool, das sich von WebSphere Partner Gateway Enterprise Edition und Advanced Edition unterscheidet. Es stellt eine Community-Integrationslösung bereit (während WebSphere Partner Gateway Enterprise Edition und Advanced Edition eine Gateway-Hub-Lösung für einen internen Partner bereitstellen). Informationen zu WebSphere Partner Gateway - Express finden Sie im *WebSphere Partner Gateway Benutzerhandbuch*.

## Rollen in der Hub-Community

WebSphere Partner Gateway Enterprise Edition und Advanced Edition verwenden drei verschiedene Partnertypen - den Hubadministrator, den internen Partner und die externen Partner. Der Hubadministrator wird bei der Installation von WebSphere Partner Gateway automatisch erstellt. Der Hubadministrator hat die Aufgabe, den Hub zu konfigurieren und die Partner zu erstellen, die mit dem Hub interagieren sollen.

Der interne Partner ist normalerweise der Eigner des Hubs, wird aber als einer der Partner des Hubs betrachtet. Der Hubadministrator erstellt ein Profil für den internen Partner. Dazu gibt er die Informationen an, die erforderlich sind, damit der interne Partner Dokumente an die Partner senden und von den Partnern empfangen kann. Wenn der Hub Dokumente an das Back-End-System sendet, verwendet er hierzu die Informationen (z. B. die URL oder die JMS-Warteschlange), die für den internen Partner konfiguriert wurden. Der Hubadministrator erstellt darüber hinaus die Profile für die Partner, wobei mehrere Partner definiert werden können.

## Prozess der Hubkonfiguration

Der Hubadministrator ist der Benutzer, der für die Verwaltung des Hubs verantwortlich ist. Er konfiguriert den Hub, so dass Geschäftsdokumente vom internen Partner und von den externen Partnern gesendet und empfangen werden können. Damit Geschäftsdokumente vom internen Partner empfangen werden können, erstellt der Hubadministrator die Empfänger für die Transportmethoden, die der interne Partner zum Senden von Dokumenten verwendet. Verwendet der interne Partner beispielsweise die Transportmethoden für Dateiverzeichnisse und JMS, definiert der Hubadministrator einen Dateiverzeichnisempfänger und einen JMS-Empfänger für den internen Partner. Verwenden die externen Partner den HTTP- und den FTP-Transport, definiert der Hubadministrator für diese einen HTTP- und einen FTP-Empfänger.

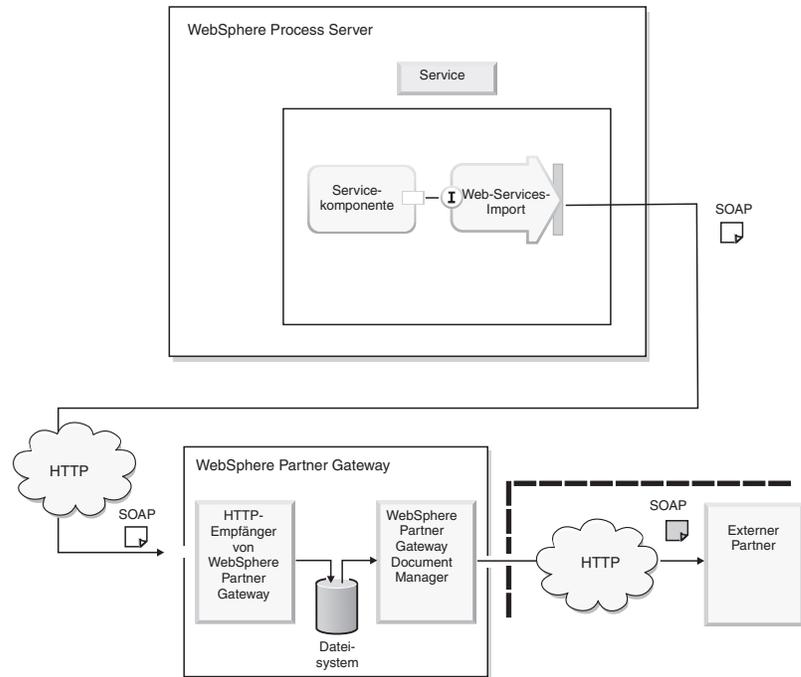


Abbildung 2. Empfänger für den internen Partner und die externen Partner

Für alle Transporte, die vom internen Partner und den externen Partnern verwendet werden, um die vom Hub gesendeten Geschäftsdokumente zu empfangen, werden Ziele erstellt.

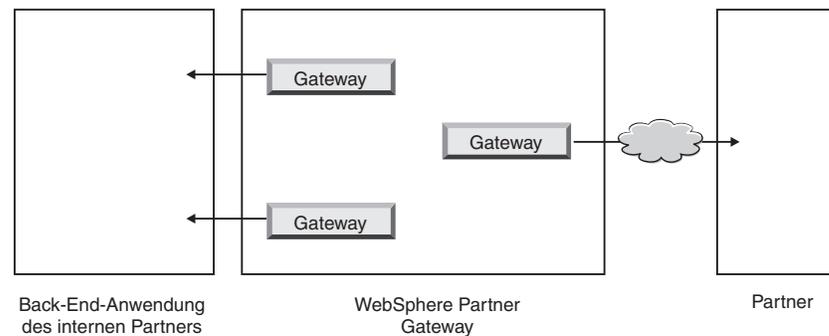


Abbildung 3. Ziele für den internen Partner und die externen Partner

Bei der Hubkonfiguration legt der Hubadministrator Dokumentenflussdefinitionen fest, in denen bestimmte Eigenschaften eines Dokumentenflusses definiert werden. Zum Beispiel:

- Pakettyp - Informationen zum Routing des Dokuments.
- Protokoll - Informationen zu dem für das Dokument verwendeten Geschäftsprotokoll.
- Dokumentenfluss - Informationen zum Dokument selbst.

Bei der Installation von WebSphere Partner Gateway wird eine Reihe von Dokumentdefinitionen bereitgestellt. Sie können auch weitere Dokumentdefinitionen hinzufügen, indem Sie selbst eigene Definitionen erstellen oder Definitionen hochladen. Der Installationsdatenträger enthält zum Beispiel ZIP-Dateien mit Dokumentdefinitionen für eine Reihe von RosettaNet-PIPs (PIP - Partner Interface Process).

Sie können diese Dateien hochladen und damit zur Benutzung verfügbar machen. Beim Austausch von EDI-Dateien können Sie Dokumentdefinitionen und die zugehörigen Zuordnungen vom Data Interchange Services-Client importieren.

Beispiel: Ein externer Partner sendet eine RNIF 2.0-Nachricht mit einem RosettaNet PIP 3A4-Bestelldokument an den HTTP-Empfänger von WebSphere Partner Gateway. Die Nachricht ist für den internen Partner bestimmt. Der interne Partner verfügt über ein Back-End-System zur Verarbeitung von Bestellungen und erwartet den Empfang der Bestellung. Diese Bestellung stellt die Nutzdaten der vom externen Partner gesendeten RNIF-Nachricht dar. Bevor die Verbindungen für den externen Partner in WebSphere Partner Gateway eingerichtet werden, wird Folgendes vereinbart:

- Der externe Partner sendet eine RNIF-Nachricht, die das RosettaNet PIP 3A4-Bestelldokument enthält, über HTTP.
- WebSphere Partner Gateway extrahiert die Nutzdaten oder den RosettaNet Service Content (RNSC) der eingehenden Nachricht.
- Das Dokument wird über JMS an das Back-End-System weitergeleitet. Hierbei wird der Pakettyp 'Backend Integration' verwendet.
- Die Back-End-Anwendung verarbeitet das empfangene Dokument.

Wird der Pakettyp 'Backend Integration' verwendet, werden von WebSphere Partner Gateway definierte Transportheader zum Dokument hinzugefügt. Diese enthalten Informationen, die für den Dokumentaustausch benötigt werden.

Im hier angeführten Beispiel würde der Hubadministrator das erforderliche PIP-Paket hochladen. Auf diese Weise würden die folgenden Dokumentdefinitionen für den RosettaNet PIP 3A4-Austausch definiert werden:

- Ein Dokumentenfluss, der aus dem RNIF-Paket, dem RosettaNet-Protokoll und dem 3A4-PIP besteht.
- Ein Dokumentenfluss, der aus dem Back-End-Integrationspaket, dem RNSC-Protokoll und dem 3A4-PIP besteht.

Nachdem der Hubadministrator die Dokumentdefinitionen festgelegt hat, erstellt er Interaktionen für die Dokumentenflussdefinitionen. Der Hubadministrator kann zum Beispiel angeben, dass die Dokumentenflussdefinition RNIF/RosettaNet/3A4 aus einer Quelle in den Hub gelangen kann.

Der Hubadministrator bzw. die externen Partner wählen die für den Dokumentaustausch erforderliche B2B-Funktionalität aus. Im vorliegenden Beispiel muss für den internen Partner die folgende B2B-Funktionalität aktiviert werden:

- Paket: Back-End-Integration.
- Protokoll: RNSC.
- Dokumentenfluss: 3A4.

Für den externen Partner muss die folgende B2B-Funktionalität aktiviert werden:

- Paket: RNIF.
- Protokoll: RosettaNet.
- Dokumentenfluss: 3A4.

Anschließend erstellt der Hubadministrator Verbindungen zwischen den externen Partnern.

In der folgenden Abbildung hat der Hubadministrator Profile für den internen Partner und den externen Partner erstellt, Empfänger zum Empfangen von Dokumenten und Ziele zum Senden von Dokumente definiert, die oben aufgeführten Dokumentenflussdefinitionen erstellt, die B2B-Funktionalität des externen Partners und des internen Partners festgelegt und eine Verbindung zwischen den beiden erstellt.

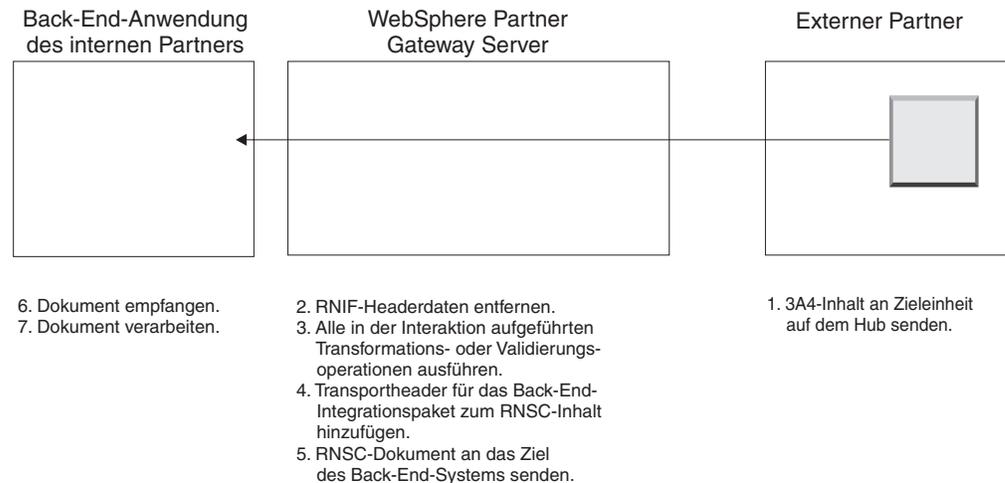


Abbildung 4. Fluss eines Dokuments zum Back-End-System

Weitere Informationen zur Konfiguration des Hubs finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

## Übersicht über die Back-End-Integration

Alle Editionen von WebSphere Partner Gateway bieten die Möglichkeit, Verbindungen zu Back-End-Systemen herzustellen. Diese Editionen unterscheiden sich in Bezug auf die jeweils unterstützten Transportprotokolle wie folgt:

- WebSphere Partner Gateway - Express unterstützt die dateibasierte Integration.
- WebSphere Partner Gateway Enterprise Edition und Advanced Edition unterstützen ebenfalls die dateibasierte Integration. Darüber hinaus ermöglichen diese Produkte die Integration über die Protokolle HTTP, HTTPS und JMS.

Die Dokumente, die zwischen dem externen Partner und WebSphere Partner Gateway ausgetauscht werden, können eine Reihe unterschiedlicher Formate aufweisen. Sie können im SOAP-, cXML-, XML- und EDI-Format, im ROD/Flat File-Format (ROD - Record-oriented Data) oder im Binärformat, aber auch in einem beliebigen angepassten Format vorliegen, das zwischen den externen Partnern vereinbart wurde. Das Handbuch *WebSphere Partner Gateway Verwaltung* enthält eine vollständige Liste der unterstützten Dokumenttypen sowie der Transportprotokolle (z. B. HTTP), die zum Senden der Dokumente verwendet werden können.

Dokumente, die zwischen WebSphere Partner Gateway und dem Back-End-System des internen Partners ausgetauscht werden können, sowie die den Dokumenten zugeordneten Transporttypen sind in Tabelle 16 auf Seite 38, Tabelle 17 auf Seite 38 und Tabelle 18 auf Seite 39 aufgeführt.

Abb. 5 veranschaulicht, wie WebSphere Partner Gateway die Back-End-Integrationsschnittstelle zur Kommunikation mit dem Back-End-System des internen Partners verwendet. Beachten Sie, dass der Pfeil in beide Richtungen zeigt. Das heißt, das Dokument kann auch aus dem Back-End-System des internen Partners stammen.

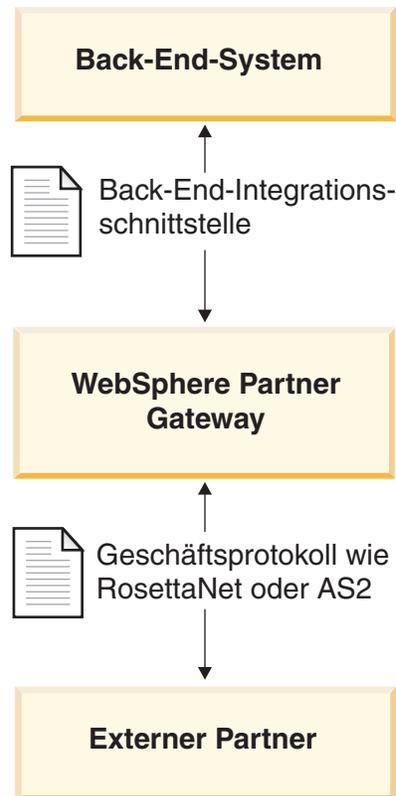


Abbildung 5. Die Rolle des Geschäftsprotokolls und der Paketerstellung im Dokumentenfluss

---

## Kapitel 2. Back-End-Integration planen

In diesem Kapitel wird die Planung der Integration von WebSphere Partner Gateway mit einem Back-End-System beschrieben. Es enthält Informationen zu den verschiedenen Entscheidungen, die während der Planung der Back-End-Integration von Ihnen getroffen werden müssen. Hierbei werden die folgenden Faktoren berücksichtigt:

- „Zu verwendendes Geschäftsprotokoll“
- „Zu verwendender Pakettyp“ auf Seite 25
- „Zu verwendendes Nachrichtentransportprotokoll“ auf Seite 37
- „Zugriff auf Back-End-Anwendungen“ auf Seite 45

Darüber hinaus finden Sie hier Informationen zu folgenden Themen:

- „Nachrichtenbehandlung“ auf Seite 45
- „WebSphere Partner Gateway konfigurieren“ auf Seite 47

---

### Zu verwendendes Geschäftsprotokoll

Das von Ihrer Nachricht verwendete Geschäftsprotokoll bestimmt das Dokumentformat. Das Geschäftsprotokoll ist für viele Entscheidungen relevant, die Sie beim Planen der Integration mit einem Back-End-System treffen müssen. Die Auswahl des Geschäftsprotokolls bestimmt die Paketerstellungsmethode, die Sie verwenden müssen und die wiederum die verwendbaren Nachrichtentransportprotokolle beeinflusst.

Eine umfassende Beschreibung der verfügbaren Geschäftsprotokolle finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*. Der vorliegende Abschnitt enthält Informationen zur Integration, die sich speziell auf die folgenden Geschäftsprotokolle beziehen:

- „Web-Services (SOAP)“
- „cXML“ auf Seite 10
- „EDI“ auf Seite 10

**Anmerkung:** Der Abschnitt zu EDI enthält auch Informationen zur Verarbeitung von XML- und ROD/flat-Dokumenten (ROD - Record Oriented Data).

- „RosettaNet“ auf Seite 16

### Web-Services (SOAP)

WebSphere Partner Gateway kann Mitgliedern der Hub-Community die folgenden Web-Services zur Verfügung stellen:

- Vom internen Partner bereitgestellte Web-Services können den externen Partnern zur Verfügung gestellt werden.

Sie müssen Ihrem externen Partner die öffentliche WSDL (WSDL - Web Service Definition Language) zur Verfügung stellen, die von WebSphere Partner Gateway generiert wird. Es ist wichtig zu beachten, dass die URL-Adresse, über die der externe Partner den Web-Service aufruft, die öffentliche Web-Service-URL ist, die beim Hochladen des Web-Service angegeben wurde. WebSphere Partner Gateway fungiert als Proxy. Es empfängt eine SOAP-Nachricht vom Partner und ermittelt den entsprechenden privaten Web-Service. Anschließend ruft es den

privaten (vom internen Partner bereitgestellten) Web-Service mit Hilfe derselben SOAP-Nachricht auf. Die vom internen Partner gelieferte Antwort wird dann an den externen Partner zurückgegeben.

- Von externen Partnern bereitgestellte Web-Services können dem internen Partner zur Verfügung gestellt werden.

Es ist wichtig zu beachten, dass die gleiche Web-Service-Schnittstelle von mehreren Partnern bereitgestellt werden kann. WebSphere Partner Gateway macht den Web-Service für den internen Partner über die Web-Service-URL verfügbar, die beim Hochladen des Web-Service in der Community Console angegeben wurde. Zusätzlich muss der interne Partner den URL-Parameter bereitstellen, um den Empfängerpartner zu identifizieren. Weitere Informationen finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*. WebSphere Partner Gateway fungiert als Proxy. Es empfängt eine SOAP-Nachricht vom internen Partner und ermittelt den entsprechenden privaten Web-Service sowie den Empfängerpartner. Dann ruft es den vom Partner bereitgestellten Web-Service mit Hilfe der gleichen SOAP-Nachricht auf. Die vom Partner gelieferte Antwortnachricht wird dann an den internen Partner zurückgegeben.

Weitere Informationen hierzu sowie Informationen zum Festlegen der Dokumentenflussdefinitionen für Web-Services finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

## cXML

Sie können cXML-Dokumente an Ihre externe Partner senden oder von ihnen empfangen. Wenn WebSphere Partner Gateway ein cXML-Dokument von einem externen Partner empfängt, wird das Dokument geprüft und konvertiert (falls angegeben), bevor es an das Back-End-System des internen Partners gesendet wird. Beachten Sie, dass die Konvertierung nicht für synchrone cXML-Nachrichten zu verwenden ist. Bei einem synchronen Austausch generiert das Back-End-System eine Antwort, die von WebSphere Partner Gateway an den externen Partner zurückgegeben wird (falls für die Nachricht zutreffend).

Ein Back-End-System beim internen Partner, das ein cXML-Dokument senden muss, hat zwei Möglichkeiten:

- Generieren und Senden eines cXML-Dokuments, das von WebSphere Partner Gateway an den externen Partner weitergeleitet wird
- Generieren und Senden eines XML-Dokuments, das von WebSphere Partner Gateway in cXML konvertiert wird, bevor es an den externen Partner weitergeleitet wird

**Anmerkung:** Wenn die Konvertierung für XML-Dokumente verwendet wird, wird die Antwort bei synchronen Anforderungs-/Antworttransaktionen mit dem externen Partner asynchron an das Back-End-System zurückgegeben.

Weitere Informationen hierzu sowie Informationen zum Festlegen von Dokumentenflussdefinitionen für cXML finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

## EDI

WebSphere Partner Gateway empfängt EDI-Dokumente von Partnern, die über ein VAN (Value-Added Network) oder das Internet auf dieses Produkt zugreifen. EDI-Dokumente, die an ein VAN gesendet oder von einem VAN empfangen wurden, arbeiten mit dem FTP-Scripting-Transport. Der FTP-Scripting-Transport kann auch zum Senden oder Empfangen von Dokumenten über das Internet verwendet wer-

den. Weitere Informationen zum FTP-Scripting-Transport finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

Ein EDI-Dokument erreicht und verlässt den Hub in einem EDI-Umschlag, der auch als *Austausch* (Interchange) bezeichnet wird. Der Austausch enthält einzelne EDI-Transaktionen oder Gruppen von Transaktionen.

Wird der EDI-Austausch über den Hub ausgeführt, ohne dass hierbei der Umschlag entfernt wird, erstellen Sie eine Verbindung zwischen dem Hub und dem internen Partner.

Wird der Umschlag des EDI-Austausches entfernt, dann unterscheidet sich die Vorgehensweise bei der Erstellung von Interaktionen und Verbindungen von der bei anderen Geschäftsprotokollen angewendeten Prozedur. Der Umschlag des Austauschs muss entfernt und die einzelnen Transaktionen müssen verarbeitet werden. Normalerweise werden die Transaktionen in ein anderes Format konvertiert. Dazu dient eine Transformationszuordnung, die vom Data Interchange Services-Client importiert wird. EDI-Transaktionen, die in XML- oder ROD/flat-Dokumente konvertiert werden, werden direkt an den internen oder externen Partner gesendet. Transaktionen, die in andere EDI-Formate konvertiert werden, werden zuerst mit einem Umschlag versehen und dann an den internen oder den externen Partner gesendet.

### Dokumentenfluss von der Back-End-Anwendung zum Partner

Eine Back-End-Anwendung kann folgende Dokumenttypen versenden:

- Einzelner EDI-Austausch mit einer oder mehreren Transaktionen

WebSphere Partner Gateway entfernt den Umschlag der einzelnen EDI-Transaktionen und setzt diese individuellen Transaktionen um. Wenn die Transaktionen ins EDI-Format konvertiert werden, fügt das System diese in einen Umschlag ein und leitet sie dann an den Partner weiter. Die Back-End-Anwendung kann die Pakettypen 'None' und 'Backend Integration' verwenden und den Austausch über eine Vielzahl von Transportprotokollen versenden, die in Tabelle 17 auf Seite 38 aufgelistet sind.

In Abb. 6 ist ein EDI-X12-Austausch dargestellt, der aus drei Transaktionen besteht, deren Umschlag entfernt wird. Die Transaktionen werden ins EDIFACT-Format transformiert und dann wieder in einen Umschlag eingefügt und an den gewünschten Partner gesendet.

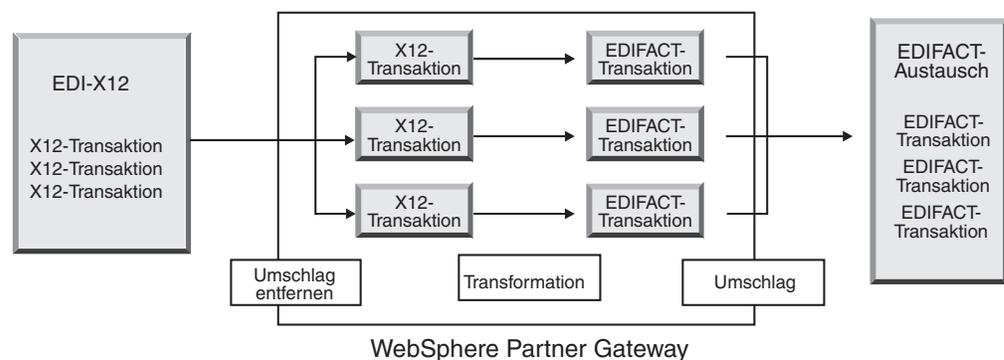


Abbildung 6. EDI-Austausch von der Back-End-Anwendung zum Partner

Alle Transaktionen verfügen über eine zugehörige Transformationszuordnung, in der definiert ist, wie die Transaktion transformiert wird. Die Transaktion kann in eine einzelne Transaktion oder (bei Verwendung einer Zuordnungsverkettung während der Zuordnungserstellung) in mehrere Transaktionen transformiert werden.

Wird die Transaktion in ein XML- oder ROD/flat-Dokument konvertiert, erfolgt das Routing auf der Basis der Konfigurationseinstellungen, die in der Anzeige **Partnerverbindungen** der Transaktion festgelegt sind.

Abb. 7 zeigt einen EDI-X12-Austausch, dessen Umschlag entfernt und der anschließend in XML-Dokumente transformiert wird, die dann an den Partner gesendet werden.

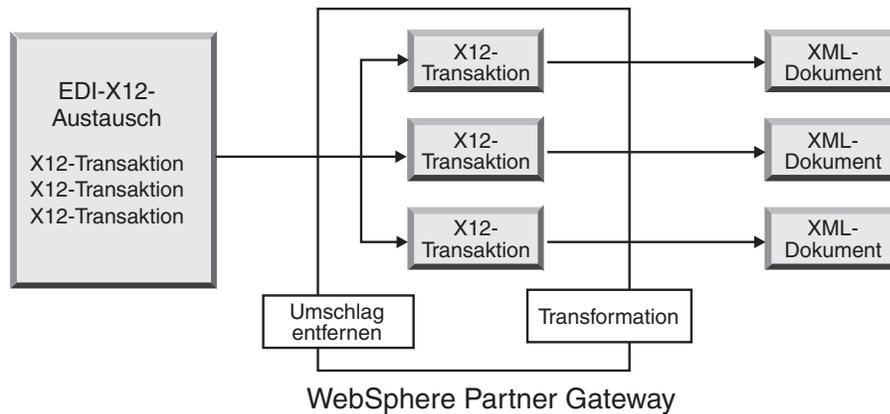


Abbildung 7. EDI-X12-Austausch von der Back-End-Anwendung an den Partner (als XML-Dokumente)

Die Transaktion kann in ein einzelnes Dokument oder (bei Verwendung einer Zuordnungsverkettung während der Zuordnungserstellung) in mehrere Dokumente transformiert werden.

- Einzelnes Dokument, zum Beispiel ein XML- oder ROD/flat-Dokument

WebSphere Partner Gateway konvertiert das Dokument in eine EDI-Transaktion, fügt diese in einen Umschlag ein und sendet sie an den Partner. Die Back-End-Anwendung kann die Pakettypen 'None' und 'Backend Integration' verwenden und das Dokument über eine Vielzahl von Transportprotokollen versenden, die in Tabelle 17 auf Seite 38 aufgelistet sind.

Abb. 8 auf Seite 13 zeigt ein XML-Dokument, das in X12-Transaktionen transformiert und dann wieder in einen Umschlag eingefügt wird.

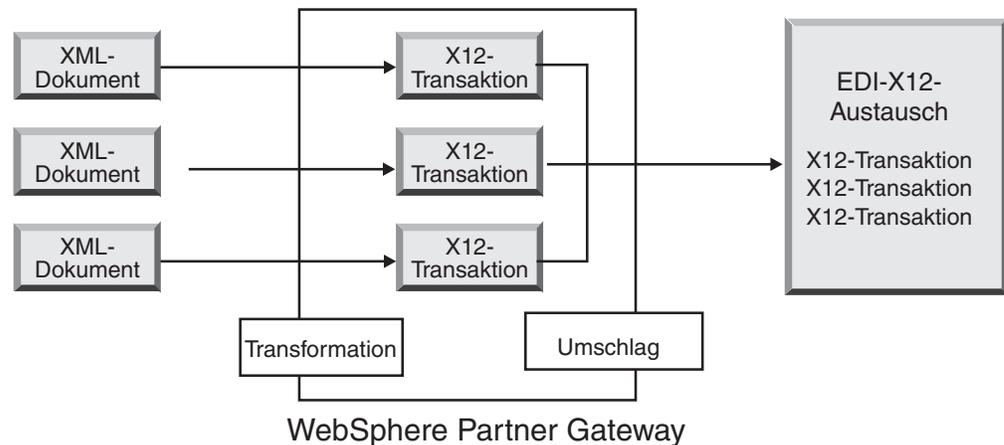


Abbildung 8. Von der Back-End-Anwendung an den Partner gesendete XML-Dokumente (als EDI-X12-Austausch)

Ein Dokument kann hierbei in mehrere Transaktionen transformiert werden (wenn bei der Zuordnungserstellung die Zuordnungsverkettung eingesetzt wurde). Die Transaktionen können anschließend in Umschläge für unterschiedliche Austauschelemente eingefügt werden.

Abb. 9 zeigt ein XML-Dokument, das in drei X12-Transaktionen transformiert wird. Hierbei werden zwei der Transaktionen in denselben Umschlag eingefügt. Die dritte Transaktion erhält einen separaten Umschlag.

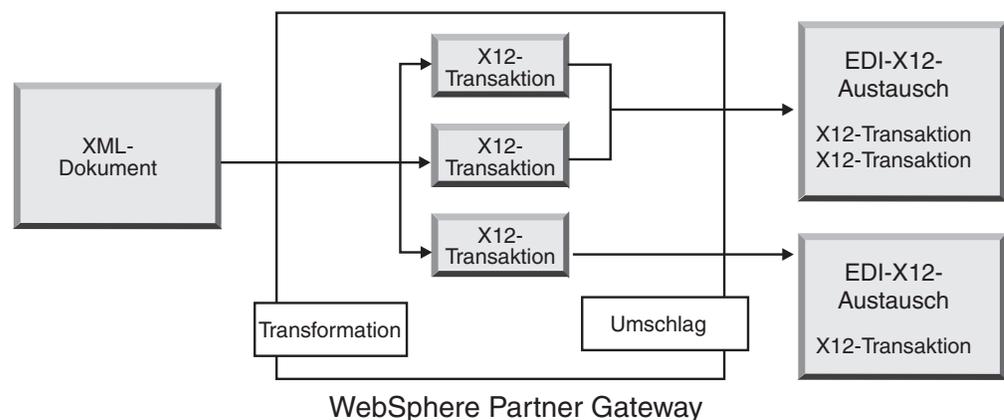


Abbildung 9. Von der Back-End-Anwendung an den Partner gesendetes XML-Dokument (als EDI-X12-Austausch)

Wird das Dokument in ein anderes XML- oder ROD/flat-Dokument konvertiert, erfolgt das Routing auf der Basis der Konfigurationseinstellungen, die in der Anzeige **Partnerverbindungen** für das Dokument festgelegt sind.

- Einzelne Datei mit mehreren XML- oder ROD/flat-Dokumenten

WebSphere Partner Gateway teilt die Dokumente auf und konvertiert sie. Wenn die Dokumente in EDI-Transaktionen konvertiert werden, fügt WebSphere Partner Gateway diese Transaktionen in einen Umschlag ein und sendet dann den Umschlag an den Partner. Wurden den XML- oder ROD/flat-Dokumenten Stapelverarbeitungs-IDs zugeordnet, versucht WebSphere Partner Gateway, die EDI-Transaktionen (als Stapelverarbeitungselemente) in einem Umschlag zu versenden. Die Back-End-Anwendung kann die Pakettypen 'None' und 'Backend

Integration' verwenden und das Dokument über eine Vielzahl von Transportprotokollen versenden, die in Tabelle 17 auf Seite 38 aufgelistet sind.

Abb. 10 zeigt eine Gruppe von XML-Dokumenten, die aufgeteilt werden. Auf diese Weise werden mehrere separate XML-Dokumente erstellt. Die XML-Dokumente werden in X12-Transaktionen transformiert, die Transaktionen anschließend in Umschläge eingefügt.

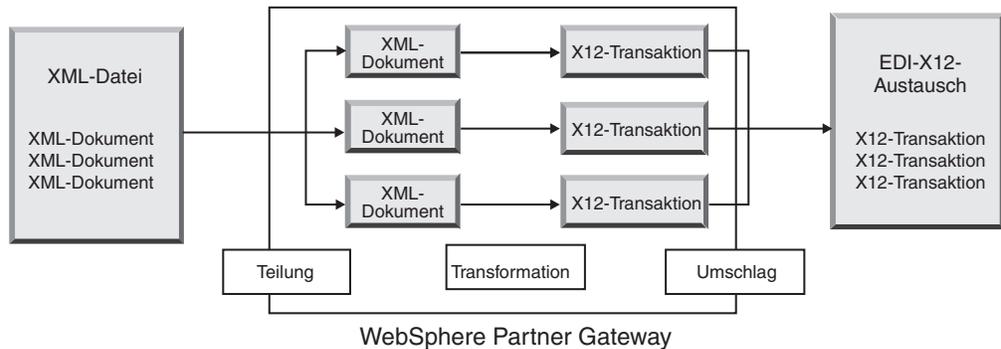


Abbildung 10. Mehrere von der Back-End-Anwendung gesendete XML-Dokumente, die aufgeteilt und dann an den Partner gesendet werden (als EDI-X12-Austausch)

Abb. 10 zeigt, wie die Dokumente aufgeteilt und die transformierten Transaktionen dann gemeinsam in einen Umschlag eingefügt werden. Um die Aufteilung von Dokumenten zu ermöglichen, müssen Sie für den Empfänger, an den die Dokumente gesendet werden sollen, eine entsprechende Aufteilungsroutine (den sog. Splitter Handler) konfigurieren. Im vorliegenden Fall wird der XML Splitter Handler verwendet. Beim XML Splitter Handler muss die Option BCG\_BATCHDOCS auf den Standardwert ON gesetzt werden, damit das dargestellte Szenario gilt. BCG\_BATCHDOCS dient zur Zuordnung einer Stapelverarbeitungs-ID zu den XML-Dokumenten, so dass die resultierenden Transaktionen in denselben Umschlag eingefügt werden können. Weitere Informationen zum XML Splitter Handler und zum Attribut BCG\_BATCHDOCS finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

Werden die Dokumente in andere XML- oder ROD/flat-Dokumente konvertiert, erfolgt das Routing auf der Basis der Konfigurationseinstellungen, die in der Anzeige **Partnerverbindungen** für die Dokumente festgelegt sind.

- Einzelne Datei mit mehreren EDI-Austauschelementen

WebSphere Partner Gateway teilt die Datei in mehrere separate Austausch-elemente auf. Anschließend werden die Austausch-elemente aus den Umschlägen entfernt und in einzelne Transaktionen aufgeteilt und dann konvertiert. Wenn die Dokumente in EDI-Transaktionen konvertiert werden, fügt WebSphere Partner Gateway diese Transaktionen in einen Umschlag ein und sendet dann den Umschlag an den Partner. Die Back-End-Anwendung kann die Pakettypen 'None' und 'Backend Integration' verwenden und das Dokument über eine Vielzahl von Transportprotokollen versenden, die in Tabelle 17 auf Seite 38 aufgelistet sind.

Werden die Dokumente in XML- oder ROD/flat-Dokumente konvertiert, erfolgt das Routing auf der Basis der Konfigurationseinstellungen, die in der Anzeige **Partnerverbindungen** für die Dokumente festgelegt sind.

## Dokumentenfluss vom Partner zur Back-End-Anwendung

Ein Partner kann die folgenden Dokumenttypen senden:

- Einzelner EDI-Austausch mit einer oder mehreren Transaktionen

WebSphere Partner Gateway entfernt den Umschlag der einzelnen EDI-Transaktionen und setzt diese Transaktionen um. Wenn die Transaktionen ins EDI-Format konvertiert werden, fügt das System diese in einen Umschlag ein und leitet sie dann an die Back-End-Anwendung weiter. Die Back-End-Anwendung kann die Pakettypen 'None' und 'Backend Integration' verwenden und die Transaktionen über eine Vielzahl von Transportprotokollen versenden, die in Tabelle 18 auf Seite 39 aufgelistet sind.

Werden die Transaktionen in XML- oder ROD/flat-Dokumente konvertiert, erfolgt das Routing auf der Basis der Konfigurationseinstellungen, die in der Anzeige **Partnerverbindungen** für die Transaktionen festgelegt sind.

- Einzelnes Dokument, zum Beispiel ein XML- oder ROD/flat-Dokument

WebSphere Partner Gateway setzt das Dokument in eine EDI-Transaktion um, fügt diese in einen Umschlag ein und sendet den Umschlag an die Back-End-Anwendung. Hierbei kann der Pakettyp 'None' oder 'Backend Integration' verwendet werden.

Wird das Dokument in ein XML- oder ROD/flat-Dokument konvertiert, erfolgt das Routing auf der Basis der Konfigurationseinstellungen, die in der Anzeige **Partnerverbindungen** für das Dokument festgelegt sind.

- Einzelne Datei mit mehreren XML- oder ROD/flat-Dokumenten

WebSphere Partner Gateway teilt die Dokumente auf und konvertiert sie. Wenn die Dokumente in EDI-Transaktionen konvertiert werden, fügt WebSphere Partner Gateway diese Transaktionen in einen Umschlag ein und sendet dann den Umschlag an die Back-End-Anwendung. Wenn den XML- oder ROD/flat-Dokumenten Stapelverarbeitungs-IDs zugeordnet wurden, versucht WebSphere Partner Gateway, die EDI-Transaktionen (als Stapelverarbeitungselemente) in einem Umschlag zu versenden. Hierbei kann der Pakettyp 'None' oder 'Backend Integration' verwendet werden.

Werden die Dokumente in andere XML- oder ROD/flat-Dokumente konvertiert, erfolgt das Routing auf der Basis der Konfigurationseinstellungen, die in der Anzeige **Partnerverbindungen** für die Dokumente festgelegt sind.

- Einzelne Datei mit mehreren EDI-Austauschelementen

WebSphere Partner Gateway teilt die Datei in mehrere separate Austausch-elemente auf. Anschließend werden die Austausch-elemente aus den Umschlägen entfernt und in einzelne Transaktionen aufgeteilt und dann konvertiert. Wenn die Dokumente in EDI-Transaktionen konvertiert werden, fügt WebSphere Partner Gateway diese Transaktionen in einen Umschlag ein und sendet dann den Umschlag an die Back-End-Anwendung. Hierbei kann der Pakettyp 'None' oder 'Backend Integration' verwendet werden.

Werden die Dokumente in XML- oder ROD/flat-Dokumente konvertiert, erfolgt das Routing auf der Basis der Konfigurationseinstellungen, die in der Anzeige **Partnerverbindungen** für die Dokumente festgelegt sind.

## Funktionale Bestätigungen

Eine funktionale Bestätigung gibt an, dass ein EDI-Austausch empfangen wurde. Sie wird vor dem Versenden immer in einen Umschlag eingefügt.

**Anmerkung:** Funktionale Bestätigungen gelten nur für die Austausch-elemente, die von WebSphere Partner Gateway aus dem Umschlag entfernt oder von WebSphere Partner Gateway generiert wurden. Für Austausch-elemente, die lediglich über WebSphere Partner Gateway weitergeleitet wurden, gelten funktionale Bestätigungen hingegen nicht.

Für von WebSphere Partner Gateway empfangene Austauschelemente gilt Folgendes:

- Wenn der Austausch von einer Back-End-Anwendung empfangen wird, kann WebSphere Partner Gateway funktionale Bestätigungen zurück an die Back-End-Anwendung senden.
- Wenn der Austausch von einem Partner empfangen wird, kann WebSphere Partner Gateway funktionale Bestätigungen zurück an den Partner senden.

Für von WebSphere Partner Gateway generierte Austauschelemente gilt Folgendes:

- Wird der Austausch an einen Partner gesendet, kann dieser eine funktionale Bestätigung zurück an WebSphere Partner Gateway senden. Diese funktionale Bestätigung wird nicht von WebSphere Partner Gateway an das Back-End-System gesendet.
- Wird der Austausch an die Back-End-Anwendung gesendet, kann diese eine funktionale Bestätigung zurück an WebSphere Partner Gateway senden. Diese funktionale Bestätigung wird von WebSphere Partner Gateway nicht an den Partner gesendet.

## RosettaNet

WebSphere Partner Gateway unterstützt das Senden und Empfangen von Dokumenten, die den Standards RosettaNet 1.1 und 2.0 entsprechen. Wenn ein Partner eine RosettaNet-Nachricht an den Hub sendet, dann muss auf der Zieleinheit der Partnerverbindung als Pakettyp 'Backend Integration' angegeben sein. Der Hub konvertiert die Nutzdaten der Nachricht ins RNSC-Format und sendet die Nachricht an das Back-End-System. Da der Pakettyp 'Backend Integration' verwendet wird, fügt der Hub Header der Transportebene zur Nachricht hinzu. Die Nachricht wird anschließend über das HTTP- oder das JMS-Transportprotokoll übertragen. Der Header der Transportebene enthält Metainformationen, die nicht Teil des PIP (Partner Interface Process) sind, und gibt WebSphere Partner Gateway die Möglichkeit, die Nachricht entsprechend weiterzuleiten.

Wenn das Back-End-System des internen Partners eine RNSC-Nachricht an den Hub sendet, muss auf der Quelleneinheit der Partnerverbindung der Pakettyp 'Backend Integration' angegeben worden sein. Das Back-End-System muss in diesem Fall auch die Header der Transportebene bereitstellen.

Nehmen Sie zum Beispiel an, eine Anwendung will eine Nachricht an einen externen Partner unter Verwendung von RosettaNet über HTTP senden. Die Anwendung stellt den RosettaNet Service Content bereit und fügt den Header der Transportebene hinzu. Der Header gibt unter anderem den externen Partner, der die Anforderung verarbeiten soll, den PIP, der gesendet wird, sowie die Version des PIP an. Diese Informationen geben WebSphere Partner Gateway die Möglichkeit, den richtigen PIP an den externen Partner zu senden.

Informationen zur Einrichtung der RosettaNet-Unterstützung und zur Konfiguration von PIPs finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

## Ereignisbenachrichtigung

WebSphere Partner Gateway führt RNIF-PI-Prozesse mit externen Partnern für die Back-End-Anwendungen des internen Partners aus. Aus diesem Grund stellt WebSphere Partner Gateway einen Mechanismus zur *Ereignisbenachrichtigung* bereit, um die Back-End-Anwendung über verschiedene Aspekte der Ausführung des RNIF-PIP-Prozesses zu informieren.

Die Ereignisbenachrichtigung ermöglicht es WebSphere Partner Gateway z. B., die Anwendung darüber zu informieren, ob ein PIP an den Partner gesendet werden kann oder nicht. Bei Bedarf kann die Anwendung dann geeignete Fehlerbehebungsmaßnahmen durchführen.

Eine Ereignisbenachrichtigung ist ein XML-Dokument, das Informationen über Ereignisse transportiert, die in WebSphere Partner Gateway oder in einer Anwendung aufgetreten sind. Diese Nachrichten weisen die gleiche Struktur wie alle anderen Nachrichten auf, die von WebSphere Partner Gateway gesendet oder empfangen werden. Dies bedeutet, dass sie einen Header der Transportebene und die Nutzdaten enthalten. WebSphere Partner Gateway kann so konfiguriert werden, dass Ereignisbenachrichtigungen gesendet oder nicht gesendet werden, da diese Nachrichten optional sind.

In Tabelle 2 sind die Ereignisbenachrichtigungen zusammengefasst, die von WebSphere Partner Gateway an Back-End-Systeme gesendet werden können.

*Tabelle 2. Ereignisbenachrichtigungen an das Back-End-System*

Ereignisbedingung	Ereignisbenachrichtigung
WebSphere Partner Gateway stellt ein RosettaNet-Dokument einem externen Partner zu und erhält eine Empfangsbestätigung.	Ereignis-Statuscode 100
Allgemeiner Fehler bei der Ausführung der RNIF-Aktion. WebSphere Partner Gateway bricht einen PIP ab, indem eine Nachricht 0A1 generiert und dem externen Partner zugestellt wird.  Wenn dies die letzte Aktion im PIP ist, wird WebSphere Partner Gateway durch diesen Aktionscode dazu veranlasst, den PIP abzurechnen, indem eine Nachricht 0A1 generiert und dem externen Partner zugestellt wird. Andernfalls sendet WebSphere Partner Gateway eine allgemeine Ausnahmebedingung an den Partner.	Ereignis-Statuscode 800
Fehler bei der Ausführung der RNIF-Aktion. Der Service-Content wird anhand eines PIP-definierten Wörterverzeichnisses überprüft. WebSphere Partner Gateway bricht einen PIP ab, indem eine Nachricht 0A1 generiert und dem externen Partner zugestellt wird.	Ereignis-Statuscode 801
WebSphere Partner Gateway empfängt eine Ausnahmebedingung im Zusammenhang mit einer Empfangsbestätigung oder eine allgemeine Ausnahmebedingung von einem externen Partner.	Ereignis-Statuscode 900

WebSphere Partner Gateway kann 0A1-Nachrichten an die Zielanwendung senden, wie dies auch für jeden anderen PIP geschieht, wenn über die Ausschlusslistenverwaltung das Senden dieser Nachrichten konfiguriert wurde. Weitere Informationen finden Sie im Abschnitt zur Verwaltung von Ausschlusslisten im Handbuch *WebSphere Partner Gateway Verwaltung*.

Eine Anwendung kann eine Ereignisbenachrichtigung an WebSphere Partner Gateway senden, um einen RosettaNet-PIP abzurechnen.

Informationen zu den von WebSphere Partner Gateway verwendeten Ereigniscodes finden Sie im Abschnitt „Von WebSphere Partner Gateway verwendete Ereigniscodes“ auf Seite 25.

## ebMS

WebSphere Partner Gateway unterstützt das Senden und Empfangen von Dokumenten, die den Standards von ebMS 2.0 entsprechen. Wenn ein Partner eine ebMS-Nachricht an den Hub sendet, dann muss auf der Zieleinheit der Partnerverbindung als Pakettyp 'Backend Integration' angegeben sein. Der Hub extrahiert die Nutzdaten der Nachricht und sendet die Nachricht an das Back-End-System. Da der Pakettyp 'Backend Integration' verwendet wird, fügt der Hub Header der Transportebene zur Nachricht hinzu. Die Nachricht wird anschließend über das HTTP- oder das JMS-Transportprotokoll übertragen. Der Header der Transportebene enthält Metainformationen, die nicht Teil der ebMS-Nachricht sind, und gibt WebSphere Partner Gateway die Möglichkeit, die Nachricht entsprechend weiterzuleiten.

Wenn das Back-End-System des internen Partners Nutzdaten an den Hub sendet, muss auf der Quelleneinheit der Partnerverbindung als Pakettyp 'Backend Integration' angegeben sein. Das Back-End-System muss in diesem Fall auch die Header der Transportebene bereitstellen.

Nehmen Sie zum Beispiel an, eine Anwendung will über HTTP eine Nachricht an einen externen Partner unter Verwendung von ebMS senden. Die Anwendung stellt ebMS-Service, Aktion, Partner-IDs (Empfänger und Absender), Protokoll- und Paketdaten, Nachrichten-ID sowie Dialog-ID bereit und fügt die Header der Transportebene hinzu. Der Header gibt an, welcher externe Partner die Anforderung verarbeiten soll. Mit Hilfe dieser Informationen kann WebSphere Partner Gateway die ebMS-Nachricht erstellen und die Nutzdaten aus dem Back-End-System als ebMS-Nutzdaten an den externen Partner senden.

Die Header in Tabelle 3 müssen vom Back-End-System für den ebMS-Fluss gesetzt werden.

*Tabelle 3. Für den ebMS-Fluss erforderliche Header des Back-End-Systems*

Headername	Erläuterung	Beispielwert	Erforderlich
x-aux-sender-id	Kennung des Nachrichtenabsenders, zum Beispiel eine DUNS-Nummer.	987654321	Ja
x-aux-receiver-id	Kennung des Nachrichtempfängers, zum Beispiel eine DUNS-Nummer.	123456789	Ja
x-aux-protocol	Protokoll des Nachrichteninhalts. Der Wert muss mit dem Wert für das Protokoll auf der Quellenseite in der Anzeige <b>Partnerverbindungen</b> übereinstimmen.	ebMS	Ja
x-aux-protocol-version	Version des Protokolls für den Nachrichteninhalt. Der Wert muss mit der Protokollversion auf der Quellenseite in der Anzeige <b>Partnerverbindungen</b> übereinstimmen.	2.0	Ja
x-aux-process-type	Der auszuführende Prozess bzw. der Typ der gesendeten Nachricht. Der Wert muss mit dem Wert für den Dokumenttyp auf der Quellenseite in der Anzeige <b>Partnerverbindungen</b> übereinstimmen.	Purchase Order Request	Ja

Tabelle 3. Für den ebMS-Fluss erforderliche Header des Back-End-Systems (Forts.)

Headername	Erläuterung	Beispielwert	Erforderlich
x-aux-process-version	Version des Prozesses. Der Wert muss mit der Version des Dokumenttyps auf der Quellenseite in der Anzeige <b>Partnerverbindungen</b> übereinstimmen. Dieser Wert entspricht dem Wert für den Servicetyp in der ebXML-Nachricht.	ALL	Ja
x-aux-activity	Die Aktivität, die für den Typ der gesendeten Nachricht ausgeführt werden muss. Der Wert muss mit dem Wert für die Aktivität auf der Quellenseite in der Anzeige <b>Partnerverbindungen</b> übereinstimmen. Für den ebMS-Fluss entspricht dieser Wert x-aux-process-type.	Purchase Order Request	Ja
x-aux-action	Die Aktion, die für die ebMS-Nachricht ausgeführt werden soll. Dieser Wert entspricht dem Aktionselement in der ebXML-SOAP-Nachricht. Darüber hinaus muss dieser Wert mit dem Aktionswert auf der Quellenseite in der Anzeige <b>Partnerverbindungen</b> übereinstimmen.	Purchase Order Action	Ja
x-aux-system-msg-id	Globale eindeutige Kennung (Global Unique Identifier - GUID) für die Nachricht, die zur Duplikatprüfung dient. Diese ID wird von WebSphere Partner Gateway für die Duplikatprüfung verwendet. In einem ebMS-Fluss wird sie nicht verwendet.	28282828282828	Ja
x-aux-production	Routing der Nachricht. Gültige Werte: Production und Test. Dieser Wert wird für Anforderungen in beide Richtungen eingetragen.	Production	Ja
x-aux-msg-id	Eindeutige Nachrichten-ID. Wird in der ebXML-Soap-Nachricht als Nachrichten-ID (MessageID) verwendet.	123454321@abc.com	Nein
contentType	Der Inhaltstyp (content-Type) der Nachricht. Dieser Wert ist nicht erforderlich, wenn Sie eine Nachricht vom Back-End-Senden senden, die in das WBI-Paket eingeschlossen ist.	text/xml	Nein
x-aux-process-instance-id	Die Prozess-ID (processId) oder Dialog-ID (ConversationId) in der ebXML-Nachricht.	329878738@abc.com	Nein

Tabelle 3. Für den ebMS-Fluss erforderliche Header des Back-End-Systems (Forts.)

Headername	Erläuterung	Beispielwert	Erforderlich
x-aux-seq-number	Die Folgenummer der Nachricht, wenn "Semantik der Nachrichtenreihenfolge" auf der Zielseite der Dokumentenflussdefinition auf "Garantiert" gesetzt ist. Dies ist die Reihenfolge der ausgehenden Nachrichten in diesem Datenaustausch.	2	Nein
x-aux-request-msg-id	Dieser Header sollte nur für das Antwortdokument gesetzt werden. Der Wert für diesen Header ist die Nachrichten-ID des Anforderungsdokuments, das dem aktuellen Antwortdokument zugeordnet ist.	123455521@abc.com	Nein
x-aux-role	Dieses Feld ist bei Verwendung des WBI-Pakets 1.2 nicht erforderlich, da der Benutzer das Attribut in diesem Fall in der XML selbst festlegen kann. Ansonsten entspricht der Wert für dieses Feld dem Wert des Rollenattributs im Manifest/Reference-Element in der ebXML-SOAP-Nachricht.	http://reqrep.org/gci/purchaseOrder	Nein
x-aux-role-schema	Das Schema, anhand dessen die Rolle validiert wird. Diese Validierung wird nicht von WebSphere Partner Gateway durchgeführt. Der Wert für dieses Feld wird als Schemaattribut in die ebXML-SOAP-Nachricht eingefügt.	http://reqrep.org/gci/purchaseOrder/po.xsd	Nein
x-aux-role-schema-version	Die Version des obigen Schemas.	2.0	Nein
x-aux-description	Die Beschreibung der Nutzdaten. Dieser Wert wird als Wert des Beschreibungselements (description) in die ebXML-SOAP-Nachricht eingefügt.	Purchase Order Request	Nein

Informationen zur Einrichtung der ebMS-Unterstützung und zur Konfiguration von ebMS-Verbindungen finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

## Ereignisbenachrichtigung

WebSphere Partner Gateway stellt einen Mechanismus zur *Ereignisbenachrichtigung* bereit, um die Back-End-Anwendung über verschiedene Aspekte der Ausführung des ebMS-Prozesses zu informieren. Weitere Details zur Ereignisbenachrichtigung finden Sie im Abschnitt zu Ereignisbenachrichtigungen/XML-Ereignissen (EventNotification/XMLEvent).

In Tabelle 4 auf Seite 21 sind die Ereignisbenachrichtigungen zusammengefasst, die von WebSphere Partner Gateway an das Back-End-System für ebMS (und umgekehrt) gesendet werden.

Tabelle 4. Ereignisbenachrichtigungen an das Back-End-System

Ereignisbedingung	Ereignisbenachrichtigung
WebSphere Partner Gateway generiert dieses Ereignis nach Erhalt der Empfangsbestätigung von einem Partner.	Ereignis-Statuscode 100
WebSphere Partner Gateway empfängt eine Ausnahmebedingung im Zusammenhang mit einer Empfangsbestätigung oder eine allgemeine Ausnahmebedingung von einem externen Partner.	Ereignis-Statuscode 900
WebSphere Partner Gateway generiert dieses Ereignis, wenn alle Wiederholungen erfolgt sind oder die Lebensdauer der Nachricht abgelaufen ist und keine Bestätigung für die Nachricht empfangen wurde.	Ereignis-Statuscode 902
WebSphere Partner Gateway generiert dieses Ereignis im Falle eines Übermittlungsfehlers. Das heißt, WebSphere Partner Gateway kann die Nachricht nicht an einen Partner senden, weil das Ziel des Partners inaktiv ist und alle Wiederholungen auf Transport- und Zielebene erfolgt sind.	Ereignis-Statuscode 903

Informationen zu den von WebSphere Partner Gateway verwendeten Ereigniscodes finden Sie im Abschnitt „Von WebSphere Partner Gateway verwendete Ereigniscodes“ auf Seite 25.

## XML-Ereignis (XMLEvent)

WebSphere Partner Gateway bietet eine Möglichkeit zur Benachrichtigung des Back-End-Systems über den Status der Nachricht. Wenn WebSphere Partner Gateway beispielsweise eine Nachricht an einen Partner sendet und der Partner eine Empfangsbestätigung zurücksendet, generiert WebSphere Partner Gateway ein XML-Ereignis mit dem Ereigniscode 100 (eine Liste der Ereigniscodes finden Sie im Abschnitt „Von WebSphere Partner Gateway verwendete Ereigniscodes“ auf Seite 25) und sendet dieses Ereignis an das Back-End-System. Die Struktur des XML-Ereignisses wird im Abschnitt „Struktur von Ereignisnachrichten“ auf Seite 22 beschrieben. XML-Ereignisse werden nur zwischen WebSphere Partner Gateway und einer Back-End-Anwendung verwendet. Sie werden nicht an die Partner gesendet. In WebSphere Partner Gateway 6.1 werden XML-Ereignisse nur für RosettaNet- und ebMS-Flüsse generiert.

Sie können ein XML-Ereignis, das zwischen WebSphere Partner Gateway und einer Back-End-Anwendung erstellt wurde, aktivieren oder inaktivieren. Informationen hierzu finden Sie im Abschnitt „XML-Ereignisse aktivieren oder inaktivieren“.

### XML-Ereignisse aktivieren oder inaktivieren

Gehen Sie wie folgt vor, um ein XML-Ereignis (XMLEvent) zu aktivieren:

1. Aktivieren Sie die B2B-Funktionalität für den Datenfluss der RosettaNet- bzw. ebMS-Dokumentdefinition.

Die B2B-Funktionalität der Quelle muss wie folgt definiert sein:

Paket: Backend Integration (1.0)

Protokoll: XMLEvent (1.0)

Dokumenttyp: XMLEvent

Die B2B-Funktionalität des Ziels muss wie folgt definiert sein:

Paket: Backend Integration (1.0)  
Protokoll: XMLEvent (1.0)  
Dokumenttyp: XMLEvent

2. Erstellen Sie eine Interaktion zwischen der Quelle und dem Ziel, wie im Folgenden beschrieben:
  - a. Klicken Sie auf **Hubadmin > Hubkonfiguration > Dokumentdefinition**.
  - b. Klicken Sie auf **Interaktionen verwalten**.
  - c. Klicken Sie auf **Interaktion erstellen**.
  - d. Erweitern Sie unter **Quelle** den Eintrag **Paket: Backend Integration (1.0) > Protokoll: XMLEvent**.
  - e. Wählen Sie **Dokumenttyp: XMLEvent (1.0)** aus.
  - f. Wiederholen Sie die vorherigen Schritte (Schritt d und Schritt e) in der Spalte **Ziel**.
  - g. Wählen Sie in der Liste **Aktion** den Eintrag **Pass-Through** aus, und klicken Sie auf **Speichern**.
3. Aktivieren Sie das XML-Ereignis, indem Sie wie folgt vorgehen:
  - a. Klicken Sie auf **Kontenadmin > Verbindungen**. Die Seite **Verbindungen verwalten** wird angezeigt.
  - b. Wählen Sie die **Quelle** und das **Ziel** aus.
  - c. Klicken Sie auf **Suchen**, um die Verbindungen zu suchen, die mit Ihren Kriterien übereinstimmen.
  - d. Klicken Sie auf **Aktivieren**, um das XML-Ereignis zu aktivieren. Die Seite **Verbindungen verwalten** wird erneut angezeigt, und die gewünschte Verbindung wird mit einer grünen Markierung hervorgehoben. Auf der Seite werden das Paket, das Protokoll und der Dokumenttyp für die Quelle und das Ziel angezeigt. Sie können darüber hinaus auch den Status und die Parameter der Partnerverbindung anzeigen und ändern.

Führen Sie einen der folgenden Schritte aus, um das XML-Ereignis (XMLEvent) zu inaktivieren:

- Klicken Sie auf **Hubadmin > Hubkonfiguration > Dokumentdefinition > Interaktionen verwalten**, und wählen Sie für **Protokoll: XMLEvent** die Option **Inaktivieren** aus. ODER
- Führen Sie die folgenden Schritte aus:
  1. Klicken Sie auf **Kontenadmin > Ausschlussliste**.
  2. Wählen Sie einen Eintrag für **Partnername** aus.
  3. Klicken Sie auf **Bearbeiten**.
  4. Wählen Sie das Kontrollkästchen **Keine Back-End-Ereignisbenachrichtigungen senden** aus.

## Ereignisbenachrichtigung

Bei der RosettaNet- und ebMS-Verarbeitung gibt es Fälle, in denen ein XML-Ereignis entweder an das Back-End-System gesendet oder vom Back-End-System empfangen wird. Informationen zu den Fällen, in denen in diesen Protokollen ein XML-Ereignis verwendet wird, finden Sie im Abschnitt „Ereignisbenachrichtigung“ auf Seite 16 (für RosettaNet-Verarbeitung) bzw. im Abschnitt „Ereignisbenachrichtigung“ auf Seite 20 (für ebMS-Verarbeitung).

## Struktur von Ereignisnachrichten

Eine Ereignisbenachrichtigung verfügt über einen Standardheader der Transportebene, dessen Feld 'x-aux-process-type' auf den Wert XMLEvent gesetzt ist.

Allerdings verfügen die Nutzdaten der Nachricht über eine bestimmte Struktur, die in dem in Abb. 11 dargestellten XML-Schema gezeigt wird.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace=
    "http://www.ibm.com/websphere/bcg/2003/v1.0/xmleventnotification"
  xmlns:evntf=
    "http://www.ibm.com/websphere/bcg/2003/v1.0/xmleventnotification"
  elementFormDefault="qualified">
  <!-- EventNotification version 1.0 document element -->
    <xsd:element name="EventNotification">
      <xsd:complexType>
        <xsd:all>
          <xsd:element ref="evntf:StatusCode"/>
          <xsd:element ref="evntf:StatusMessage"/>
          <xsd:element ref="evntf:EventMessageID"/>
          <xsd:element ref="evntf:BusinessObjectID"/>
          <xsd:element ref="evntf:GlobalMessageID"/>
          <xsd:element ref="evntf:Timestamp"/>
        </xsd:all>
      </xsd:complexType>
    </xsd:element>
  <!-- StatusCode element -->
    <xsd:element name="StatusCode">
      <xsd:simpleType>
        </xsd:simpleType>
      </xsd:element>
  <!-- StatusMessage element -->
    <xsd:element name="StatusMessage">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string"/>
      </xsd:simpleType>
    </xsd:element>
  <!-- EventMessageID element -->
    <xsd:element name="EventMessageID">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string"/>
      </xsd:simpleType>
    </xsd:element>
  <!-- BusinessObjectID element -->
    <xsd:element name="BusinessObjectID">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string"/>
      </xsd:simpleType>
    </xsd:element>
  <!-- GlobalMessageID element -->
    <xsd:element name="GlobalMessageID">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string"/>
      </xsd:simpleType>
    </xsd:element>
  <!-- Timestamp element -->
    <xsd:element name="Timestamp">
      <xsd:simpleType>
        <xsd:restriction base="xsd:dateTime"/>
      </xsd:simpleType>
    </xsd:element>
</xsd:schema>
```

Abbildung 11. XML-Schema für eine Ereignisbenachrichtigung

In Tabelle 5 sind die einzelnen Felder der Ereignisnutzdaten beschrieben.

Tabelle 5. XML-Felder zur Ereignisbenachrichtigung

Feld	Beschreibung
StatusCode	Der Typ der Nachricht. Gültige Werte finden Sie im Abschnitt „Von WebSphere Partner Gateway verwendete Ereigniscodes“ auf Seite 25.
StatusMessage	Alphanumerische Beschreibung dieser Ereignisbenachrichtigung.
EventMessageID	Alphanumerische Kennung dieser speziellen Ereignisbenachrichtigung.
BusinessObjectID	Das Feld 'x-aux-msg-id' im Header der Transportebene der Nachricht, die von diesem Benachrichtigungsereignis betroffen ist. Dies stellt die Verbindung zu den Nutzdaten der ursprünglichen Nachricht zu diesem Ereignis her.
GlobalMessageID	Das Feld 'x-aux-system-msg-id' im Header der Transportebene der Nachricht, die dieses Benachrichtigungsereignis verursacht hat.
Timestamp	Gibt im WEZ-Zeitmarkenformat an, wann das Ereignis aufgetreten ist: CCYY-MM-DDThh:mm:ssZ  Dies schließt die Bruchteilgenauigkeit von Sekunden (...ss.ssssZ) mit ein. Die Datumszeitmarke muss dem Datentyp des XML-Schemas für 'dateTime' (w3.org/TR/2001/REC-xmlschema-2-20010502#dateTime) entsprechen.

## Beispiel für eine Ereignisbenachrichtigung

Abb. 12 zeigt ein Beispiel für eine mit dem HTTP-Protokoll gesendete Ereignisbenachrichtigung.

```
POST /builderURL HTTP/1.1
Content-Type: application/xml
Content-length: 250
x-aux-sender-id: 000000001
x-aux-receiver-id: 000000002
x-aux-third-party-bus-id: 000000003
x-aux-create-datetime: 2002-10-28T23:05:02Z
x-aux-protocol: XMLEvent
x-aux-protocol-version: 1.0
x-aux-process-type: XMLEvent
x-aux-process-version: 1.0
x-aux-payload-root-tag: evtntf:EventNotification
x-aux-msg-id: 98732
x-aux-system-msg-id: 12345
x-aux-production: Production
x-aux-process-instance-id: 3456
x-aux-event-status-code: 100
x-aux-transport-retry-count: 0
<?xml version="1.0" encoding="UTF-8"?>
<evtntf:EventNotification xmlns:evtntf=
"http://www.ibm.com/websphere/bcg/2003/v1.0/xmleventnotification">
  <evtntf:StatusCode>100</evtntf:StatusCode>
  <evtntf:StatusMessage>The message was delivered</evtntf:StatusMessage>
  <evtntf:EventMessageID>12345</evtntf:EventMessageID>
  <evtntf:BusinessObjectID>34234</evtntf:BusinessObjectID>
  <evtntf:GlobalMessageID>98732</evtntf:GlobalMessageID>
  <evtntf:Timestamp>2001-01-31T13:20:00Z</evtntf:Timestamp>
</evtntf:EventNotification>
```

Abbildung 12. Beispiel für eine Ereignisbenachrichtigung über HTTP

## Von WebSphere Partner Gateway verwendete Ereigniscodes

Dieser Abschnitt enthält eine Zusammenfassung aller von WebSphere Partner Gateway verwendeten Ereigniscodes. Informationen darüber, wie diese Ereigniscodes in den einzelnen Protokollen verwendet werden, finden Sie im Abschnitt „Ereignisbenachrichtigung“ auf Seite 16 (für RosettaNet-Protokolle) bzw. im Abschnitt „Ereignisbenachrichtigung“ auf Seite 20 (für ebMS-Protokolle).

In Tabelle 6 sind die von WebSphere Partner Gateway verwendeten Ereigniscodes zusammengefasst.

Tabelle 6. Von WebSphere Partner Gateway verwendete Ereigniscodes

Ereignisstatus-code	Verwendetes Protokoll in	Absender	Empfänger
100	RosettaNet ebMS	WebSphere Partner Gateway	Back-End-System
800	RosettaNet	Back-End-System	WebSphere Partner Gateway
801	RosettaNet	Back-End-System	WebSphere Partner Gateway
900	RosettaNet ebMS	WebSphere Partner Gateway	Back-End-System
902	ebMS	WebSphere Partner Gateway	Back-End-System
903	ebMS	WebSphere Partner Gateway	Back-End-System

---

## Zu verwendender Paketty

Der Paketty bestimmt das Format, in dem die Nachricht von WebSphere Partner Gateway an das Back-End-System gesendet wird. Außerdem wird durch den Paketty das Format angegeben, in dem das Back-End-System die Nachricht an WebSphere Partner Gateway sendet.

Über die Community Console können Sie die Verbindung mit Ihren externen Partnern einrichten und den Paketty angeben, der zwischen WebSphere Partner Gateway und dem Back-End-System verwendet wird. Welcher Paketty dabei zu verwenden ist, hängt von folgenden Gesichtspunkten ab:

- Welche Paketty sind für die Verwendung bei einem Back-End-System zulässig?
- Welche Paketty sind für eine Nachricht in einem bestimmten Geschäftsprotokoll zulässig?

Weitere Informationen zur Einrichtung von Partnerverbindungen finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

Nicht alle Paketty sind gültig, wenn Sie WebSphere Partner Gateway zur Integration verwenden. In Tabelle 7 sind die Paketty aufgelistet, die relevant sind, wenn WebSphere Partner Gateway Dokumente oder Nachrichten mit einer Back-End-Anwendung des internen Partners austauscht.

Tabelle 7. Für die Back-End-Integration relevante Pakettypen

Pakettyp	Beschreibung
Paket 'None'	Weist das System an, die Nachricht <i>ohne</i> Headerdaten an das Back-End-System oder den Hub zu senden.
Paket 'Backend Integration'	Fügt dem Nachrichtenheader zusätzliche Attribute hinzu und fügt (optional) den Nachrichteninhalte in einen XML-Transportumschlag ein.

**Anmerkung:** Mit WebSphere Partner Gateway stehen auch andere Pakettypen (wie AS) zur Verfügung. Für die Integration mit Back-End-Systemen werden jedoch nur die Typen 'None' und 'Backend Integration' empfohlen.

## None

Wenn 'None' festgelegt ist, wird von WebSphere Partner Gateway beim Senden einer Nachricht an ein Back-End-System weder ein Header der Transportebene hinzugefügt, noch ein solcher Header beim Empfang einer Nachricht von einem Back-End-System erwartet. Stattdessen sendet WebSphere Partner Gateway nur die Nachricht an das Back-End-System. Informationen innerhalb des Dokuments steuern das Routing.

## Back-End-Integrationspaket

Wenn der Pakettyp 'Backend Integration' verwendet wird, enthalten Nachrichten, die an ein Back-End-System gesendet oder von diesem empfangen werden, folgende Komponenten:

- Einen Header der Transportebene, der Metainformationen zur Nachricht enthält (erforderlich)
- Nutzdaten, die den eigentlichen Nachrichteninhalte enthalten (erforderlich)
- Einen Anhang (optional)

Der Header und die Nutzdaten sind verbindlich, während Anhänge optional sind. In den folgenden Abschnitten werden die einzelnen Komponenten eines Dokuments beschrieben, das ein Back-End-Integrationspaket verwendet.

### Inhalte des Headers der Transportebene

Der Header der Transportebene enthält Informationen, die von WebSphere Partner Gateway zum Verarbeiten und zum Weiterleiten der Nachricht an die korrekte Zieladresse verwendet werden. Der Header der Transportebene ist bidirektional, so dass alle Nachrichten, die von WebSphere Partner Gateway empfangen bzw. gesendet werden, über die verbindlichen Felder und alle relevanten optionalen Felder verfügen.

In Tabelle 8 sind die Felder des Headers der Transportebene aufgelistet.

Tabelle 8. Felder des Headers der Transportebene

Headerfeld	Beschreibung	Erforderlich
x-aux-sender-id	Kennung des Nachrichtenabsenders, zum Beispiel eine DUNS-Nummer.	Ja
x-aux-receiver-id	Kennung des Nachrichteneempfängers, zum Beispiel eine DUNS-Nummer.	Ja

Table 8. Felder des Headers der Transportebene (Forts.)

Headerfeld	Beschreibung	Erforderlich
x-aux-protocol	Protokoll des Nachrichteninhalts. Gültige Wert sind RNSC (RosettaNet Service Content), XMLEvent und Binary. Für WebSphere Partner Gateway hat der Wert in diesem Feld Vorrang vor allen Protokollfeldern in den Nutzdaten.	Ja
x-aux-protocol-version	Version des Protokolls für den Nachrichteninhalt.	Ja
x-aux-process-type	Der auszuführende Prozess bzw. der Typ der gesendeten Nachricht. Bei RosettaNet-Nachrichten ist dies der PIP-Code, zum Beispiel 3A4. Bei Ereignisnachrichten hat dieses Feld den Wert 'XMLEvent', bei binären Nachrichten 'Binary'. Für WebSphere Partner Gateway hat der Wert in diesem Feld Vorrang vor allen Prozessfeldern in den Nutzdaten.	Ja
x-aux-process-version	Version des Prozesses. Bei RosettaNet-Nachrichten ist dies die Versionsnummer des PIP.	Ja
x-aux-create-datetime	Gibt an, wann die Nachricht erfolgreich übergeben wurde (im WEZ-Zeitmarkenformat: CCYY-MM-DDThh:mm:ssZ).	
x-aux-msg-id	Die Kennung des Inhalts der Nutzdaten. Dies könnte zum Beispiel die Kennung der RNPIPServiceContent-Instanz bei einer RosettaNet-Nachricht oder eine proprietäre Dokumentkennung sein. Dieser Wert stellt zu Tracingzwecken die Verbindung der Nutzdaten der Nachricht zu einer Komponente des Systems des Nachrichtenabsenders her.	
x-aux-production	Routing der Nachricht. Gültige Werte: Production und Test. Dieser Wert wird für Anforderungen in beide Richtungen eingetragen. Beachten Sie, dass bei einer Nachricht, die eine Antwort auf einen von einem externen Partner eingeleiteten PIP in beide Richtungen ist, WebSphere Partner Gateway den Wert 'GlobalUsageCode' in der Anforderung verwendet und den Wert im Header der Transportebene ignoriert.	
x-aux-system-msg-id	Globale eindeutige Kennung (Global Unique Identifier - GUID) für die Nachricht, die zur Duplikatprüfung dient.	Ja
x-aux-payload-root-tag	Das 'root-tag'-Element der Nutzdaten. Für einen 3A4-RosettaNet Service Content wäre der Wert dieses Felds zum Beispiel 'Pip3A4PurchaseOrderRequest'. Bei Ereignisbenachrichtigungen wäre der Wert dieses Felds 'EventNotification'.	
x-aux-process-instance-id	Kennung, die Dokumente in einem Geschäftsprozess mit mehreren Nachrichten mit einer eindeutigen Prozessinstanz verbindet. Für RosettaNet muss dieser Wert für RosettaNet-Prozesse innerhalb der letzten 30 Tage eindeutig sein. Alle Nachrichten, die im Rahmen einer RosettaNet-Prozessinstanz ausgetauscht werden, einschließlich wiederholter Nachrichten, verwenden die gleiche Prozessinstanz-ID.	
x-aux-event-status-code	Statuscode für die Ereignisbenachrichtigung. Siehe das Feld 'StatusCode' im Abschnitt „Struktur von Ereignisnachrichten“ auf Seite 22.	
x-aux-third-party-bus-id	Kennung, zum Beispiel eine DUNS-Nummer der Partei, von der die Nachricht zugestellt wurde. Dieser Wert kann sich sowohl von 'x-aux-sender-id' als auch von 'x-aux-receiver-id' unterscheiden, wenn eine dritte Partei im Namen des Community-Eigners als Host für WebSphere Partner Gateway fungiert.	
x-aux-transport-retry-count	Anzahl der vor diesem Versuch nicht erfolgreichen Versuche, diese Nachricht zu übergeben. Wenn eine Nachricht beim ersten Versuch erfolgreich übergeben wird, erhält dieses Feld den Wert 0.	
x-aux-in-file-name	Der ursprüngliche Dateiname der Nachricht, die über JMS oder HTTP an WebSphere Partner Gateway gesendet wird. (Vgl. hierzu Anmerkung 3 auf Seite 28.)	Nein

Table 8. Fields of the Header of the Transport Layer (Forts.)

Headerfeld	Beschreibung	Erforderlich
x-out-filename	Der ursprüngliche Dateiname von Nachrichten, die über JMS gesendet werden und für die der Pakettyp 'Backend Integration' verwendet wird. (Vgl. hierzu Anmerkung 2.)	Nein
content-type	Der Inhaltstyp der Nachricht.	
content-length	Die Länge der Nachricht (in Byte).	
x-aux-SyncResponse	Im ebMS-Fluss wird dieses Attribut verwendet, um das Back-End-System darüber zu informieren, dass 'SyncResponse' für diese Anforderung erforderlich ist. Wenn das Back-End-System 'SyncResponse' nicht benötigt, ist es nicht vorhanden. Die folgenden Werte sind möglich: <b>Anmerkung:</b> Für alle Werte dieses Attributs gilt die Kleinschreibung. <ul style="list-style-type: none"> <li>• <b>signalonly:</b> Ein Signal kann ein Abnahmesignal oder ein Ausnahmesignal sein. Das Back-End-System legt fest, was ein Signal ist. Wenn der Attributwert 'signalonly' lautet, erwartet der Absenderpartner nur die Signale über eine synchrone Verbindung.</li> <li>• <b>responseonly:</b> Ein Antwortdokument ist eine Geschäftsantwort für das empfangene Dokument. Wenn der Wert 'responseonly' lautet, erwartet der Absenderpartner eine Antwort über eine synchrone Verbindung.</li> <li>• <b>signalsandresponse:</b> In diesem Fall werden Signale und Antwort über eine synchrone Verbindung gesendet. Dieses Attribut bezieht sich auf ein Routing-Objekt. Es ist im CPA vorhanden und kann auf der Ebene der Partnerverbindung gesetzt werden, wenn kein CPA vorhanden ist.</li> </ul>	Nein
x-aux-TimeToAccept	Dieses Attribut wird im ebMS-Fluss verwendet und enthält den Wert von timeToAcknowledgeAcceptance im CPA. Beim Hochladen des CPA wird dieser Wert als TimeToPerform (Attribut der Quelldokumentdefinition) festgelegt. Der Wert ist eine Anzahl von Minuten. Er gibt den Zeitraum an, in dem der Empfänger den Empfang eines Geschäftsdokuments bestätigen muss (d. h. nach erfolgter Geschäftsregelprüfung).	Nein
x-aux-IntelligibleCheckRequired	Dieses Attribut wird im ebMS-Fluss verwendet. Die möglichen Werte sind <b>Yes</b> (Ja) und <b>No</b> (Nein). Wenn der Wert <b>Yes</b> ist, muss das Back-End-System sicherstellen, dass ein Geschäftsdokument nicht fehlerhaft ist (d. h. die Schemaprüfung war erfolgreich), bevor ein Empfangsbestätigungssignal zurückgegeben wird.	

**Anmerkungen:**

1. Aus Gründen der Kompatibilität mit IBM WebSphere MQ (ein JMS-Provider) werden in den Feldern einer Nachricht des JMS-Protokolls Unterstreichungszeichen an Stelle von Silbentrennungsstrichen verwendet. In einer JMS-Nachricht gibt es zum Beispiel ein Feld x\_aux\_sender\_id anstatt eines Felds x-aux-sender-id.
2. Wenn als Gateway HTTP und als Paket 'None' angegeben ist, wird der ursprüngliche Dateiname in den HTTP-Headern als Content-Disposition: attachment;po.xml angegeben.  
Wenn als Empfänger JMS und als Pakettyp 'Backend Integration' angegeben ist, wird der ursprüngliche Dateiname als x-out-filename zu den anderen x-aux-\*-Headern hinzugefügt.
3. Wenn als Empfänger HTTP und als Quellenpaket 'None' angegeben ist, wird der ursprüngliche Dateiname in den HTTP-Headern als Content-Disposition: attachment;po.xml angegeben.

Wenn als Empfänger JMS und als Quellenpaket 'Backend Integration' angegeben ist, wird der ursprüngliche Dateiname auf x\_aux\_in\_file\_name: po.xml gesetzt. Bei einem HTTP-Empfänger wird der Name auf Content-Disposition: attachment;po.xml gesetzt.

Tabelle 8 enthält eine Übersicht zu den Informationen für die Header der Transportebene. Die folgenden Abschnitte enthalten Informationen zu Headern der Transportebene, die speziell für bestimmte Geschäftsprotokolle gelten:

- „Header der Transportebene und eine RosettaNet-Nachricht“
- „Header der Transportebene und eine AS2-Nachricht“ auf Seite 30
- „Header der Transportebene und eine AS1-Nachricht“ auf Seite 31

**Header der Transportebene und eine RosettaNet-Nachricht:** In Tabelle 9 wird beschrieben, wo WebSphere Partner Gateway Werte für die Felder des Headers der Transportebene aus einer RosettaNet-Nachricht abrufen.

Tabelle 9. Felder des Headers der Transportebene und RosettaNet-Inhalt

Headerfeld	Quelle des Werts: RosettaNet 2.0	Quelle des Werts: RosettaNet 1.1
x-aux-sender-id	<(DeliveryHeader)> <messageSenderIdentification> <PartnerIdentification> <GlobalBusinessIdentifier>	<ServiceHeader> <ProcessControl> <TransactionControl> <ActionControl> or <SignalControl> <PartnerRouter> <fromPartner> <PartnerDescription> <BusinessDescription> <GlobalBusinessIdentifier>
x-aux-receiver-id	<(DeliveryHeader)> <messageReceiverIdentification> <PartnerIdentification> <GlobalBusinessIdentifier>	<ServiceHeader> <ProcessControl> <TransactionControl> <ActionControl> or <SignalControl> <PartnerRouter> <toPartner> <PartnerDescription> <BusinessDescription> <GlobalBusinessIdentifier>
x-aux-protocol	Festgelegter Wert für RosettaNet: RNSC	Wie für RosettaNet 2.0
x-aux-protocol-version	Festgelegter Wert: 1.0	Wie für RosettaNet 2.0
x-aux-process-type	Der Quellen-XPath ist: /ServiceHeader/ProcessControl/ pipCode/GlobalProcessIndicatorCode	Der Quellen-XPath ist: /ServiceHeader/ProcessControl/ ProcessIdentity/GlobalProcessIndicatorCode
x-aux-process-version	Der Quellen-XPath ist: /ServiceHeader/ProcessControl/ pipVersion/VersionIdentifier  Der Wert der Versionskennung für jeden PIP befindet sich in der zugehörigen PIP-Spezifikation.	Der Quellen-XPath ist: /ServiceHeader/ProcessControl/ ProcessIdentity/VersionIdentifier  Der Wert der Versionskennung für jeden PIP befindet sich in der zugehörigen PIP-Spezifikation.
x-aux-payload-root-tag	Name des PIP wie 'Pip3A4PurchaseOrderRequest'	Wie für RosettaNet 2.0

Tabelle 9. Felder des Headers der Transportebene und RosettaNet-Inhalt (Forts.)

Headerfeld	Quelle des Werts: RosettaNet 2.0	Quelle des Werts: RosettaNet 1.1
x-aux-process-instance-id	Für Prozesse, die von einer Anwendung eingeleitet werden, ist der Wert die ID der Prozessinstanz. Für Prozesse, die von einem externen Partner eingeleitet werden und die kein Pass-Through-Arbeitsablauf sind, ist der Wert die Prozess-ID in der einleitenden RosettaNet-Anforderung: <pre>&lt;ServiceHeader&gt;   &lt;ProcessControl&gt;     &lt;pipInstanceId&gt;       &lt;InstanceIdentifier&gt;</pre>	<pre>&lt;ServiceHeader&gt;   &lt;ProcessControl&gt;     &lt;ProcessIdentity&gt;       &lt;InstanceIdentifier&gt;</pre>
x-aux-msg-id	<pre>&lt;(RNPipServiceContent)&gt; &lt;thisDocumentIdentifier&gt;   &lt;ProprietaryDocumentIdentifier&gt;</pre>	Wie für RosettaNet 2.0
x-aux-production	<pre>&lt;ServiceHeader&gt;   &lt;ProcessIndicator&gt;   &lt;GlobalUsageCode&gt;</pre>	<pre>&lt;Preamble&gt;   &lt;GlobalUsageCode&gt;</pre>

**Header der Transportebene und eine AS2-Nachricht:** In Tabelle 10 wird beschrieben, wo WebSphere Partner Gateway Werte für die Felder des Headers der Transportebene aus einer AS2-Nachricht abrufen.

**Anmerkung:** Die Werte sind von der Groß-/Kleinschreibung abhängig.

Tabelle 10. Felder des Headers der Transportebene aus AS2-Inhalt

Headerfeld	Quelle des Werts, wenn ein externer Partner eine AS2-Nachricht an den Hub sendet	Quelle des Werts, wenn eine AS2-Nachricht an einen externen Partner gesendet wird
x-aux-sender-id	Das AS2-Headerfeld für den Absender (From) wird in das Feld 'x-aux-sender-id' der Back-End-Integrationsnachricht eingetragen, die an den internen Partner gesendet wird.	Das Feld 'x-aux-sender-id' der eingehenden Back-End-Integrationsnachricht wird als AS2-Headerwert für den Absender der AS2-Nachricht verwendet.
x-aux-receiver-id	Das AS2-Headerfeld für den Empfänger (To) wird in das Feld 'x-aux-receiver-id' der Back-End-Integrationsnachricht eingetragen, die an den internen Partner gesendet wird.	Das Feld 'x-aux-receiver-id' der eingehenden Back-End-Integrationsnachricht wird als AS2-Headerwert für den Empfänger der AS2-Nachricht verwendet.
x-aux-protocol	Das Empfängerprotokoll (ToProtocol) der Partnerverbindung wird in das Feld 'x-aux-protocol' der Back-End-Integrationsnachricht eingetragen, die an den internen Partner gesendet wird.	Das Feld 'x-aux-protocol' der eingehenden Back-End-Integrationsnachricht wird zur Ermittlung des Absenderprotokolls (FromProtocol) der Partnerverbindung verwendet.
x-aux-protocol-version	Die Version des Empfängerprotokolls (ToProtocolVersion) der Partnerverbindung wird in das Feld 'x-aux-protocol-version' der Back-End-Integrationsnachricht eingetragen, die an den internen Partner gesendet wird.	Das Feld 'x-aux-protocol-version' der eingehenden Back-End-Integrationsnachricht wird als Wert für die Version des Empfängerprotokolls der Partnerverbindung verwendet.

Tabelle 10. Felder des Headers der Transportebene aus AS2-Inhalt (Forts.)

Headerfeld	Quelle des Werts, wenn ein externer Partner eine AS2-Nachricht an den Hub sendet	Quelle des Werts, wenn eine AS2-Nachricht an einen externen Partner gesendet wird
x-aux-process-type	Der Prozesscode des Empfängers (ToProcessCode) der Partnerverbindung wird in das Feld 'x-aux-process-type' der Back-End-Integrationsnachricht eingetragen, die an den internen Partner gesendet wird.	Das Feld 'x-aux-process-type' der eingehenden Back-End-Integrationsnachricht wird als Wert für den Prozesscode des Absenders (FromProcessCode) der Partnerverbindung verwendet.
x-aux-process-version	Die Version des Empfängerprozesses (ToProcessVersion) der Partnerverbindung wird in das Feld 'x-aux-process-version' der Back-End-Integrationsnachricht eingetragen, die an den internen Partner gesendet wird.	Das Feld 'x-aux-process-version' der eingehenden Back-End-Integrationsnachricht wird als Wert für die Version des Absenderprozesses (FromProcessVersion) der Partnerverbindung verwendet.
x-aux-payload-root-tag	Nur beim angepassten XML-Protokoll wird der im XPATH angegebene Root-Tag aus der Nachricht herausgefiltert und im Feld 'x-aux-payload-root-tag' verwendet.	Bei eingehenden Back-End-Integrationsnachrichten muss für dieses Feld kein Wert angegeben werden.
x-aux-process-instance-id	Dieses wird für AS2 nicht verwendet.	Dieses wird für AS2 nicht verwendet.
x-aux-msg-id	Nur beim angepassten XML-Protokoll wird die in XPATH angegebene Doc-ID aus der Nachricht herausgefiltert und im Feld 'x-aux-msg-id' verwendet.	Bei eingehenden Back-End-Integrationsnachrichten muss für dieses Feld kein Wert angegeben werden.
x-aux-system-msg-id	Dieses Feld wird auf die intern generierte eindeutige Kennung (ID) für diese Nachricht gesetzt.	Bei eingehenden Back-End-Integrationsnachrichten muss für dieses Feld kein Wert angegeben werden.
x-aux-production	Dieses wird für AS2 nicht verwendet.	Dieses wird für AS2 nicht verwendet.

**Header der Transportebene und eine AS1-Nachricht:** In Tabelle 11 wird beschrieben, wo WebSphere Partner Gateway Werte für die Felder des Headers der Transportebene aus einer AS1-Nachricht abrufen.

**Anmerkung:** Die Werte sind von der Groß-/Kleinschreibung abhängig.

Tabelle 11. Felder des Headers der Transportebene aus AS1-Inhalt

Headerfeld	Quelle des Werts, wenn ein externer Partner eine AS1-Nachricht an den Hub sendet	Quelle des Werts, wenn eine AS1-Nachricht an einen externen Partner gesendet wird
x-aux-sender-id	Die <i>AbsenderID</i> im Headerfeld "Subject: <i>EmpfängerID</i> ; <i>AbsenderID</i> " der AS1-Nachricht wird in das Feld 'x-aux-sender-id' der Back-End-Integrationsnachricht eingetragen, die an den internen Partner gesendet wird.	Das Feld 'x-aux-sender-id' der eingehenden Back-End-Integrationsnachricht wird als <i>AbsenderID</i> im Headerwert "Subject: <i>EmpfängerID</i> ; <i>AbsenderID</i> " der AS1-Nachricht verwendet.
x-aux-receiver-id	Die <i>EmpfängerID</i> im Headerfeld "Subject: <i>EmpfängerID</i> ; <i>AbsenderID</i> " der AS1-Nachricht wird in das Feld 'x-aux-receiver-id' der Back-End-Integrationsnachricht eingetragen, die an den internen Partner gesendet wird.	Das Feld 'x-aux-receiver-id' der eingehenden Back-End-Integrationsnachricht wird als <i>EmpfängerID</i> im Headerwert "Subject: <i>EmpfängerID</i> ; <i>AbsenderID</i> " der AS1-Nachricht verwendet.

Tabelle 11. Felder des Headers der Transportebene aus AS1-Inhalt (Forts.)

Headerfeld	Quelle des Werts, wenn ein externer Partner eine AS1-Nachricht an den Hub sendet	Quelle des Werts, wenn eine AS1-Nachricht an einen externen Partner gesendet wird
x-aux-protocol	Das Empfängerprotokoll (ToProtocol) der Partnerverbindung wird in das Feld 'x-aux-protocol' der Back-End-Integrationsnachricht eingetragen, die an den internen Partner gesendet wird.	Das Feld 'x-aux-protocol' der eingehenden Back-End-Integrationsnachricht wird als Wert des Absenderprotokolls (FromProtocol) der Partnerverbindung verwendet.
x-aux-protocol-version	Die Version des Empfängerprotokolls (ToProtocolVersion) der Partnerverbindung wird in das Feld 'x-aux-protocol-version' der Back-End-Integrationsnachricht eingetragen, die an den internen Partner gesendet wird.	Das Feld 'x-aux-protocol-version' der eingehenden Back-End-Integrationsnachricht wird als Wert für die Version des Empfängerprotokolls der Partnerverbindung verwendet.
x-aux-process-type	Die Prozesscode des Empfängers (ToProcessCode) der Partnerverbindung wird in das Feld 'x-aux-process-type' der Back-End-Integrationsnachricht eingetragen, die an den internen Partner gesendet wird.	Das Feld 'x-aux-process-type' der eingehenden Back-End-Integrationsnachricht wird als Wert für den Prozesscode des Absenders (FromProcessCode) der Partnerverbindung verwendet.
x-aux-process-version	Die Version des Empfängerprozesses (ToProcessVersion) der Partnerverbindung wird in das Feld 'x-aux-process-version' der Back-End-Integrationsnachricht eingetragen, die an den internen Partner gesendet wird.	Das Feld 'x-aux-process-version' der eingehenden Back-End-Integrationsnachricht wird als Wert für die Version des Absenderprozesses (FromProcessVersion) der Partnerverbindung verwendet.
x-aux-payload-root-tag	Nur beim angepassten XML-Protokoll wird der im XPATH angegebene Root-Tag aus der Nachricht herausgefiltert und im Feld 'x-aux-payload-root-tag' festgelegt.	Bei eingehenden Back-End-Integrationsnachrichten muss für dieses Feld kein Wert angegeben werden.
x-aux-process-instance-id	Dieses wird für AS1 nicht verwendet.	Dieses wird für AS1 nicht verwendet.
x-aux-msg-id	Nur beim angepassten XML-Protokoll wird die in XPATH angegebene Doc-ID aus der Nachricht herausgefiltert und im Feld 'x-aux-msg-id' verwendet.	Bei eingehenden Back-End-Integrationsnachrichten muss für dieses Feld kein Wert angegeben werden.
x-aux-system-msg-id	Dieses Feld wird auf die intern generierte eindeutige Kennung (ID) für diese Nachricht gesetzt.	Bei eingehenden Back-End-Integrationsnachrichten muss für dieses Feld kein Wert angegeben werden.
x-aux-production	Dieses wird für AS1 nicht verwendet.	Dieses wird für AS1 nicht verwendet.

**Header der Transportebene und eine ebMS-Nachricht:** In Tabelle 11 auf Seite 31 wird beschrieben, wo WebSphere Partner Gateway Werte für die Felder des Headers der Transportebene aus einer ebMS-Nachricht abrufen.

**Anmerkung:** Die Werte sind von der Groß-/Kleinschreibung abhängig.

Tabelle 12. Felder des Headers der Transportebene aus ebMS-Inhalt

Headerfeld	Quelle des Werts, wenn ein externer Partner ein ebMS-Dokument an den Hub sendet	Quelle des Werts, wenn ein ebMS-Dokument an einen externen Partner gesendet wird
x-aux-receiver-id	(SOAP-Header) <eb:MessageHeader><eb:To><eb:PartyId>	Der Wert dieses Attributs wird vom Back-End-System als Back-End-Header bereitgestellt.

Tabelle 12. Felder des Headers der Transportebene aus ebMS-Inhalt (Forts.)

Headerfeld	Quelle des Werts, wenn ein externer Partner ein ebMS-Dokument an den Hub sendet	Quelle des Werts, wenn ein ebMS-Dokument an einen externen Partner gesendet wird
x-aux-sender-id	(SOAP-Header) <eb:MessageHeader><eb:From><eb:PartyId>	Der Wert dieses Attributs wird vom Back-End-System als Back-End-Header bereitgestellt.
x-aux-SyncResponse	Attribut für Quellendokumentdefinition bcg.ro.ebxml.SYNCREPLYMODE  Gültige Werte: <ul style="list-style-type: none"> <li>• responseOnly</li> <li>• signalAndResponse</li> <li>• signalOnly</li> </ul>	Attribut für Zieldokumentdefinition
x-aux-process-instance-id	(SOAP-Header) <eb:MessageHeader><eb:ConversationId>	Wird vom Back-End-System als Back-End-Header bereitgestellt. Wenn er vom Back-End-System nicht gesetzt wird, generiert WebSphere Partner Gateway diesen Wert.  Das erforderliche ConversationId-Element in einem ebMS-Fluss ist eine Zeichenfolge, die eine Gruppe von zusammengehörigen Nachrichten angibt, aus denen sich der Dialog zwischen zwei Parteien zusammensetzt. Die Nachrichten in einem Dialog haben dieselbe Dialog-ID, aber unterschiedliche Nachrichten-IDs.
x-aux-process-type	(SOAP-Header) <eb:MessageHeader><eb:Service>	Wird vom Back-End-System als Back-End-Header bereitgestellt.
x-aux-process-version	(SOAP-Header) <eb:MessageHeader><eb:Service> Der Text des Elementwerts nach dem Zeichen '\$'. Wenn der Elementwert das Zeichen '\$' nicht enthält oder keine URI ist, wird der Standardwert 'ALL' verwendet.	Wird vom Back-End-System als Back-End-Header bereitgestellt.
x-aux-protocol	EBMS	Zielprotokoll in der Anzeige der Partnerverbindung
x-aux-protocol-version	(SOAP-Header) <eb:MessageHeader><eb:version>	Zielprotokollversion in der Anzeige der Partnerverbindung
x-aux-third-party-bus-id	<b>Konsole &gt; Systemverwaltung&gt; DocMgr-Eigenschaften</b>	N/V
x-aux-TimeToAccept	Attribut für Quellendokumentdefinition	
x-aux-system-msg-id	Dieses Feld wird auf die intern generierte eindeutige Kennung (ID) für diese Nachricht gesetzt.	Bei eingehenden Back-End-Integrationsnachrichten muss für dieses Feld kein Wert angegeben werden.
x-aux-msg-id	<soapenv:Header><eb:MessageHeader> <eb:MessageData><eb:MessageId>	Kann vom Back-End-System als Back-End-Header gesetzt werden. Sollte das nicht der Fall ein, wird dieser Wert von WebSphere Partner Gateway intern generiert.

Tabelle 12. Felder des Headers der Transportebene aus ebMS-Inhalt (Forts.)

Headerfeld	Quelle des Werts, wenn ein externer Partner ein ebMS-Dokument an den Hub sendet	Quelle des Werts, wenn ein ebMS-Dokument an einen externen Partner gesendet wird
x-aux-IntelligibleCheckRequired	Attribut für Quelldokumentdefinition bcg.ro.ebxml.ISINTELLIGIBLECHECKREQUIRED	N/V

Beispiel für ebMS-Transportheader:

```

Connection : Keep-Alive
x-aux-process-version : ALL
x-aux-receiver-id : 124456789
x-aux-SyncResponse : signalonly
Content-Length : 1326186
x-aux-process-instance-id : 918423619611581617869031
x-aux-process-type : ALMService
x-aux-protocol : ebMS
x-aux-sender-id : 987654421
x-aux-third-party-bus-id :
Content-Type: application/xml
x-aux-create-datetime : 2006-09-13T16:15:01Z
x-aux-TimeToAccept : 1
x-aux-transport-retry-count : 3
Host : 9.184.251.32:58080
x-aux-system-msg-id : 1158164092823000D606BBA390013320000000000000021
ReferenceId : 1158164101686000D606BBA390013320000000000000003
x-aux-msg-id : 1158163556558000C7627E08C025182D1C3AD7C5B55F7A3
                @wks184446wss.in.ibm.com
Keep-Alive : timeout=5, max=10000
x-aux-IntelligibleCheckRequired : no
x-aux-protocol-version : 2.0
    
```

## Nutzdaten

Die Nutzdaten der Nachricht enthalten den eigentlichen Inhalt der Nachricht. Die Position der Nutzdaten hängt von dem Transportprotokoll ab, mit dem die Nachricht gesendet wird, wie Sie in Tabelle 13 sehen.

Tabelle 13. Position der Nutzdaten

Transportprotokoll	Position der Nutzdaten
HTTP-Protokollnachrichten	Hauptteil der HTTP-Sendung
JMS-Protokollnachrichten	Hauptteil der JMS-Nachricht
RosettaNet-Nachrichten	Service-Content aus dem PIP
EDI	EDI-Umschlag
ROD/flat- oder XML-Dokument	ROD/flat- oder XML-Dokument

Wenn eine der beiden folgenden Bedingungen gilt, können die Nutzdaten im Base64-Format codiert und in einem XML-Transportumschlag enthalten sein:

- Wenn das Dokument einen Anhang enthält  
Ein Dokument mit Anhängen *muss* in einen XML-Transportumschlag eingefügt werden. Weitere Informationen zur Formatierung von Anhängen finden Sie unter „Anhänge“ auf Seite 35.
- Wenn Sie für die Umschlagsmarkierung für das Back-End-Integrationspaket 'Ja' angeben  
Wenn ein Dokument *unabhängig* davon, ob es Anhänge enthält oder nicht, in einen XML-Transportumschlag eingefügt werden soll, müssen Sie die Markierung für den Back-End-Integrationsumschlag in der Anzeige für die B2B-Funkti-

onalität des Profils auf 'Ja' setzen. Wenn Sie diese Markierung z. B. im Profil des internen Partners definieren möchten, müssen Sie die folgenden Arbeitsschritte ausführen:

1. Klicken Sie auf **Kontenadmin > Profile**.
2. Geben Sie den Namen des internen Partners ein (oder führen Sie eine Suchoperation für alle externen Partner aus).
3. Klicken Sie auf das Symbol **Details anzeigen** neben dem Namen des internen Partners.
4. Klicken Sie auf **B2B-Funktionalitäten**.
5. Klicken Sie auf das Symbol **Bearbeiten** neben **Back-End-Integration**.
6. Legen Sie für **Umschlagsmarkierung** die Einstellung **Ja** fest.

Dieser XML-Transportumschlag umgibt das Dokument mit dem Root-Tag <transport-envelope>. Innerhalb dieses Root-Tags befindet sich ein <payload>-Tag, der die Nutzdaten des Dokuments enthält. Wenn Anhänge vorhanden sind, ist jeder Anhang in einem <attachment>-Tag enthalten. Weitere Informationen zur Struktur dieser Tags finden Sie unter „Anhänge“.

WebSphere Partner Gateway enthält die folgende W3C-XML-Schemadatei, in der die Struktur des XML-Transportumschlags der Back-End-Integration beschrieben ist:

wbipackaging\_v1.1\_ns.xsd

**Anmerkung:** Für den ebMS-Fluss sollte das Paket wbipackaging\_v1.2\_ns.xsd verwendet werden.

Diese Schemadatei befindet sich im folgenden Verzeichnis auf dem Installationsdatenträger:

B2BIntegrate\packagingSchemas

Sie können ein beliebiges XML-Bearbeitungstool zur Überprüfung Ihres Back-End-Integrations-XMLs an dieser Schemadatei verwenden, um sicherzustellen, dass das Dokument gültig ist, bevor es an den Document Manager gesendet wird.

## Anhänge

Sofern das Geschäftsnachrichtenprotokoll dies zulässt, kann jedes Dokument einen oder mehrere Anhänge haben. Wenn das Dokument Anhänge enthält, *muss* es in einen XML-Transportumschlag eingefügt werden. Weitere Informationen hierzu finden Sie im Abschnitt „Nutzdaten“ auf Seite 34. In Tabelle 14 werden die XML-Attribute in den Tags payload und attachment beschrieben.

Tabelle 14. XML-Attribute der Tags payload und attachment

XML-Attribut	Beschreibung	Erforderlich
Content-Type	Gibt den MIME-Typ/-Subtyp an, z. B. 'text/xml' oder 'image/gif'.	Ja
Encoding	Gibt die Codierung an. Da der Anhang und die Nutzdaten in der Base64-Codierung vorliegen müssen, ist 'Base64' der einzige gültige Wert für dieses Attribut.	Nein

Abb. 13 zeigt ein Beispiel für ein Dokument in einem XML-Transportumschlag, das Nutzdaten und einen Anhang enthält.

**Anmerkung:** Der Namespace in diesem Beispiel ist erforderlich:

```
xmlns="http://www.ibm.com/websphere/bcg/2003/v1.0/wbipackaging"
```

```
<?xml version="1.0" encoding="utf-8"?>
<transport-envelope
xmlns="http://www.ibm.com/websphere/bcg/2003/v1.0/wbipackaging">
  <payload encoding="base64" contentType="application/xml">
    ...Im Base64-Format codierte XML-Nachricht...
  </payload>
  <attachment encoding="base64" Content-Type="text/xml">
    ...Im Base64-Format codierter XML-Anhang...
  </attachment>
</transport-envelope>
```

Abbildung 13. Beispiel eines XML-Transportumschlags für Nutzdaten und einen Anhang

**Anmerkungen:**

1. Zur Verarbeitung von Dokumenten, die mit WebSphere Interchange Server in den XML-Transportumschlag eingefügt wurden, stellt WebSphere Partner Gateway den Attachment-Data-Handler zur Verfügung. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumente mit Anhängen verarbeiten“ auf Seite 133.
2. Zur Verarbeitung von Dokumenten mit Anhängen in WebSphere Process Server stellt WebSphere Partner Gateway die Datenbindung des Back-End-Integrationspakets bereit. Informationen hierzu finden Sie im Abschnitt „Paketnachricht der Back-End-Integration verarbeiten“ auf Seite 66.

## Welcher Pakettyp eignet sich für Ihre Dokumente?

Dokumente in bestimmten Geschäftsprotokollen können nur bestimmte Typen von Paketen verwenden. Zum Beispiel kann ein RosettaNet- oder ein ebMS-Dokument *nur* verarbeitet werden, wenn der Pakettyp 'Backend Integration' angegeben wurde. Eine vollständige Liste darüber, welche Dokumenttypen welchen Pakettypen zugeordnet werden können, finden Sie in Tabelle 16 auf Seite 38, Tabelle 17 auf Seite 38 und Tabelle 18 auf Seite 39.

## Beispiel für ein Back-End-Integrationspaket über HTTP

Abb. 14 zeigt ein Beispiel einer Nachricht, die über das HTTP-Transportprotokoll von WebSphere Partner Gateway an eine Anwendung gesendet wird. Beachten Sie, dass die Nachricht keinen Anhang enthält.

```

POST /sample/receive HTTP/1.1
Host: sample.COM
Content-Type: application/xml
Content-Length: nnn
x-aux-sender-id: 000000001
x-aux-receiver-id: 000000002
x-aux-third-party-bus-id: 000000003
x-aux-create-datetime: 2002-10-28T23:05:02Z
x-aux-protocol: RNSC
x-aux-protocol-version: 1.0
x-aux-process-type: 3A4
x-aux-process-version: V02.00
x-aux-payload-root-tag: Pip3A4PurchaseOrderRequest
x-aux-msg-id: 1021358129419
x-aux-system-msg-id: 2
x-aux-production: Production
x-aux-process-instance-id: 123456
x-aux-transport-retry-count: 0
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Pip3A4PurchaseOrderRequest SYSTEM
  "3A4PurchaseOrderRequestMessageGuideline_v1_2.dtd">
<Pip3A4PurchaseOrderRequest>
  <PurchaseOrder>
    ...
  </PurchaseOrder>
  ...
  <thisDocumentIdentifier>
    <ProprietaryDocumentIdentifier>1021358129419
    </ProprietaryDocumentIdentifier>
  </thisDocumentIdentifier>
  <GlobalDocumentFunctionCode>Request</GlobalDocumentFunctionCode>
</Pip3A4PurchaseOrderRequest>

```

Abbildung 14. Beispielnachricht mit dem HTTP-Transportprotokoll

---

## Zu verwendendes Nachrichtentransportprotokoll

Wenn das Back-End-System und WebSphere Partner Gateway einander Nachrichten senden, müssen beide Seiten dasselbe Nachrichtentransportprotokoll verwenden. Das *Nachrichtentransportprotokoll* definiert das Kommunikationsprotokoll, in dem die Nachrichten gesendet werden.

WebSphere Partner Gateway kommuniziert mit einem Back-End-System über die zugehörige Back-End-Integrationschnittstelle. In Tabelle 15 sind die Transportprotokolle aufgeführt, die von dieser Back-End-Integrationschnittstelle unterstützt werden.

Tabelle 15. Von der Back-End-Integration unterstützte Transportprotokolle

Transportprotokoll	Weitere Informationen
HTTP oder HTTPS	„HTTP-Transportprotokoll“ auf Seite 40
Dateisystemdateien	„Dateisystemprotokoll“ auf Seite 44
JMS	„JMS-Protokoll“ auf Seite 41

In Tabelle 16 auf Seite 38 sind die Transportprotokolle aufgeführt, die für die verschiedenen Pakettypen und Geschäftsprotokolle unterstützt werden, wenn der Hub Dokumente an das Back-End-System sendet.

*Tabelle 16. Unterstützte Transportprotokolle für Nachrichten von WebSphere Partner Gateway an das Back-End-System*

<b>Pakettyp</b>	<b>Geschäftsprotokoll</b>	<b>HTTP oder HTTPS?</b>	<b>JMS?</b>	<b>Dateisystem?</b>
Backend Integration	RosettaNet (RNSC)	Ja	Ja	Nein
	ebMS	Ja	Ja	Nein
	Binär	Ja	Ja	Nein
None	EDI (Informationen zu EDI finden Sie in Tabelle 18 auf Seite 39.)			
	XML	Ja	Ja	Nein
	ROD/flat	Ja	Ja	Nein
	EDI (Informationen zu EDI finden Sie in Tabelle 18 auf Seite 39.)			
	Nur cXML	Ja	Nein	Nein
	Nur SOAP	Ja	Nein	Nein
None	Binär	Ja	Ja	Ja
	XML	Ja	Ja	Ja
	ROD/flat	Ja	Ja	Ja
	ROD/flat	Ja	Ja	Ja

In Tabelle 17 sind die Transportprotokolle aufgeführt, die für die verschiedenen Pakettypen und Geschäftsprotokolle unterstützt werden, wenn das Back-End-System Dokumente an den Hub sendet.

*Tabelle 17. Unterstützte Transportprotokolle für Nachrichten vom Back-End-System an WebSphere Partner Gateway*

<b>Pakettyp</b>	<b>Geschäftsprotokoll</b>	<b>HTTP oder HTTPS?</b>	<b>JMS?</b>	<b>Datei-system?</b>
Backend Integration	RosettaNet (RNSC)	Ja	Ja	Nein
	ebMS	Ja	Ja	Nein
	XML	Ja	Ja	Nein
	Binär	Ja	Ja	Nein
	ROD/flat	Ja	Ja	Nein
None	Nur XML	Ja	Ja	Ja
	EDI (Informationen zu EDI finden Sie in Tabelle 18 auf Seite 39.)			
	Nur cXML	Ja	Nein	Nein
	Nur SOAP	Ja	Nein	Nein
	Nur binär	Nein	Nein	Nein
	Nur ROD/flat	Ja	Ja	Ja

In Tabelle 18 auf Seite 39 sind die Transportprotokolle und Pakettypen aufgeführt, die für die verschiedenen EDI-, XML- und ROD/flat-Dokumente unterstützt werden.

Tabelle 18. Unterstützte Transportprotokolle für den Austausch von EDI-Dokumenten zwischen WebSphere Partner Gateway und dem Back-End-System

Pakettyp	Dokument	HTTP oder HTTPS?	JMS?	Datei-system?
Back-End-Integration	Einzelner Austausch mit einzelner Transaktion (z. B. X12-850-Transaktion in Umschlag)	Ja	Ja	Nein
	Einzelner Austausch mit mehreren Transaktionen (z. B. X12-850-Transaktion und X12-890-Transaktion in einem gemeinsamen Umschlag)	Ja	Ja	Nein
	Mehrere Austauschelemente mit einer einzigen Transaktion (z. B. zwei X12-Umschläge in derselben Datei, wobei jeder eine einzelne Transaktion enthält)	Ja	Ja	Nein
	Mehrere Austauschelemente mit mehreren Transaktionen (z. B. zwei X12-Umschläge in derselben Datei, wobei jeder mehrere Transaktionen enthält)	Ja	Ja	Nein
	EDI-Transaktion (z. B. eine X12-850-Transaktion), die alleine nicht zugestellt werden kann, weil eine Transaktion sich in einem EDI-Austausch befinden muss	Nein	Nein	Nein
	Dokument (zum Beispiel im XML-Format), das später in eine EDI-Transaktion transformiert wird	Ja	Ja	Nein
None	Einzelner Austausch mit einer einzigen Transaktion	Ja	Ja	Ja
	Einzelner Austausch mit mehreren Transaktionen	Ja	Ja	Ja
	Mehrere Austauschelemente mit einer einzigen Transaktion	Ja	Ja	Ja
	Mehrere Austauschelemente mit mehreren Transaktionen	Ja	Ja	Ja
	EDI-Transaktion (nicht unterstützt; muss einen Austauschumschlag haben)	Nein	Nein	Nein
	Dokument (zum Beispiel im XML-Format), das später in eine EDI-Transaktion transformiert wird	Ja	Ja	Ja

Die vorangegangenen Tabellen enthalten eine Aufstellung der Transportprotokolle, die zwischen dem Hub und dem Back-End-System verwendet werden können. Der Hub kann zusätzliche Transportprotokolle verwenden, um Dokumente an Partner zu senden oder von diesen zu empfangen. Er kann z. B. ein Dokument an einen fernen FTP-Server senden und hierzu den FTP-Scripting-Transport verwenden.

Darüber hinaus kann der Hub den FTP-Scripting-Transport auch zum Empfang von Dokumenten einsetzen. Der FTP-Scripting-Transport wird im Handbuch *WebSphere Partner Gateway Hubkonfiguration* beschrieben und kann zum Senden und Empfangen von Dokumenten über das Internet verwendet werden. Beim Senden und Empfangen von Dokumenten an bzw. von VANs (Value Added Network) muss es sogar eingesetzt werden.

## HTTP-Transportprotokoll

Zum Senden von Nachrichten mit einem HTTP-Protokoll verwendet WebSphere Partner Gateway das Protokoll HTTP/S 1.1. Zum Empfangen von Nachrichten aus Back-End-Systemen unterstützt WebSphere Partner Gateway die beiden HTTP/S-Versionen 1.0 und 1.1.

Die HTTP-Nachricht kann die Attribute des Integrationspakets enthalten. Ob diese Attribute enthalten sind, hängt wie folgt vom Paketty ab, der der Partnerverbindung zugeordnet ist:

- Wenn die Partnerverbindung angibt, dass für die HTTP-Nachricht der Paketty 'Backend Integration' verwendet wird, sind im Header der Transportebene der HTTP-Nachricht zusätzliche Attribute mit Informationen zu dieser Nachricht enthalten, die zum Beispiel das Protokoll des Inhalts, die Kennung (ID) der Nachricht und den Absender der Nachricht angeben. Eine vollständige Liste der Felder im Header finden im Abschnitt „Inhalt des Headers der Transportebene“ auf Seite 26.

RosettaNet-Nachrichten müssen ein Back-End-Integrationspaket verwenden.

- Wenn für die Partnerverbindung der Paketty 'None' definiert ist, enthält die HTTP-Nachricht diese zusätzlichen Attribute *nicht*. In diesem Fall analysiert WebSphere Partner Gateway die Nachricht syntaktisch, um diese Informationen zu ermitteln.

SOAP und cXML-Nachrichten müssen 'None' verwenden.

**Anmerkung:** XML-Nachrichten können entweder den Paketty 'None' oder 'Backend Integration' verwenden. Ebenso können EDI-Nachrichten entweder 'None' oder 'Backend Integration' verwenden. Binäre Nachrichten, die vom Back-End-System empfangen werden, müssen 'Backend Integration' verwenden. Allerdings gilt dies umgekehrt nicht, da WebSphere Partner Gateway das Senden binärer Nachrichten an die Anwendung mit beiden Pakettyen unterstützt.

### Ablauf

Wenn HTTP- oder HTTPS-Nachrichten zwischen WebSphere Partner Gateway und einer Anwendung zum asynchronen Austausch hin- und hergesendet werden, erfolgt dies in den folgenden Schritten:

1. Das Quellensystem (WebSphere Partner Gateway oder das Back-End-System) übergibt eine HTTP-Nachricht an das Empfängersystem unter Angabe einer URL-Adresse.
2. Das Empfängersystem empfängt die Nachricht und sendet die Empfangsbestätigung der Protokollebene, HTTP 200 oder 202, um die Änderung des Eigentumsrechts zu signalisieren. Das Quellensystem ignoriert den Hauptteil dieser Empfangsbestätigungsnachricht. Wenn während dieser Verarbeitung ein Fehler auftritt, sendet das Empfängersystem eine Nachricht HTTP 500 zurück an das Quellensystem.
3. Wenn WebSphere Partner Gateway als Empfängersystem definiert ist (d. h. wenn WebSphere Partner Gateway eine Nachricht empfängt), bleibt die Nachricht erhalten und die Verbindung zum Quellensystem wird freigegeben.
4. Das Zielsystem kann die Nachricht dann asynchron verarbeiten.

Wenn der Austausch synchron erfolgt (z. B. bei einem SOAP- oder cXML-Dokument), wird zusammen mit der Nachricht HTTP 200 eine Antwort in derselben HTTP-Verbindung zurückgegeben.

### **Nachrichten über das HTTP-Protokoll vom Back-End-System senden**

Zum Senden einer Nachricht an WebSphere Partner Gateway über das HTTP-Protokoll führt ein Back-End-System die folgenden Schritte aus:

1. Es erstellt die Nachricht.  
Das Attribut 'Content-Type' im Header der Transportebene gibt die Codierung an, die für die Nachricht verwendet wird.
2. Es packt die Nachricht entsprechend dem Pakettyp, der für die Verbindung festgelegt ist.  
Für das Back-End-Integrationspaket fügt das Back-End-System die Attribute des Protokollheaders hinzu, die für WebSphere Partner Gateway erforderlich sind.
3. Es übergibt die Nachricht an die URL-Adresse, die von WebSphere Partner Gateway zum Empfang solcher Nachrichten verwendet wird.
4. Wenn der Austausch synchron stattfindet, wartet das Back-End-System auf den Empfang einer Antwort über dieselbe Verbindung, die für die Anforderung verwendet wurde.

Zur Aktivierung des HTTP-Nachrichtenaustauschs in dieser Richtung verwenden Sie die Seite **Empfängerdetails** der Community Console, um auf dem Hub einen Empfänger für eingehende Dokumente zu definieren. Dieser Empfänger legt eine URL-Adresse fest. Das Back-End-System muss diese Adresse kennen, um Dokumente an den Hub zu senden.

### **Nachrichten über das HTTP-Protokoll auf dem Back-End-System empfangen**

Zum Empfang einer Nachricht von WebSphere Partner Gateway über das HTTP-Protokoll führt ein Back-End-System die folgenden Schritte aus:

1. Es ist für eine Nachricht unter einer bestimmten URL-Adresse empfangsbereit.
2. Wenn eine Nachricht empfangen wird, verarbeitet es die Nachricht:
  - Wenn die Verbindung 'None' verwendet, muss das Back-End-System die Nachricht syntaktisch analysieren, um festzustellen, wie sie zu behandeln ist.
  - Wenn die Verbindung mit dem Back-End-Integrationspaket arbeitet, kann die Anwendung anhand der Attribute der Back-End-Integration feststellen, wie die Nachricht zu behandeln ist.
3. Wenn der Austausch synchron stattfindet, gibt das Back-End-System eine Antwort über dieselbe Verbindung zurück, die für die Anforderung verwendet wurde.

Zur Aktivierung des HTTP-Nachrichtenaustauschs in dieser Richtung müssen Sie die Seite **Ziel** der Community Console verwenden, um ein Ziel zu definieren, das die Adresse angibt, unter der Dokumente an das Back-End-System gesendet werden.

## **JMS-Protokoll**

Das JMS-Protokoll basiert auf dem Java<sup>(TM)</sup> Message Service (JMS) und übermittelt Nachrichten über transaktionsorientierte, persistente JMS-Warteschlangen, die zum Beispiel von IBM WebSphere MQ bereitgestellt werden. Das JMS-Protokoll unterstützt die folgenden JMS-Nachrichtentypen:

- StreamMessage (als Bytefeldgruppe)

- BytesMessage (als Bytefeldgruppe)
- TextMessage

Im JMS-Protokoll sendet ein System eine JMS-Nachricht an ein anderes System. Nachdem das zweite System die Nachricht empfangen hat, wird sie von diesem aus der Warteschlange entfernt. Von diesem Zeitpunkt an kann das empfangende System die Nachricht asynchron verarbeiten.

Die JMS-Nachricht kann Attribute des Integrationspakets enthalten. Ob diese Attribute enthalten sind, hängt wie folgt vom Pakettyp ab, der der Partnerverbindung zugeordnet ist:

- Wenn die Partnerverbindung angibt, dass die JMS-Nachricht das Back-End-Integrationspaket enthält, sind in der JMS-Nachricht Informationen der Transportebene (z. B. das Protokoll des Inhalts, die Kennung (ID) der Nachricht und der Absender der Nachricht) in Form von JMS-Eigenschaften (Properties) innerhalb der Nachricht enthalten. Eine vollständige Liste der Eigenschaften finden Sie im Abschnitt „Inhalt des Headers der Transportebene“ auf Seite 26.

**Anmerkung:** Aus Gründen der Kompatibilität mit WebSphere MQ JMS werden in den Eigenschaftsnamen der JMS-Nachrichten Unterstreichungszeichen an Stelle von Silbentrennungsstrichen verwendet. Zum Beispiel entspricht der Name der Eigenschaft 'x\_aux\_system\_msg\_id' in einer JMS-Nachricht dem HTTP-Headerfeld 'x-aux-system-msg-id'. Wenn WebSphere Partner Gateway eine JMS-Nachricht verarbeitet, werden die Unterstreichungszeichen dieser Eigenschaften in Silbentrennungsstriche konvertiert.

- Wenn die Partnerverbindung als Pakettyp 'None' angibt, enthält die JMS-Nachricht diese zusätzlichen Attribute *nicht*.

Mit Ausnahme von Binärnachrichten unterstützt WebSphere Partner Gateway das Senden und Empfangen von JMS-Nachrichten mit beiden Pakettypen. Binärnachrichten, die aus einer Anwendung empfangen werden, müssen das Back-End-Integrationspaket verwenden. Dies gilt umgekehrt jedoch nicht, da WebSphere Partner Gateway das Senden binärer Nachrichten an die Anwendung mit beiden Pakettypen unterstützt.

## JMS-Umgebung einrichten

Zum Einrichten der JMS-Umgebung werden die folgenden Provider benötigt:

- JMS-Provider

Ein JMS-Provider stellt die Implementierung der JMS-API-Unterstützung für die Nachrichtenübertragung bereit. Das Back-End-System, mit dem Sie Dokumente austauschen, legt den JMS-Provider fest.

- Wenn Sie Dokumente mit WebSphere Interchange Server austauschen, wird WebSphere MQ als JMS-Provider verwendet.
- Wenn Sie Dokumente mit WebSphere Process Server austauschen, werden WebSphere Platform Messaging und ein Service-Integration-Bus als JMS-Provider verwendet. Details zu den JMS-Providern, die mit WebSphere Process Server verwendet werden können, finden Sie in Kapitel 5, „Integration von WebSphere Process Server mit JMS als Transportprotokoll“, auf Seite 97.

Der JMS-Provider stellt normalerweise ein Programm bereit, mit dem Sie die JMS-Umgebung konfigurieren können. WebSphere MQ stellt beispielsweise das Programm JMSAdmin bereit, mit dem Sie die Objekte erstellen können, die von JMS benötigt werden. Hierzu gehören die JMS-Verbindungsfactory sowie die JMS-Warteschlangenobjekte. Werden diese Objekte erstellt, dann speichert das System die entsprechenden Verweise in der JNDI.

**Anmerkung:** Bei der Nachrichtenübertragung unterstützt WebSphere Partner Gateway ausschließlich das Punkt-zu-Punkt-Modell.

- JNDI-Provider

Der JNDI-Provider stellt die JNDI-Implementierung bereit, die zum Speichern von Verweisen auf JMS-Objekte verwendet wird.

Damit eine Back-End-Anwendung Geschäftsdokumente über das JMS-Protokoll an WebSphere Partner Gateway senden kann, muss ein JMS-Empfänger konfiguriert werden. Dieser JMS-Empfänger empfängt Nachrichten aus einer JMS-Warteschlange. Die Dokumente werden in den WebSphere Partner Gateway-Workflow eingegliedert. Die Konfiguration des JMS-Empfängers umfasst das Festlegen der erforderlichen Parameter für den Zugriff auf die JNDI sowie der Namen der JMS-Objekte. Zur Integration mit dem Back-End-System muss die für den JMS-Empfänger konfigurierte Warteschlange mit der Warteschlange übereinstimmen, über die das Back-End-System JMS-Nachrichten sendet.

In ähnlicher Weise verwendet WebSphere Partner Gateway ein JMS-Ziel, um Geschäftsdokumente an die Warteschlange zu senden, in der die Partner den Empfang erwarten. Aus diesem Grund müssen Sie beim Senden von Nachrichten an das Back-End-System sicherstellen, dass im Profil des internen Partners ein JMS-Ziel konfiguriert ist. Das Ziel sollte hierbei so konfiguriert sein, dass abgehende Nachrichten an die Warteschlange gesendet werden, in der das Back-End-System eingehende Daten empfängt. Die Konfiguration des JMS-Ziels umfasst das Festlegen der erforderlichen Parameter für den Zugriff auf die JNDI sowie der Namen der JMS-Objekte.

## Übersicht zur Einrichtung der JMS-Umgebung

Zur Kommunikation über das Transportprotokoll JMS benötigen WebSphere Partner Gateway und das Back-End-System eine JMS-Warteschlange für *jede* Richtung der Kommunikation. Daher müssen Sie die folgenden Schritte ausführen, um die erforderlichen JMS-Warteschlangen bereitzustellen:

- Konfigurieren Sie Ihre JMS-Umgebung.
- Erstellen Sie einen Warteschlangenmanager und die erforderlichen Warteschlangen, zu denen auch die Übertragungswarteschlange, die ferne Warteschlange und die Empfangswarteschlange gehören.

Der JMS-Warteschlangenmanager kann sich auf jedem beliebigen Computer einschließlich der folgenden Einheiten befinden:

- Computer, auf dem sich das Back-End-System befindet.
- Computer, auf dem sich WebSphere Partner Gateway befindet.

Darüber hinaus können Sie einen Warteschlangenmanager auf *beiden* Computern haben, d. h. auf dem, auf dem sich das Back-End-System befindet, und dem, auf dem sich WebSphere Partner Gateway befindet. In diesem Fall verwenden Sie Setupkanäle, um die beiden Warteschlangenmanager zu verbinden. Bei dieser Methode muss keine Seite Clientverbindungen über das Netz herzustellen.

Anweisungen zur Konfiguration eines JMS-Transportprotokollmechanismus mit Hilfe von WebSphere MQ Version 6.0 mit dem neuesten Fixpack finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*. Anweisungen zur Konfiguration der JMS-Umgebung beim Austausch von Dokumenten mit WebSphere Process Server finden Sie in Kapitel 5, „Integration von WebSphere Process Server mit JMS als Transportprotokoll“, auf Seite 97.

## Nachrichten über das JMS-Protokoll vom Back-End-System senden

Zum Senden einer Nachricht an WebSphere Partner Gateway über das JMS-Protokoll führt ein Back-End-System die folgenden Schritte aus:

1. Es erstellt die Nachricht.
2. Es packt die Nachricht entsprechend dem Pakettyp, der für die Verbindung festgelegt ist.  
Für das Back-End-Integrationspaket fügt die Anwendung die erforderlichen JMS-Headerattribute hinzu.
3. Es sendet die Nachricht an die JMS-Warteschlange, die vom Back-End-System zum Senden von Nachrichten an WebSphere Partner Gateway verwendet wird.

## Nachrichten über das JMS-Protokoll auf dem Back-End-System empfangen

Zum Empfang einer Nachricht von WebSphere Partner Gateway über das JMS-Protokoll führt ein Back-End-System die folgenden Schritte aus:

1. Es ist für eine Nachricht in der JMS-Warteschlange empfangsbereit.
2. Wenn eine Nachricht empfangen wird, verarbeitet es die Nachricht:
  - Wenn die Verbindung 'None' verwendet, muss das Back-End-System die Nachricht syntaktisch analysieren, um festzustellen, wie sie zu behandeln ist.
  - Wenn die Verbindung mit dem Back-End-Integrationspaket arbeitet, kann die Anwendung anhand der Attribute der Back-End-Integration feststellen, wie die Nachricht zu behandeln ist.

## Dateisystemprotokoll

Über das Dateisystemprotokoll kann WebSphere Partner Gateway Nachrichten senden, indem es sie in eine definierte Verzeichnisstruktur platziert. WebSphere Partner Gateway empfängt Nachrichten, indem es sie aus der Verzeichnisstruktur liest. Das Dateisystemprotokoll unterstützt nur den Pakettyp 'None'.

### Nachrichten über das Dateisystemprotokoll vom Back-End-System senden

Zum Senden einer Nachricht an WebSphere Partner Gateway mit dem Dateisystemprotokoll sollte eine Anwendung die folgenden Schritte ausführen:

1. Sie sollte die Nachrichtendatei in einem temporären Verzeichnis erstellen.
2. Wenn die Datei bereit ist, sollte sie die Datei in das Verzeichnis verschieben, das von WebSphere Partner Gateway regelmäßig abgefragt wird.

Zur Aktivierung des Nachrichtenaustauschs über ein Dateisystem in dieser Richtung verwenden Sie die Seite **Empfängerdetails** der Community Console, um einen Empfänger für eingehende Dokumente zu definieren. Der Empfänger der Nachricht bestimmt das Verzeichnis, das von WebSphere Partner Gateway abgefragt wird. Wenn Sie einen Empfänger erstellen, erstellt WebSphere Partner Gateway ein Verzeichnis 'Documents' und zugehörige Unterverzeichnisse für den Empfänger wie folgt:

```
<dokumentstammverzeichnis>  
  Documents  
    Production  
    Test  
<andere zieltypen>
```

WebSphere Partner Gateway fragt die Verzeichnisse 'Documents' und die zugehörigen Unterverzeichnisse regelmäßig ab, um Nachrichtendateien zu erkennen. Wenn

eine Nachricht gefunden wird, nimmt WebSphere Partner Gateway die Nachricht auf und löscht die Nachrichtendatei aus dem Verzeichnis. Anschließend verarbeitet WebSphere Partner Gateway die Nachricht in üblicher Weise. Informationen zur Erstellung eines Empfängers finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

### **Nachrichten über das Dateisystemprotokoll auf dem Back-End-System empfangen**

Zum Empfangen von Nachrichten über das Dateisystemprotokoll sollte eine Anwendung folgende Schritte ausführen:

1. Sie sollte das entsprechende Verzeichnis regelmäßig auf Nachrichtendateien überprüfen.
2. Wenn eine Nachricht vorhanden ist, muss die Anwendung diese aufnehmen.
3. Sie sollte die Nachricht aus dem Verzeichnis löschen.
4. Sie sollte die Nachricht verarbeiten.

Zur Aktivierung des Dateisystem-Nachrichtenaustauschs in dieser Richtung verwenden Sie die Seite **Ziel** der Community Console, um ein Ziel zu definieren, das festlegt, wohin die Dokumente gesendet werden. WebSphere Partner Gateway stellt die Nachrichtendatei in das Verzeichnis 'Documents', das vom Ziel definiert wird. Durch die Definition des Zielverzeichnisses über das Ziel kann jede Partnerverbindung ein anderes Verzeichnis haben. Informationen zu Zielen finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

---

## **Zugriff auf Back-End-Anwendungen**

WebSphere Partner Gateway bietet die Möglichkeit der Integration mit vielen verschiedenen Back-End-Anwendungen. In der Regel erfolgt der Zugriff auf eine Back-End-Anwendung über ein Back-End-System, wie zum Beispiel einen Integrationsbroker. Im vorliegenden Handbuch finden Sie Informationen zur Integration mit den Back-End-Systemen, die in Tabelle 19 aufgeführt sind.

*Tabelle 19. Unterstützte Back-End-Systeme für WebSphere Partner Gateway*

<b>Back-End-System</b>	<b>Weitere Informationen</b>
WebSphere Process Server	Kapitel 3, „Einführung zur WebSphere Process Server-Integration“, auf Seite 59
WebSphere Interchange Server	Kapitel 8, „Einführung zur InterChange Server-Integration“, auf Seite 121
WebSphere Message Broker	Kapitel 11, „Integration mit WebSphere Message Broker“, auf Seite 191
WebSphere Data Interchange	Kapitel 12, „Integration mit WebSphere Data Interchange“, auf Seite 211

---

## **Nachrichtenbehandlung**

In diesem Abschnitt wird beschrieben, wie WebSphere Partner Gateway die folgenden Situationen behandelt, die sich auf die Zustellung von Nachrichten auswirken:

- „Zustellung über Warteschlangen“ auf Seite 46
- „Kommunikationsfehlerbehandlung“ auf Seite 46
- „Doppelte Nachrichten“ auf Seite 47

## Zustellung über Warteschlangen

WebSphere Partner Gateway übergibt Informationen zu allen Dokumenten, die an ein bestimmtes Ziel gesendet werden sollen, an eine Warteschlange. Der Document Manager verarbeitet diese Nachrichten in der Reihenfolge, in der sie von der Warteschlange empfangen werden (FIFO), und verwendet zum Senden einen Thread für jede einzelne Nachricht. Beachten Sie, dass bei einem Ziel (z. B. URL-Adresse beim HTTP-Transportprotokoll oder JMS-Ziel beim JMS-Transportprotokoll), das für den Offlinestatus konfiguriert ist (siehe Kommunikationsfehlerbehandlung), die Nachrichten in der Warteschlange verbleiben, bis das Ziel aktiviert (online verfügbar) wird. Wenn der Document Manager einen Fehler in einem Thread empfängt, hindert er andere Threads an dem Versuch, ihre Nachrichten zuzustellen. Der Document Manager verlegt diese Nachrichten zurück in die Warteschlange, bis er die Nachricht, die den Fehler verursachte, zustellen kann.

Wenn die Anzahl der fehlgeschlagenen Versuche die maximale Anzahl von Versuchen überschreitet, setzt der Document Manager die Nachricht in ein Verzeichnis für fehlgeschlagene Nachrichten und versucht, die nächste Nachricht in der Warteschlange zuzustellen, sofern das Ziel nicht offline ist.

## Kommunikationsfehlerbehandlung

Wenn WebSphere Partner Gateway der Absender ist und die Anwendung einen Fehler zurückgibt (z. B. eine HTTP-Antwortnachricht, die keine Nachricht HTTP 200 oder 202 bei Verwendung des HTTP-Protokolls ist), kann WebSphere Partner Gateway je nach Konfiguration für das betroffene Ziel erneut versuchen, die Nachricht zu senden. Jedes Ziel (bzw. bei HTTP jede URL-Adresse) verfügt über die folgenden Optionen, die die Anzahl von Wiederholversuchen sowie die Art und Weise, in der die Nachrichten gesendet werden, bestimmen:

Tabelle 20. Konfigurationsoptionen für Ziele

Konfigurationsoptionen	Beschreibung
Wiederholungszähler	Gibt die Anzahl der auszuführenden Wiederholversuche für ein Dokument an, falls ein Fehler empfangen wird.
Wiederholungsintervall	Gibt das Zeitintervall zwischen Wiederholungsversuchen an.
Online/Offline	Startet und stoppt Zustellversuche.
Anzahl Threads	Gibt die Anzahl der übergebenden Threads an, die Nachrichten pro Ziel verarbeiten.

Wenn WebSphere Partner Gateway nicht zum Wiederholen der Nachrichtensendeveruche konfiguriert ist oder wenn alle Zustellversuche fehlschlagen, signalisiert WebSphere Partner Gateway das Problem durch eine oder alle der folgenden Aktionen:

- Darstellen der Fehler in verschiedenen Anzeigen der Community Console, wie der Dokumentanzeige und der RosettaNet-Anzeige.
- Senden einer E-Mail an relevante Personen, um sie von dem Problem in Kenntnis zu setzen, so dass sie entsprechende Maßnahmen ergreifen können, sofern ein E-Mail-Alert für das Ereignis einer fehlgeschlagenen Zustellung definiert wurde.
- Erstellen eines Ereignisdokuments und Senden dieses Dokuments an den Empfänger.

Weitere Informationen hierzu finden Sie im Abschnitt zur Verwaltung von Zielkonfigurationen im Handbuch *WebSphere Partner Gateway Verwaltung*.

## Doppelte Nachrichten

Alle Nachrichten, die von WebSphere Partner Gateway gesendet oder empfangen werden, müssen eine GUID aufweisen. WebSphere Partner Gateway verwendet diese GUID, um doppelte Nachrichten zu erkennen. Wenn das Back-End-Integrationspaket verwendet wird, enthält jede Nachricht ihre GUID im Header der Transportebene. Für das HTTP-Protokoll ist die GUID zum Beispiel im Feld `x-aux-system-msg-id` enthalten (siehe „Inhalt des Headers der Transportebene“ auf Seite 26). Die GUID wird vom Absender der Nachricht generiert. Das Dateisystemprotokoll unterstützt keine Prüfung auf doppelte Nachrichten.

Wenn der Versuch, eine Nachricht zu senden, zu einem Fehler führt, verwendet WebSphere Partner Gateway die GUID der Nachricht bei jedem Wiederholungsversuch wieder. Wenn WebSphere Partner Gateway eine Nachricht empfängt, die eine doppelte GUID enthält, gibt es eine positive Empfangsbestätigung zurück (z. B. HTTP 200), verarbeitet die doppelte Nachricht jedoch nicht.

**Anmerkung:** WebSphere Partner Gateway prüft auf doppelte Nachrichten auf der RosettaNet-Prozessebene, wenn RosettaNet verwendet wird. Darüber hinaus wird auf doppelte Nachrichten geprüft, wenn XML verwendet wird.

---

## WebSphere Partner Gateway konfigurieren

Im Abschnitt „Prozess der Hubkonfiguration“ auf Seite 4 wurde eine allgemeine Beschreibung der Arbeitsschritte aufgeführt, die vom Hubadministrator ausgeführt werden, um den Hub zu konfigurieren. In diesem Abschnitt werden die Schritte zur Konfiguration von WebSphere Partner Gateway zur Verwendung mit einem Back-End-System zusammengefasst. Diese Konfigurationsschritte setzen voraus, dass Sie die externen Partner in Ihrer Hub-Community bereits konfiguriert haben. Insbesondere geht dieser Abschnitt von der Annahme aus, dass die folgenden Konfigurationsschritte bereits ausgeführt wurden:

- Ein Partnerprofil für den internen Partner wurde bereits erstellt.
- Es wurden bereits externe Partner für die Dokumentenquelle (bzw. das Dokumentenziel) erstellt.
- In den B2B-Programmen der externen Partner wurden Partnerprofile für den internen Partner erstellt.
- Es wurde ein Empfänger definiert, so dass der WebSphere Partner Gateway-Empfänger für eingehende Dokumente des externen Partners über das entsprechende Transportprotokoll empfangsbereit ist.
- Im Profil des externen Partners (von dem das Dokument empfangen wurde) wurde B2B-Funktionalität definiert und aktiviert, so dass WebSphere Partner Gateway Dokumente aus dieser Quelle erwartet.
- Es sind Partnerverbindungen zwischen dem internen Partner und den externen Partnern vorhanden, so dass ein externer Partner und der interne Partner ein Dokument empfangen (oder senden) können.

**Anmerkung:** Sie sollten sich als Hubadministrator (Hubadmin) anmelden.

Eine vollständige Beschreibung der Konfiguration von WebSphere Partner Gateway zur Unterstützung einer Hub-Community finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

Wenn die externen Partner konfiguriert sind, müssen Sie WebSphere Partner Gateway zur Kommunikation mit einem Back-End-System konfigurieren. Dieser Abschnitt enthält die folgenden Informationen zur Einbindung eines Back-End-Systems in Ihre Hub-Community:

- „Dokumente an das Back-End-System senden“
- „Dokumente vom Back-End-System empfangen“ auf Seite 52

## Dokumente an das Back-End-System senden

Zum Senden eines Dokuments an das Back-End-System führt WebSphere Partner Gateway die folgenden Schritte aus:

1. Es empfängt ein Dokument von einem externen Partner.  
Der Empfänger ruft dieses Quelldokument aus einem Empfänger ab, der auf dem Hub für eingehende Nachrichten von diesem externen Partner und für das zugehörige Transportprotokoll definiert wurde. Beim Senden eines Dokuments an das Back-End-System ist das Quelldokument das von einem externen Partner empfangene Dokument, das daher als *Partnerdokument* bezeichnet wird.
2. Es konvertiert das Partnerdokument in das Zieldokument, welches das für das Back-End-System erforderliche Format hat.  
Diese Konvertierung in das Zieldokument wird durch den Document Manager von WebSphere Partner Gateway ausgeführt. Beim Senden eines Dokuments an das Back-End-System ist das Zieldokument das an das Back-End-System gesendete Dokument, das daher als *Back-End-Dokument* bezeichnet wird.
3. Es sendet das Back-End-Dokument an das Back-End-System.  
Der Document Manager sendet das Back-End-Dokument über ein Ziel, das auf dem Hub für ausgehende Nachrichten an das Back-End-System definiert wurde.

Damit der Hub ein Dokument an das Back-End-System senden kann, müssen Sie daher sicherstellen, dass die in Tabelle 21 zusammengefassten Konfigurationsschritte innerhalb von WebSphere Partner Gateway ausgeführt wurden.

Tabelle 21. Konfigurationsschritte zum Senden von Dokumenten an das Back-End-System

Konfigurationsschritt	Schritte in WebSphere Partner Gateway	Weitere Informationen
1. Definieren, wohin das Dokument zu senden ist.	1. Erstellen Sie ein Ziel zum Back-End-System.	„Zieleinheit für Partnerdokument definieren“ auf Seite 49
2. Definieren, wie das Dokument zu verarbeiten ist.	2. Erstellen Sie Dokumentenflussdefinitionen für das Quellen- und das Zielformat.  3. Aktivieren Sie die B2B-Funktionalität für die Dokumentenflussdefinition für das an das Back-End-System gesendete Dokument.  4. Erstellen Sie eine Interaktion für Dokumentenflussdefinitionen zwischen der Quellen- und der Zieldokumentenflussdefinition.	„Verarbeitungsweise des Partnerdokuments definieren“ auf Seite 49
3. Definieren, wie die Verbindung zum Back-End-System herzustellen ist.	5. Erstellen Sie eine Partnerverbindung, die Dokumente an das Back-End-System sendet.	„Verbindungsaufbau zum Back-End-System definieren“ auf Seite 51

## Zieleinheit für Partnerdokument definieren

Zum Senden von Dokumenten an das Back-End-System muss im Hub ein Ziel definiert werden. Dieses Ziel gibt das Ziel für die konvertierten Dokumente an. Es definiert also die (als URI angegebene) Position, an die der Hub das Back-End-Dokument sendet. Diese Position stimmt mit der Position überein, an der das Back-End-System für eingehende Nachrichten empfangsbereit ist. Das Ziel identifiziert den Eintrittspunkt in die Unternehmensanwendungsschicht (innerhalb des Back-End-Systems). Innerhalb von WebSphere Partner Gateway ist der Document Manager für die Prüfung auf ein Ziel zuständig. Wenn der Document Manager das Dokument verarbeitet hat, sendet er das konvertierte Dokument an das Back-End-System an der Position, die im Ziel definiert ist.

Gehen Sie wie folgt vor, um in WebSphere Partner Gateway ein Ziel zu definieren:

1. Klicken Sie auf **Kontenadmin > Profile**.
2. Klicken Sie auf **Suchen**, um eine Liste der Partner aufzurufen.
3. Wählen Sie das Symbol **Details anzeigen** neben dem Namen des internen Partners aus.
4. Klicken Sie auf **Ziele**.
5. Klicken Sie auf **Erstellen**.

Wenn Sie das Ziel definieren, geben Sie das Transportprotokoll an, das vom Hub und vom Back-End-System zur Übertragung des Back-End-Dokuments verwendet wird. Wie Tabelle 16 auf Seite 38 zu entnehmen ist, hängt die Auswahl des Transportprotokolls vom Format des Dokuments ab. Das Format beinhaltet den Pakettyp und das Geschäftsprotokoll, die in der zugehörigen Dokumentenflussdefinition definiert sind.

**Anmerkung:** Weitere Informationen zur Erstellung eines Ziels in WebSphere Partner Gateway finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

Die Auswahl des Transportprotokolls hängt außerdem von den Transportprotokollen ab, die von Ihrem speziellen Back-End-System unterstützt werden. Weitere Informationen finden Sie im Kapitel über die Integration Ihres speziellen Back-End-Systems in diesem Handbuch.

Sobald Sie ein gültiges Transportprotokoll für Ihr Dokument ausgewählt haben, können Sie die anderen Informationen angeben, die Sie zur Definition des Ziels in der Anzeige **Ziele** benötigen.

## Verarbeitungsweise des Partnerdokuments definieren

Damit der Document Manager das Partnerdokument verarbeiten kann, muss er das Format kennen, in das dieses Dokument zu konvertieren ist. Das heißt, er muss das Format des Back-End-Dokuments kennen. Als Teil der Back-End-Integration müssen Sie sicherstellen, dass die folgenden Entitäten innerhalb von WebSphere Partner Gateway definiert sind:

- Es müssen Dokumentdefinitionen zur Definition des Formats des Partnerdokuments und des Formats des Back-End-Dokuments vorhanden sein.
- Die B2B-Funktionalität des internen Partners muss eine Aktivierung der Dokumentenflussdefinition des Back-End-Dokuments als Ziel beinhalten.
- Eine Interaktion für Dokumentenflussdefinitionen muss vorhanden sein, die das Partnerdokument als Quelle und das Back-End-Dokument als Ziel kombiniert.

**Dokumentdefinition definieren:** Jede Dokumentdefinition definiert, wie WebSphere Partner Gateway ein bestimmtes Dokument verarbeitet.

Sie enthält den Pakettyp und das Geschäftsprotokoll des Dokuments. WebSphere Partner Gateway stellt einige vordefinierte Pakettyper und Protokolldefinitionen zur Verfügung. Wenn diese vordefinierten Formate Ihre Partner- und Back-End-Dokumente korrekt definieren, müssen Sie keine Dokumentdefinition definieren. Wenn die vordefinierten Formate Ihre Partner- und Back-End-Dokumente jedoch nicht geeignet definieren, müssen Sie eine gültige Dokumentdefinition für diese Dokumente erstellen. Zur Definition einer Dokumentenflussdefinition in WebSphere Partner Gateway verwenden Sie die Seite **Dokumentenflussdefinitionen verwalten (Hubadmin > Hubkonfiguration > Dokumentenflussdefinition > Dokumentenflussdefinition erstellen)**.

**Anmerkung:** Weitere Informationen zu vordefinierten Dokumentenflussdefinitionen sowie zur Erstellung von Dokumentenflussdefinitionen in WebSphere Partner Gateway finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

Für Back-End-Integration muss der Pakettyp des Back-End-Dokuments eine der folgenden Einstellungen haben:

- Paket 'None'
- Paket 'Backend Integration'

Sie müssen je nach Geschäftsprotokoll Ihres Dokuments und dem speziellen, von Ihnen verwendeten Back-End-System feststellen, welcher dieser Pakettyper zutrifft. Informationen zu Pakettyper für Back-End-Systeme finden Sie im Abschnitt „Zu verwendender Pakettyp“ auf Seite 25. Informationen zu unterstützten Back-End-Systemen finden Sie im Abschnitt „Zugriff auf Back-End-Anwendungen“ auf Seite 45.

**B2B-Funktionalität zum Senden einstellen:** Bevor der Document Manager das Quelldokument konvertieren kann, muss er feststellen, ob er das Format des gewünschten Zieldokuments verarbeiten kann. Hierzu überprüft der Document Manager die B2B-Funktionalität des Profils für den internen Partner, in der festgelegt ist, welche Dokumentdefinitionen für den internen Partner aktiviert wurden. Bei unterstützten Dokumentdefinitionen sind alle zugehörigen Komponentendokumententypen (wie Pakettyp, Geschäftsprotokoll und Dokumenttyp) aktiviert. Zum Aktivieren einer bestimmten Dokumentenflussdefinition verwenden Sie die Anzeige **B2B-Funktionalität** von WebSphere Partner Gateway. Führen Sie die folgenden Arbeitsschritte aus, um auf diese Anzeige zuzugreifen:

1. Klicken Sie auf **Kontenadmin > Profile > Partner**.
2. Klicken Sie auf **Suchen**, um eine Liste der Partner aufzurufen.
3. Wählen Sie das Symbol **Details anzeigen** neben dem Namen des internen Partners aus.
4. Klicken Sie auf **B2B-Funktionalitäten**.
5. Stellen Sie für die Back-End-Integration sicher, dass jeder Dokumenttyp, der an das Back-End-System gesendet wird, unter dem entsprechenden Paket, Protokoll und Dokumenttyp des Dokuments in **Ziel festlegen** aktiviert ist. Aktivieren Sie hierzu unter **Ziel festlegen** alle Dokumenttypkomponenten in der Dokumentdefinition des Back-End-Dokuments.
6. Wenn der Hub außerdem Dokumente vom Back-End-System empfangen soll, ist es möglicherweise sinnvoll, auch die dazu erforderliche B2B-Funktionalität zu aktivieren, solange die Anzeige **B2B-Funktionalität** noch geöffnet ist. In diesem Fall können Sie unter **Quelle festlegen** die Dokumenttypkomponente des Back-End-Dokuments aktivieren.

**Anmerkung:** Weitere Informationen zur Konfiguration der B2B-Funktionalität in WebSphere Partner Gateway finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

**Dokumenteninteraktion zum Senden definieren:** Damit der Document Manager feststellen kann, wie ein bestimmtes Partnerdokument konvertiert werden muss, ist es erforderlich, dass eine Interaktion lokalisiert werden kann, die die Dokumentenflussdefinitionen für das Partnerdokument und das Back-End-Dokument kombiniert und angibt, wer der Quellen- und wer der Zielpartner ist.

Wenn der Document Manager bereit ist, das konvertierte Dokument an das Back-End-System zu senden, muss er eine Partnerverbindung zwischen dem Quellenpartner und dem Zielpartner (Back-End-System) lokalisieren können. Damit eine Partnerverbindung definiert werden kann, muss jedoch eine gültige Interaktion zwischen dem Quellen- und dem Zieldokument vorhanden sein. Zum Definieren einer Interaktion für Dokumentenflussdefinitionen in WebSphere Partner Gateway klicken Sie auf folgende Optionen: **Hubadmin > Hubkonfiguration > Dokumentdefinition > Interaktionen verwalten > Interaktion erstellen**.

**Anmerkung:** Weitere Informationen zur Erstellung von Interaktionen für Dokumentenflussdefinitionen in WebSphere Partner Gateway finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

Zum Senden von Dokumenten an das Back-End-System definieren Sie eine Interaktion zwischen dem Quellen- und dem Zieldokument, wie dies in Tabelle 22 zusammengefasst ist.

*Tabelle 22. Erstellen einer Interaktion zum Senden eines Dokuments*

Abschnitt in 'Interaktionen verwalten'	Aktion
Quelle	Wählen Sie die Komponentendokumenttypen in der Dokumenttypdefinition des <i>Partnerdokuments</i> aus.
Ziel	Wählen Sie die Komponentendokumenttypen in der Dokumenttypdefinition des <i>Back-End-Dokuments</i> aus.

## **Verbindungsaufbau zum Back-End-System definieren**

Damit der Document Manager das konvertierte Dokument an das Back-End-System senden kann, muss eine gültige Partnerverbindung ermittelt werden, die den Quellen- und den Zielpartner sowie die Position angibt, über die diese beiden Partner kommunizieren. Zur Erstellung einer Partnerverbindung verwenden Sie die Seite **Verbindungen verwalten** in WebSphere Partner Gateway. Klicken Sie auf **Kontenadmin > Verbindungen**, um auf diese Anzeige zuzugreifen.

Für die Definition einer Partnerverbindung muss bereits eine Interaktion für Dokumentdefinitionen zwischen dem Quellendokument und dem Zieldokument vorhanden sein. In der Anzeige **Verbindungen verwalten** prüfen Sie zunächst, ob eine Interaktion vorhanden ist, indem Sie den Quellen- und den Zielpartner angeben. In Tabelle 23 sind die Partner aufgeführt, die auf der Seite **Verbindungen verwalten** zur Definition einer Partnerverbindung zum Senden eines Dokuments an das Back-End-System auszuwählen sind.

Tabelle 23. Erstellen einer Partnerverbindung zum Senden eines Dokuments

Dropdown-Liste 'Verbindungen verwalten'	Name des externen Partners
Quelle	Name des externen Partners, der das Dokument an den internen Partner sendet
Ziel	Name des internen Partners, der das Dokument vom externen Partner empfängt

Wenn Sie die Quelle und das Ziel angegeben haben, klicken Sie auf **Suchen**, um zu überprüfen, ob eine Interaktion für Dokumentenflussdefinitionen vorhanden ist. Wenn keine Interaktion vorhanden ist, müssen Sie eine Interaktion erstellen, bevor Sie mit der Erstellung einer Partnerverbindung fortfahren können. Wenn eine Interaktion vorhanden ist (eine, deren Quelle die Dokumentenflussdefinition des Partnerdokuments und deren Ziel die Dokumentenflussdefinition des Back-End-Dokuments ist), können Sie die Partnerverbindung für die Kommunikation mit dem Back-End-System konfigurieren.

**Anmerkung:** Weitere Informationen zur Erstellung einer Partnerverbindung in WebSphere Partner Gateway finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

Für die Back-End-Integration muss diese Partnerverbindung als Ziel das Ziel angegeben, das Sie im Abschnitt „Zieleinheit für Partnerdokument definieren“ auf Seite 49 definiert haben.

## Dokumente vom Back-End-System empfangen

Zum Empfangen eines Dokuments vom Back-End-System führt der Hub die folgenden Schritte aus:

1. Er empfängt ein Dokument vom Back-End-System.  
Der Empfänger von WebSphere Partner Gateway ruft dieses Quelldokument aus einem Empfänger ab, der im Hub für eingehende Nachrichten vom Back-End-System und für das zugehörige Transportprotokoll definiert wurde. Beim Empfangen eines Dokuments vom Back-End-System ist das Quelldokument das vom Back-End-System empfangene Dokument, das daher als *Back-End-Dokument* bezeichnet wird.
2. Er konvertiert das Back-End-Dokument in das Zieldokument, das in dem Format codiert ist, das vom externen Partner benötigt wird.  
Diese Konvertierung in das Zieldokument wird durch den Document Manager ausgeführt. Beim Empfangen eines Dokuments vom Back-End-System ist das Zieldokument das an einen externen Partner gesendete Dokument, das daher als *Partnerdokument* bezeichnet wird.
3. Er sendet das Partnerdokument an den vorgesehenen externen Partner.  
Der Document Manager sendet das Partnerdokument über ein Ziel, das im Hub für ausgehende Nachrichten an den entsprechenden externen Partner definiert wurde.

Damit der Hub ein Dokument vom Back-End-System empfangen kann, müssen Sie daher sicherstellen, dass die in Tabelle 24 zusammengefassten Konfigurationsschritte innerhalb von WebSphere Partner Gateway ausgeführt wurden.

Tabelle 24. Konfigurationsschritte zum Empfangen von Dokumenten vom Back-End-System

Konfigurationsschritt	Schritte in WebSphere Partner Gateway	Weitere Informationen
1. Definieren, wo das Dokument abzurufen ist.	1. Erstellen Sie einen Empfänger, der vom Back-End-System eingehende Nachrichten empfängt.	„Abrufposition für Back-End-Dokumente definieren“
2. Definieren, wie das Dokument zu verarbeiten ist.	2. Erstellen Sie Dokumentenflussdefinitionen für das Quellen- und das Zielformat.  3. Aktivieren Sie die B2B-Funktionalität für die Dokumentenflussdefinition für das vom Back-End-System empfangene Dokument.  4. Erstellen Sie eine Interaktion für Dokumentenflussdefinitionen zwischen der Quellen- und der Zieldokumentenflussdefinition.	„Verarbeitungsweise des Back-End-Dokuments definieren“ auf Seite 54
3. Definieren, wie die Verbindung zu WebSphere Partner Gateway herzustellen ist.	5. Erstellen Sie eine Partnerverbindung, die Dokumente an WebSphere Partner Gateway sendet.	„Verbindungsaufbau zu WebSphere Partner Gateway definieren“ auf Seite 55

### Abrufposition für Back-End-Dokumente definieren

Zum Empfangen von Dokumenten vom Back-End-System muss im Hub ein Empfänger definiert werden. Dieser Empfänger gibt die Quelle der Dokumente, d. h. die Position (in Form einer URI) an, unter der der Hub für eingehende Dokumente empfangsbereit ist. Diese Position stimmt mit der Position überein, an die das Back-End-System Dokumente sendet. Der Empfänger identifiziert den Eintrittspunkt in die Empfängerkomponente (Receiver) innerhalb von WebSphere Partner Gateway. In WebSphere Partner Gateway ist es der Empfänger, der Dokumente für WebSphere Partner Gateway empfängt. Wenn die Empfängerkomponente das Dokument verarbeitet hat, speichert sie das konvertierte Dokument im persistenten gemeinsamen Speicher zum späteren Abrufen durch den Document Manager.

Zum Definieren eines Empfängers in von WebSphere Partner Gateway klicken Sie auf **Hubadmin > Hubkonfiguration > Empfänger**.

**Anmerkung:** Weitere Informationen zur Erstellung eines Empfängers in WebSphere Partner Gateway finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

Wenn Sie den Empfänger definieren, geben Sie das Transportprotokoll an, das vom Hub und vom Back-End-System zur Übertragung des Back-End-Dokuments verwendet wird. Wie Tabelle 17 auf Seite 38 zu entnehmen ist, hängt die Auswahl des Transportprotokolls vom Format des Dokuments ab. Das Format beinhaltet den Pakettyp und das Geschäftsprotokoll, die in der zugehörigen Dokumentenflussdefinition definiert sind.

**Anmerkung:** Die Auswahl des Transportprotokolls hängt außerdem von den Transportprotokollen ab, die von Ihrem speziellen Back-End-System unterstützt werden. Weitere Informationen finden Sie im Kapitel über die Integration mit Ihrem speziellen Back-End-System in diesem Handbuch.

Sobald Sie ein gültiges Transportprotokoll für Ihr Dokument ausgewählt haben, können Sie die anderen Informationen angeben, die Sie zur Definition des Empfängers in der Anzeige **Empfängerdetails** benötigen.

### Verarbeitungsweise des Back-End-Dokuments definieren

Damit der Document Manager das Back-End-Dokument verarbeiten kann, muss er das Format kennen, in das dieses Dokument zu konvertieren ist. Das heißt, er muss das Format des Partnerdokuments kennen. Als Teil der Back-End-Integration müssen Sie sicherstellen, dass die in Tabelle 25 zusammengefassten Entitäten in WebSphere Partner Gateway definiert sind.

Tabelle 25. Vorgehensweise zum Konvertieren des Back-End-Dokuments

Schritt	Weitere Informationen
1. Es müssen Dokumentdefinitionen zur Definition des Formats des Partnerdokuments <i>und</i> des Formats des Back-End-Dokuments vorhanden sein.	„Dokumentdefinition definieren“ auf Seite 49
2. Die B2B-Funktionalität des internen Partners muss eine Aktivierung der Dokumentdefinition des Back-End-Dokuments als Quelle beinhalten.	„B2B-Funktionalität zum Empfangen einstellen“
3. Eine Interaktion für Dokumentdefinitionen muss vorhanden sein, die das Back-End-Dokument als Quelle und das Partnerdokument als Ziel kombiniert.	„Interaktion zum Empfangen definieren“

**B2B-Funktionalität zum Empfangen einstellen:** Eine Zusammenfassung der B2B-Funktionalität, wie sie für die Back-End-Integration erforderlich ist, finden Sie im Abschnitt „B2B-Funktionalität zum Senden einstellen“ auf Seite 50.

Falls Sie diesen Arbeitsschritt noch nicht ausgeführt haben, aktivieren Sie nun unter **Quelle festlegen** die Dokumenttypkomponente des Back-End-Dokuments.

**Interaktion zum Empfangen definieren:** Eine Zusammenfassung der Interaktionen für Dokumentenflussdefinitionen, wie sie für die Back-End-Integration erforderlich sind, finden Sie im Abschnitt „Dokumenteninteraktion zum Senden definieren“ auf Seite 51. Dieser Abschnitt enthält eine Zusammenfassung darüber, wie die Interaktion zum Empfangen eines Dokuments vom Back-End-System definiert wird.

Zum Empfangen von Dokumenten vom Back-End-System definieren Sie eine Interaktion zwischen den Dokumentenflussdefinitionen des Quellen- und des Zieldokuments, wie dies in Tabelle 26 *zusammengefasst ist*.

Tabelle 26. Erstellen einer Interaktion zum Empfangen eines Dokuments

Abschnitt in 'Interaktionen verwalten'	Aktion
Quelle	Wählen Sie die Komponentendokumenttypen in der Dokumentenflussdefinition des <i>Back-End-Dokuments</i> aus.
Ziel	Wählen Sie die Komponentendokumenttypen in der Dokumentenflussdefinition des <i>Partnerdokuments</i> aus.

## Verbindungsaufbau zu WebSphere Partner Gateway definieren

Damit der Document Manager das Dokument vom Back-End-System verarbeiten kann, muss eine gültige Verbindung ermittelt werden, die den Quellen- und den Zielpartner sowie die Position angibt, über die diese beiden Partner kommunizieren. Eine Zusammenfassung der Verbindungen, wie sie für die Back-End-Integration erforderlich sind, finden Sie im Abschnitt „Verbindungsaufbau zum Back-End-System definieren“ auf Seite 51.

In Tabelle 27 sind die Partner der Anzeige **Verbindungen verwalten** in WebSphere Partner Gateway aufgeführt, die zur Definition einer Partnerverbindung zum Empfangen eines Dokuments vom Back-End-System auszuwählen sind.

Tabelle 27. Erstellen einer Partnerverbindung zum Empfangen eines Dokuments

Dropdown-Liste 'Verbindungen verwalten'	Name des externen Partners
Quelle	Name des internen Partners
Ziel	Name des externen Partners, der das Dokument vom internen Partner empfängt

Wenn Sie die Quelle und das Ziel angegeben haben, klicken Sie auf **Suchen**, um zu überprüfen, ob eine Interaktion für Dokumentenflussdefinitionen vorhanden ist. Wenn keine Interaktion vorhanden ist, müssen Sie eine Interaktion erstellen, *bevor* Sie mit der Erstellung einer Partnerverbindung fortfahren können. Wenn eine Interaktion vorhanden ist (eine, deren Quelle die Dokumentenflussdefinition des Back-End-Dokuments und deren Ziel die Dokumentdefinition des Teilnehmersdokuments ist), können Sie die Partnerverbindung für die Kommunikation mit dem Back-End-System konfigurieren.

**Anmerkung:** Weitere Informationen zur Erstellung einer Partnerverbindung in WebSphere Partner Gateway finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.



---

## **Teil 2. Integration mit WebSphere Process Server**

In den folgenden Kapiteln wird die Vorgehensweise für die Integration von WebSphere Partner Gateway mit WebSphere Process Server beschrieben.



---

## Kapitel 3. Einführung zur WebSphere Process Server-Integration

Dieses Kapitel enthält eine Übersicht darüber, wie die Integration von WebSphere Partner Gateway mit WebSphere Process Server durchgeführt werden kann.

**Anmerkung:** Eine Beschreibung des allgemeinen Verfahrens zur Integration von WebSphere Partner Gateway mit einem Back-End-System finden Sie in Kapitel 2, „Back-End-Integration planen“, auf Seite 9.

Dieses Kapitel enthält die folgenden Abschnitte:

- „Übersicht“
- „Integration mit WebSphere Process Server planen“ auf Seite 61
- „Integration von WebSphere Partner Gateway mit WebSphere Process Server - Aufgabenübersicht“ auf Seite 64
- „Paketnachricht der Back-End-Integration verarbeiten“ auf Seite 66
  - „Klasse BCGBackEndIntegrationDataBindingUtil“ auf Seite 79
  - „Klasse BCGBackEndIntegrationJMSDataBindingImpl“ auf Seite 85

---

### Übersicht

In diesem Abschnitt wird beschrieben, wie mit WebSphere Partner Gateway B2B-Funktionalität für WebSphere Process Server bereitgestellt werden kann.

**Anmerkung:** Detaillierte Informationen zu WebSphere Process Server finden Sie im Information Center für WebSphere Process Server.

WebSphere Process Server verwendet die B2B-Funktionalität von WebSphere Partner Gateway zur Verwaltung von Interaktionen mit Handelspartnern, die in WebSphere Partner Gateway als *externe Partner* bezeichnet werden. Beispiel: Ein Service in WebSphere Process Server muss ein Dokument an einen externen Partner senden. WebSphere Process Server sendet das Dokument an WebSphere Partner Gateway. WebSphere Partner Gateway ermittelt die Transformationszuordnung (falls vorhanden), die das Dokument in das vom externen Partner erwartete Format transformieren soll. Ferner verarbeitet WebSphere Partner Gateway sämtliche Profilinginformationen für die externen Partner.

Beispiel: WebSphere Process Server muss ein Dokument an mehrere externe Partner senden. Die externen Partner empfangen das Dokument in unterschiedlichen Formaten. WebSphere Process Server muss nur einen Formattyp verarbeiten (das Format, das an WebSphere Partner Gateway gesendet wird). WebSphere Partner Gateway ist für die Interaktion mit den externen Partnern verantwortlich.

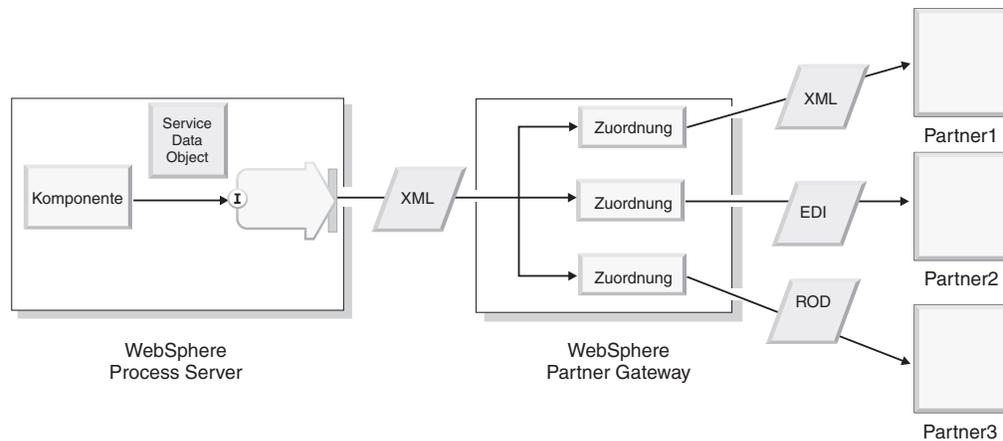


Abbildung 15. WebSphere Partner Gateway sendet Dokumente in mehreren Formaten an externe Partner

Sie können die Transformationszuordnungen entwickeln, die das Dokument in das vom externen Partner geforderte Format konvertieren, oder Sie können Zuordnungen aus dem Data Interchange Services-Clientprogramm importieren.

Wenn WebSphere Process Server Dokumente von externen Partnern empfängt, werden diese Dokumente ebenfalls von WebSphere Partner Gateway verarbeitet. Die Dokumente können verschiedene Formate aufweisen. WebSphere Partner Gateway transformiert die Dokumente und sendet sie an das Ziel, das in WebSphere Process Server für den internen Partner definiert wurde.

## Kommunikation zwischen WebSphere Process Server und WebSphere Partner Gateway

WebSphere Partner Gateway sendet ein Dokument von einem externen Partner an WebSphere Process Server, damit das Dokument von einem Service in WebSphere Process Server verarbeitet werden kann. Die *Exportbindung* eines Service in WebSphere Process Server macht es möglich, dass andere Anwendungen (einschließlich WebSphere Partner Gateway) diesen Service nutzen können.

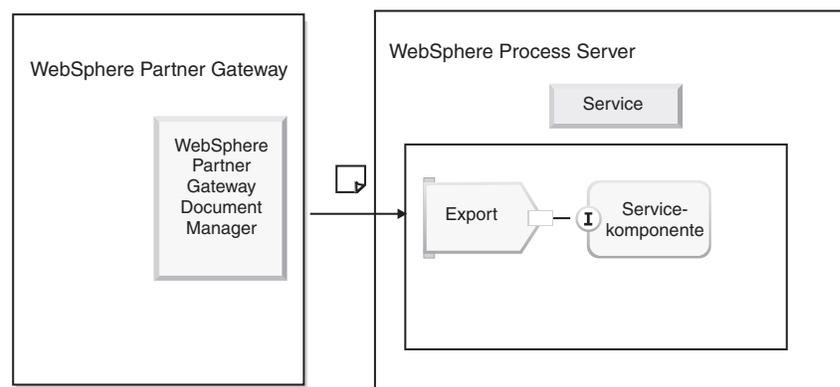


Abbildung 16. Nutzung der Exportbindung durch WebSphere Partner Gateway, um Dokumente an WebSphere Process Server zu senden

Die WebSphere Process Server-Komponente verfügt über eine Schnittstelle, die den Service (die verfügbaren Methoden sowie die Eingabe- und Ausgabedaten) und eine Bindung (in diesem Fall eine Exportbindung) beschreibt.

In Abb. 16 auf Seite 60 wird generisch dargestellt, wie die Exportbindung einen Service in WebSphere Process Server aufruft. Der zum Senden der Nachricht verwendete Transporttyp (z. B. JMS oder HTTP) hat Auswirkungen darauf, wohin die Nachricht gesendet und wie sie abgerufen wird (siehe „Von WebSphere Process Server unterstützte Nachrichtentransporte“ auf Seite 62).

Wenn ein Service in WebSphere Process Server ein Geschäftsdokument an einen externen Partner senden möchte, wird analog die entsprechende *Importbindung* verwendet. Importe geben Services außerhalb eines Moduls an, so dass sie innerhalb des Moduls aufgerufen werden können. In diesem Fall verwendet WebSphere Process Server die Importbindung zum Aufrufen von WebSphere Partner Gateway. WebSphere Partner Gateway verarbeitet das Dokument und sendet es an den externen Partner.

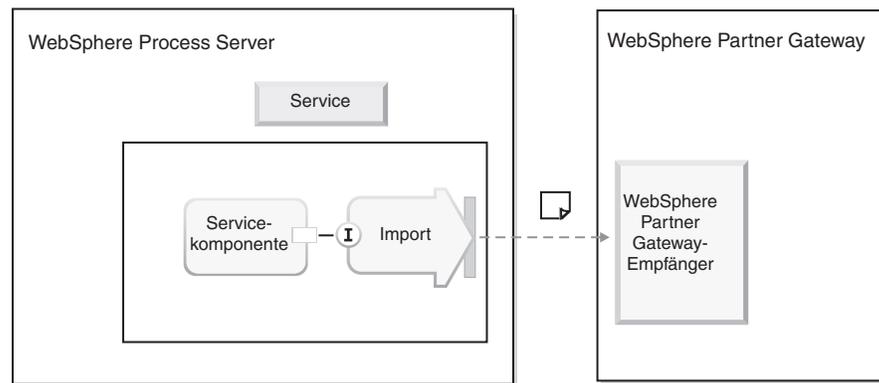


Abbildung 17. Nutzung der Importbindung durch WebSphere Process Server, um WebSphere Partner Gateway aufzurufen

## Integration mit WebSphere Process Server planen

Zur Planung der Integration mit WebSphere Process Server befolgen Sie die in Kapitel 2, „Back-End-Integration planen“, auf Seite 9 aufgeführten Schritte.

### Von WebSphere Partner Gateway unterstützte WebSphere Process Server-Versionen

WebSphere Partner Gateway Version 6.1 unterstützt die Integration mit WebSphere Process Server Version 6.0.2.

WebSphere Process Server ist für verschiedene Plattformen verfügbar, z. B. Windows 2000 und mehrere UNIX-basierte Plattformen. Weitere Informationen hierzu finden Sie im WebSphere Process Server-Installationshandbuch im Information Center für WebSphere Process Server.

### Unterstützte Installationsszenarios

In der folgenden Tabelle wird in jeder Zeile eine unterstützte Kombination von WebSphere Partner Gateway-Installationen und WebSphere Process Server-Installationen angezeigt.

**Anmerkung:** Während der Installation erstellt WebSphere Partner Gateway separate WebSphere Application Server-Profile für die einzelnen Komponenten. Diese

Profile werden nur von WebSphere Partner Gateway verwendet. Fügen Sie WebSphere Process Server oder andere WebSphere Application Server-Anwendungen nicht in diese Profile ein.

Tabelle 28. Unterstützte Installationsszenarios

WebSphere Partner Gateway-Installation	WebSphere Process Server-Installation	Details
Wird in WebSphere Application Server 6.1 oder in mindestens einer Instanz von WebSphere Application Server Network Deployment 6.1 installiert.	Das Installationsprogramm von WebSphere Process Server installiert die unterstützte WebSphere Application Server ND-Instanz zusammen mit WebSphere Process Server.	
Wird in WebSphere Application Server 6.1 oder in mindestens einer Instanz von WebSphere Application Server Network Deployment 6.1 installiert.	Wird in derselben Installation, aber nicht in derselben Instanz von WebSphere Application Server 6.1 installiert, in der WebSphere Partner Gateway installiert ist. Das für WebSphere Process Server verwendete Profil muss sich von dem Profil unterscheiden, das von den WebSphere Partner Gateway-Komponenten verwendet wird.	Wird nur auf den Plattformen (Betriebssysteme und Versionen) unterstützt, die von WebSphere Partner Gateway und von WebSphere Process Server unterstützt werden.

## Von WebSphere Process Server unterstützte Nachrichtentransporte

Wenn WebSphere Partner Gateway über ein bestimmtes Nachrichtentransportprotokoll eine Nachricht an WebSphere Process Server sendet, wird die Nachricht von dem transportspezifischen Ziel, das für den internen Partner definiert wurde, an den WebSphere Process Server-Endpunkt gesendet. Die Nachricht wird von WebSphere Process Server aus dem Endpunkt abgerufen und verarbeitet. Der Typ des Nachrichtentransports bestimmt, wie die Nachricht weiter verarbeitet wird, nachdem sie von WebSphere Process Server empfangen wurde:

- Für den JMS-Transport muss ein JMS-Export konfiguriert werden, um die Nachricht aus der JMS-Warteschlange abzurufen.
- Für den HTTP-Transport muss ein Servlet in WebSphere Process Server erstellt werden, das den Empfang der Nachricht von WebSphere Partner Gateway abwickelt.
- Für SOAP-Dokumente, die über das HTTP-Transportprotokoll gesendet werden, muss eine Web-Service-Exportbindung in WebSphere Process Server konfiguriert werden, um die SOAP-Anforderung abzurufen.
- Für das Dateisystem muss ein eingehender Adapter für Flachdateien (Flat Files) definiert werden, der die Nachricht an WebSphere Process Server weiterleitet.

Wenn Sie Nachrichten von WebSphere Process Server an WebSphere Partner Gateway senden, werden diese Nachrichten an den transportspezifischen Empfänger (z. B. eine JMS-Warteschlange oder eine URL-Adresse) in WebSphere Partner Gateway gesendet. Der Typ des Nachrichtentransports bestimmt, wie die Nachricht gesendet wird.

- Für den JMS-Transport muss ein JMS-Import konfiguriert werden, um die Nachricht an die JMS-Warteschlange zu senden.
- Für den HTTP-Transport muss eine Komponente in WebSphere Process Server erstellt werden, die eine HTTP POST-Operation für die URL-Adresse ausführt, die für den WebSphere Partner Gateway-Empfänger angegeben wurde.
- Für SOAP-Dokumente, die über das HTTP-Transportprotokoll gesendet werden, muss eine Web-Service-Importbindung in WebSphere Process Server konfiguriert werden, um die SOAP-Anforderung an eine URL-Adresse von WebSphere Partner Gateway zu senden.
- Für das Dateisystem muss ein abgehender Adapter für Flachdateien (Flat Files) definiert werden, der die Nachricht an ein Verzeichnis in WebSphere Partner Gateway weiterleitet.

Informationen zum Senden und Empfangen von Nachrichten mit Hilfe der unterstützten Transporte finden Sie in den folgenden Abschnitten:

- Kapitel 4, „Integration von WebSphere Process Server mit HTTP als Transportprotokoll“, auf Seite 89
- Kapitel 5, „Integration von WebSphere Process Server mit JMS als Transportprotokoll“, auf Seite 97
- Kapitel 6, „Integration von WebSphere Process Server mit SOAP/HTTP“, auf Seite 109
- Kapitel 7, „Integration von WebSphere Process Server mit Dateisystem als Transportprotokoll“, auf Seite 115

## Unterstützung für WebSphere Process Server-Integration

In diesem Abschnitt werden die Muster, die Dokumentation und die Dienstprogrammklassen beschrieben, die von WebSphere Partner Gateway bereitgestellt werden, um Sie bei der WebSphere Process Server-Integration zu unterstützen.

### Muster

WebSphere Partner Gateway stellt Muster für die Verwendung des JMS-Transportprotokolls bereit, die Sie beim Integrationsprozess mit WebSphere Process Server unterstützen. Diese Muster befinden sich im folgenden Unterverzeichnis des Produktverzeichnisses von WebSphere Partner Gateway:

Integration/WBI/WPG/samples

In Tabelle 29 sind die Unterverzeichnisse des Verzeichnisses `samples` aufgelistet.

Tabelle 29. Muster für die WebSphere Process Server-Integration

Mustertyp	Unterverzeichnis 'samples'
Allgemeine Muster	JMS
RosettaNet-spezifische Muster	RosettaNet/JMS

### Dokumentation

Zusätzlich zu den Informationen in diesem Dokument stellt WebSphere Partner Gateway die Dokumentation *PIP Sample for WebSphere Process Server* bereit, die eine schrittweise Anleitung für das Konfigurieren eines PIP-Flusses zwischen einem externen Partner und WebSphere Process Server enthält.

## Dienstprogrammklassen

WebSphere Partner Gateway stellt zwei Dienstprogrammklassen bereit, die Sie zum Transformieren einer Nachricht des Back-End-Integrationspakets in ein Geschäftsobjekt (und umgekehrt) verwenden können:

- Klasse `BCGBackEndIntegrationDataBindingUtil`  
Diese Klasse implementiert die `DataBinding`-Schnittstelle und stellt Dienstprogramm-Methoden zum Lesen und Schreiben von Zeichenfolgen, Datenströmen und Bytefeldgruppen bereit. Die können diese Klasse unverändert übernehmen oder mit Hilfe dieser Klasse eine neue Datenbindung entwickeln.
- Klasse `BCGBackEndIntegrationJMSDataBindingImpl`  
Diese Klasse implementiert die `JMSDataBinding`-Schnittstelle. Diese Klasse kann in SCA-JMS-Export- bzw. -Importbindungen (SCA - Service Component Architecture) angegeben werden. Sie erstellt ein Datenobjekt aus einer JMS-Nachricht, die Nutzdaten enthält, oder schreibt das Datenobjekt in eine JMS-Nachricht.

---

## Integration von WebSphere Partner Gateway mit WebSphere Process Server - Aufgabenübersicht

In diesem Abschnitt sind die Aufgaben aufgelistet, die Sie ausführen müssen, damit WebSphere Partner Gateway Dokumente an WebSphere Process Server senden oder von WebSphere Process Server empfangen kann. Dieser Abschnitt bildet die Grundlage für die transportspezifischen Kapitel, in denen die Integration mit WebSphere Process Server beschrieben wird.

### Auf dem WebSphere Partner Gateway-System

Dieser Abschnitt enthält eine sehr kurze Übersicht über die Aufgaben, die Sie zum Konfigurieren des Hubs ausführen müssen, damit Sie Dokumente an WebSphere Partner Gateway senden und von WebSphere Partner Gateway empfangen können. Diese Aufgaben werden detailliert im Handbuch *WebSphere Partner Gateway Hub-konfiguration* beschrieben und über die Community Console von WebSphere Partner Gateway ausgeführt.

- Erstellen Sie einen transportspezifischen Empfänger auf dem Hub, um Dokumente zu empfangen, die von WebSphere Process Server oder von externen Partnern an den Hub gesendet werden.
- Erstellen Sie gegebenenfalls ein Profil für den internen Partner, das ein transportspezifisches Ziel enthält, das WebSphere Partner Gateway zum Senden von Dokumenten an WebSphere Process Server verwendet.
- Erstellen Sie Profile für externe Partner, die transportspezifische Ziele enthalten, die WebSphere Partner Gateway zum Senden von Dokumenten an die Partner verwendet.
- Importieren Sie WSDL-Dateien, Transformationszuordnungen, RosettaNet-Pakete oder andere Dokumentdefinitionsmechanismen, damit eine Dokumentdefinition für den ausgetauschten Dokumenttyp auf der Seite für Dokumentenflussdefinitionen der Community Console von WebSphere Partner Gateway angezeigt wird.
- Erstellen Sie Interaktionen zwischen den Dokumenttypen, die der Hub von WebSphere Process Server oder von einem externen Partner empfängt, und den Dokumenttypen, die der Hub an WebSphere Process Server oder an den externen Partner sendet.
- Erstellen Sie im Profil des internen Partners und in den Profilen der externen Partner B2B-Funktionalität, um die Dokumenttypen anzugeben, die gesendet und empfangen werden können.

- Erstellen Sie Partnerverbindungen zwischen dem internen Partner und den externen Partnern, um den Quellenpartner (Absender des Dokuments), den Zielpartner (Empfänger des Dokuments) und die Aktion anzugeben, die der Hub gegebenenfalls zum Transformieren des Dokuments ausführen soll.

## Auf dem WebSphere Process Server-System

Das Modul ist das WebSphere Process Server-Artefakt, mit dessen Hilfe ein Service zusammengesetzt und implementiert wird. Der erste Schritt besteht also in der Erstellung eines Moduls unter Verwendung von WebSphere Integration Developer.

Nach Erstellung des Moduls erstellen Sie die Komponenten sowie die zugehörigen Schnittstellen und geben dann die verwendete Bindung an.

1. Geben Sie eine Schnittstelle für die Komponente an. Sie können eine Schnittstelle importieren (z. B. eine vorhandene WSDL-Datei), oder Sie können die Schnittstelle erstellen.

Wenn Sie eine Schnittstelle erstellen, definieren Sie mindestens eine Operation, die von der Komponente ausgeführt wird, sowie die Ein- und Ausgaben, die von der Komponente erwartet werden.

Die Schnittstelle für eine Komponente kann eine WSDL-Datei sein, oder Sie können eine Java-Implementierung für die Schnittstelle verwenden. Informationen darüber, in welchen Fällen eine WSDL- und in welchen Fällen eine Java-Schnittstelle erstellt werden muss, finden Sie in der WebSphere Process Server-Dokumentation.

2. Geben Sie eine Implementierung für die Komponente an. Sie können eine Implementierung importieren (z. B. ein vorhandenes Java-Programm), oder Sie können die Implementierung erstellen.
3. Erstellen Sie die Anwendung mit Hilfe des Assembly-Editors von WebSphere Integration Developer. Sie erstellen eine Servicekomponente und geben dann die Schnittstelle für die Komponente an, die Sie in Schritt 1 erstellt oder importiert haben. Ferner geben Sie die Implementierung an, die Sie in Schritt 2 erstellt oder importiert haben.
4. Erstellen Sie eine Exportbindung, damit WebSphere Partner Gateway ein Dokument an den Service senden kann, oder erstellen Sie eine Importbindung, damit WebSphere Process Server ein Dokument an WebSphere Partner Gateway senden kann.

Geben Sie bei der Erstellung der Bindung die Informationen an, die zum Senden und Empfangen von Dokumenten erforderlich sind. In der Definition für die JMS-Bindung geben Sie beispielsweise die JMS-Warteschlange, den JMS-Bus sowie die Datenbindung an, die für die Transformation eines Geschäftsdokuments in ein bzw. aus einem Geschäftsobjekt verwendet werden soll. (Informationen zur Verwendung und zu den Anforderungen einer Datenbindung finden Sie im Abschnitt „Paketnachricht der Back-End-Integration verarbeiten“ auf Seite 66.) Die transportspezifischen Anforderungen für die Servicebindungen werden in späteren Kapiteln beschrieben.

5. Nach Zusammenstellung der Komponenten, aus denen sich das Modul zusammensetzt, können Sie das Modul implementieren.

---

## Paketnachricht der Back-End-Integration verarbeiten

Für bestimmte Protokolle, wie z. B. RosettaNet, geht WebSphere Partner Gateway davon aus, dass die Back-End-Anwendungen das Back-End-Integrationspaket verwenden. WebSphere Partner Gateway unterstützt die Verwendung von Back-End-Integrationspaketen über HTTP- und JMS-Protokolle. Dazu müssen die WebSphere Process Server-Services in der Lage sein, Paketnachrichten der Back-End-Integration zu verarbeiten. WebSphere Process Server-Services verwenden Geschäftsobjekte. Zum Senden von Paketnachrichten der Back-End-Integration an WebSphere Partner Gateway müssen die WebSphere Process Server-Services deshalb die Geschäftsobjekte in Paketnachrichten der Back-End-Integration serialisieren. Damit Paketnachrichten der Back-End-Integration von WebSphere Partner Gateway empfangen werden können, müssen die Paketnachrichten der Back-End-Integration durch die WebSphere Process Server-Services in Geschäftsobjekte entserialisiert werden.

In diesem Abschnitt werden die Dienstprogramm-API für Datenbindung und die JMS-Datenbindung beschrieben, die von WebSphere Partner Gateway bereitgestellt werden und von den WebSphere Process Server-Services zur Verarbeitung von Paketnachrichten der Back-End-Integration verwendet werden können.

Datenbindung bezieht sich auf einen Mechanismus mit den folgenden Aufgaben:

- Transformation eines von WebSphere Process Server gesendeten Geschäftsobjekts in ein Geschäftsdokument, das von WebSphere Partner Gateway verarbeitet werden kann.
- Transformation eines von WebSphere Partner Gateway gesendeten Geschäftsdokuments in ein Geschäftsobjekt, das von WebSphere Process Server verarbeitet werden kann.

Die Dienstprogramm-API für Datenbindung und die JMS-Datenbindung, die von WebSphere Partner Gateway für Back-End-Integrationspakete bereitgestellt werden, ermöglichen die Erstellung einer Paketnachricht der Back-End-Integration aus einem Geschäftsobjekt bzw. die Konvertierung einer Paketnachricht der Back-End-Integration in ein Geschäftsobjekt. Die von WebSphere Partner Gateway bereitgestellte Datenbindung kann beim Senden und Empfangen von Dokumenten mit dem Back-End-Integrationspaket unverändert verwendet werden. Wenn Sie die Verarbeitung jedoch anpassen möchten, können Sie mit Hilfe der von WebSphere Partner Gateway bereitgestellten Dienstprogramm-API für Datenbindung eine eigene Datenbindung erstellen.

### JMS-Transport

Wenn Sie als Transportprotokoll JMS verwenden und ein XML-Dokument ohne Back-End-Integrationspaket senden oder empfangen (d. h. als Pakettyp wurde 'None' angegeben), können Sie die von WebSphere Process Server bereitgestellte JMS-Standardbindung verwenden oder mit Hilfe der Geschäftsobjekt-APIs von WebSphere Process Server eine eigene Datenbindung erstellen. Wenn Sie ein Nicht-XML-Dokument ohne Back-End-Integrationspaket senden oder empfangen (d. h. als Pakettyp wurde 'None' angegeben), müssen Sie eine eigene Datenbindung erstellen. Die JMS-Bindung wird bei der Konfiguration der SCA-JMS-Export- bzw. -Importbindung ausgewählt. Weitere Informationen zur Verwendung der JMS-Standardbindung finden Sie im Information Center für WebSphere Process Server.

Wenn Sie als Transportprotokoll JMS verwenden und Dokumente mit dem Back-End-Integrationspaket senden oder empfangen, können Sie die Datenbindung `BCGBackEndIntegrationJMSDataBindingImpl` unverändert übernehmen. Wenn Sie die Verarbeitung anpassen möchten, können Sie mit `BCGBackEndIntegrationJMSDataBindingImpl` auch eine eigene Datenbindung erstellen. Zur Verwendung von `BCGBackEndIntegrationJMSDataBindingImpl` müssen Sie diese Klasse als Datenbindung in der SCA-JMS-Exportbindung bzw. in der SCA-JMS-Importbindung konfigurieren.

- Für die Verarbeitung von Anforderungen ruft der Export die Methode 'read' der Schnittstelle `MSDataBinding` auf, um die JMS-Nachricht des Back-End-Integrationspakets von WebSphere Partner Gateway in ein Geschäftsobjekt zu konvertieren. Für die Verarbeitung von Anforderungen/Antworten (request-response) ruft der Export unter Umständen auch die Methode 'write' auf.
- Für die Verarbeitung von Anforderungen ruft der Import die Methode 'write' der Schnittstelle `JMSDataBinding` auf, um aus einem Geschäftsobjekt eine JMS-Nachricht des Back-End-Integrationspakets zu erstellen, bevor die Nachricht an WebSphere Partner Gateway gesendet wird. Für die Verarbeitung von Anforderungen/Antworten (request-response) ruft der Import unter Umständen auch die Methode 'read' auf, um die Antwort des Service lesen zu können.

Weitere Informationen darüber, wie und wann die Methoden der Datenbindung aufgerufen werden, finden Sie im Information Center für WebSphere Process Server.

## HTTP-Transport

Wenn Sie HTTP als Transportprotokoll verwenden, können Sie eine Klasse für HTTP-Datenbindung schreiben, die die generische Klasse `BCGBackendIntegrationDataBindingUtil` erweitert und außer Kraft setzt.

- Ein HTTP-Servlet kann Methoden von `BCGBackendIntegrationDataBindingUtil` aufrufen, um ein Geschäftsobjekt aus dem HTTP-Datenstrom zu erstellen, der von WebSphere Partner Gateway gesendet wird. Das Servlet kann dieses Geschäftsobjekt dann zum Aufrufen eines Service verwenden.
- Eine Komponente kann geschrieben werden, um eine HTTP POST-Operation für WebSphere Partner Gateway auszuführen. Dann kann ein Service diese Komponente mit dem Geschäftsobjekt aufrufen. Wenn die Komponente eine Pakettachricht der Back-End-Integration erfordert, kann sie die Nachricht mit Hilfe der Klasse `BCGBackendIntegrationDataBindingUtil` aus dem Geschäftsobjekt erstellen.

**Anmerkung:** WebSphere Partner Gateway unterstützt für Back-End-Integrationspakete Aufrufe, die nur Anforderungen (request-only) verwenden. Mit der Datenbindung des Back-End-Integrationspakets können jedoch Datenflüsse entwickelt werden, die Anforderungen und Antworten (request-response) verwenden (über WebSphere Partner Gateway-Benutzerexits). Diese Dienstprogrammklasse kann als Basis für die Konvertierung eines Antwortgeschäftsobjekts in einen Antwortdatenstrom (oder umgekehrt) verwendet werden.

## Geschäftsobjekt der höchsten Ebene und untergeordnete Geschäftsobjekte

Dieser Abschnitt beschreibt das Geschäftsobjekt der höchsten Ebene sowie die untergeordneten Geschäftsobjekte, die von der Datenbindung des Back-End-Integrationspakets verwendet werden.

### Geschäftsobjekt der höchsten Ebene

Zur Verwendung der Datenbindung des Back-End-Integrationspakets erstellen Sie ein Geschäftsobjekt der höchsten Ebene, das über drei untergeordnete Attribute verfügt:

*Tabelle 30. Geschäftsobjekt der höchsten Ebene*

Attribut	Typ
payload	Geschäftsobjekt für Nutzdatencontainer. Der Typ kann beliebig sein, aber das Attribut muss die in Tabelle 31 auf Seite 69 oder in Tabelle 32 auf Seite 70 angegebenen Eigenschaften aufweisen.
attachment	Geschäftsobjekt für Anhangscontainer. Der Typ kann beliebig sein, aber das Attribut muss die in Tabelle 33 auf Seite 71 angegebenen Eigenschaften aufweisen.
packagingHeaders	Geschäftsobjekt für Paketheader. Der Typ kann beliebig sein, aber das Attribut muss die in Tabelle 34 auf Seite 71 angegebenen Eigenschaften aufweisen.

Bei der Konvertierung einer Paketnachricht der Back-End-Integration in ein Objekt der höchsten Ebene wird das Objekt der höchsten Ebene mit den Daten aus der Nachricht gefüllt. Bei der Konvertierung eines Objekts der höchsten Ebene in eine Paketnachricht der Back-End-Integration stellt das Objekt der höchsten Ebene die Eingabe für die Datenbindung dar.

Die folgende Abbildung zeigt das Geschäftsobjekt der höchsten Ebene sowie die zugehörigen untergeordneten Objekte:

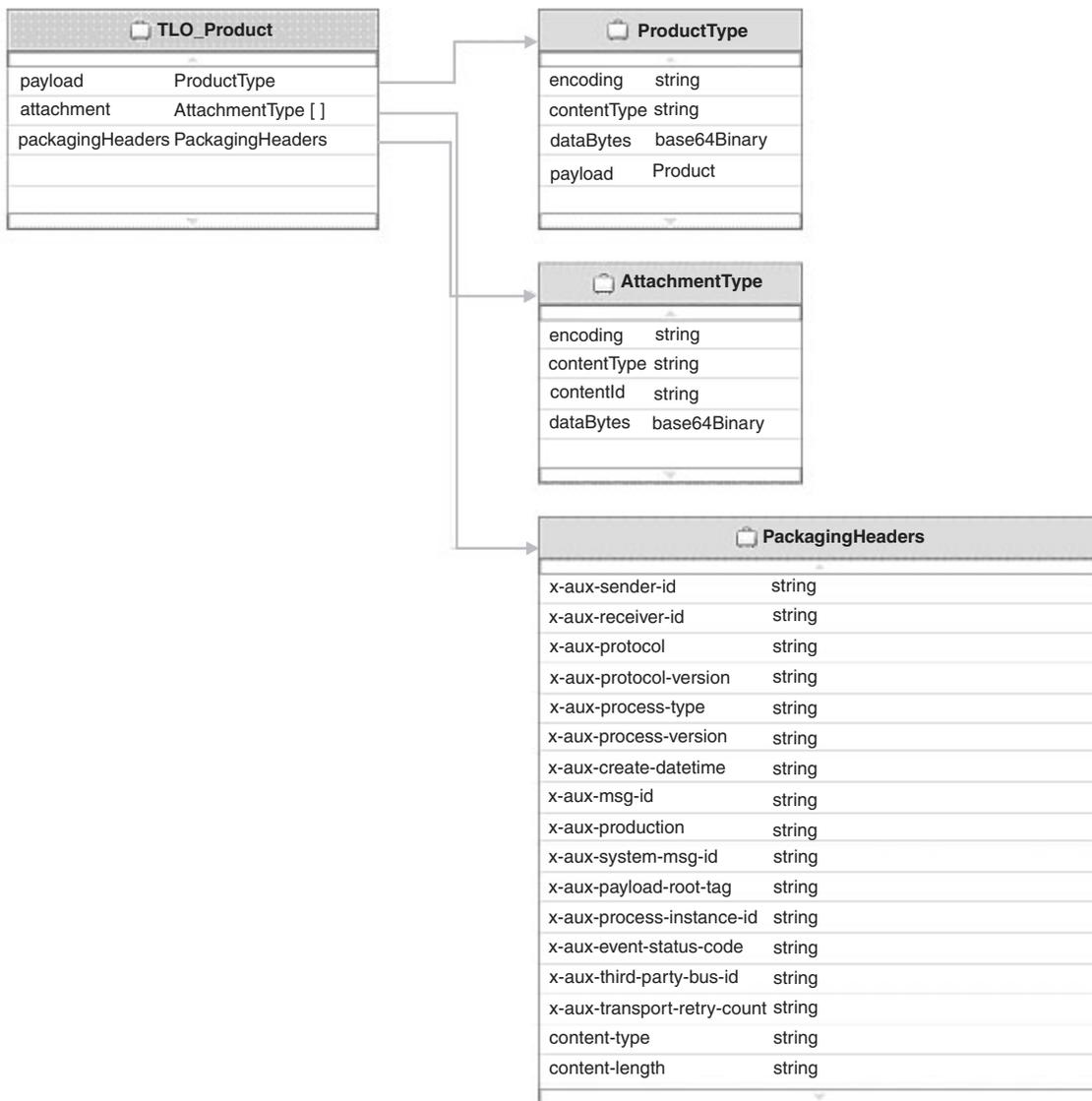


Abbildung 18. Geschäftsobjekt der höchsten Ebene

### Geschäftsobjekt für Nutzdatencontainer

Die Struktur des Geschäftsobjekts für Nutzdatencontainer hängt davon ab, ob es sich um XML-Nutzdaten handelt oder nicht.

**XML-Nutzdaten:** Wenn es sich um XML-Nutzdaten handelt, verfügt der Nutzdatencontainer über die folgenden Eigenschaften:

Tabelle 31. Objekt für Nutzdatencontainer für XML-Nutzdaten

Attribut	Typ
contentType	xsd:string
encoding	xsd:string
role	xsd:string

Tabelle 31. Objekt für Nutzdatencontainer für XML-Nutzdaten (Forts.)

Attribut	Typ
schema	xsd:string
schemaVersion	xsd:string
description	xsd:string
dataBytes	xsd:base64Binary
payload	Ein Verweis auf das Geschäftsobjekt für Nutzdaten. Wenn zum Beispiel die Nutzdaten selbst in einem Pip3A4PurchaseOrderRequest-Objekt enthalten sind, lautet der Typ des Nutzdatenattributs PIP3A4PurchaseOrderRequest.

Im Abschnitt „Geschäftsobjekte für RosettaNet Service Content erstellen“ auf Seite 72 wird beschrieben, wie Sie ein Geschäftsobjekt für einen PIP erstellen können.

**Nicht-XML-Nutzdaten:** Wenn es sich nicht um XML-Nutzdaten handelt, verfügt der Nutzdatencontainer über die folgenden Eigenschaften:

Tabelle 32. Objekt für Nutzdatencontainer für Nicht-XML-Nutzdaten

Attribut	Typ
contentType	xsd:string
encoding	xsd:string
role	xsd:string
schema	xsd:string
schemaVersion	xsd:string
description	xsd:string
dataBytes	xsd:base64Binary
dataString	xsd:string

Die Nutzdaten sind der Wert von 'dataBytes' oder 'dataString'.

- Wenn die Nutzdaten keine XML-Daten sind, wird bei der Konvertierung der Paketnachricht der Back-End-Integration in ein Objekt der höchsten Ebene der Wert des Elements <payload> in der Nachricht von der Datenbindung des Back-End-Integrationspakets nicht in das Geschäftsobjekt für Nutzdaten konvertiert. Stattdessen wird der Wert des Attributs 'dataBytes' (oder 'dataString') des Geschäftsobjekts für Nutzdatencontainer auf den Byte- oder Zeichenfolgewert der Nutzdaten gesetzt.
- Gleichermaßen wird bei der Erstellung der Paketnachricht der Back-End-Integration aus einem Objekt der höchsten Ebene der Inhalt des Attributs 'dataBytes' (oder 'dataString') des Geschäftsobjekts für Nutzdatencontainer von der Datenbindung des Back-End-Integrationspakets als Wert für das Element <payload> der Paketnachricht der Back-End-Integration festgelegt.

Die Attribute role, schema, SchemaVersion und description werden im WBI-Paket 1.2 hinzugefügt. Diese Attribute werden nur im ebXML-Fluss verwendet. Der Wert dieser Attribute für bestimmte Nutzdaten (payload) wird als Wert der entsprechenden Attribute in der ebXML-Soap-Nachricht für diese Nutzdaten festgelegt.

## Geschäftsobjekt für Anhangscontainer

Dieses Geschäftsobjekt stellt einen Anhang dar. Es kann sich um einen beliebigen komplexen Typ handeln, der aber die folgenden Attribute aufweisen muss:

*Tabelle 33. Geschäftsobjekt für Anhangscontainer - Attribute*

Attribut	Typ
contentID	xsd:string
contentType	xsd:string
encoding	xsd:string
role	xsd:string
schema	xsd:string
schemaVersion	xsd:string
description	xsd:string
dataBytes	xsd:base64Binary
dataString	xsd:string

Der Anhang ist der Wert von 'dataBytes' oder 'dataString'.

## Geschäftsobjekt für Paketheader

Dieses Geschäftsobjekt enthält die Header der Transportebene. Die x-aux-Transportheader werden im Abschnitt „Inhalt des Headers der Transportebene“ auf Seite 26 beschrieben. Dieses Geschäftsobjekt muss die folgenden Attribute aufweisen.

*Tabelle 34. Geschäftsobjekt für Paketheader - Attribute*

Attribut	Typ
x-aux-sender-id	string
x-aux-receiver-id	string
x-aux-protocol	string
x-aux-protocol-version	string
x-aux-process-type	string
x-aux-process-version	string
x-aux-create-datetime	string
x-aux-msg-id	string
x-aux-production	string
x-aux-system-msg-id	string
x-aux-payload-root-tag	string
x-aux-process-instance-id	string
x-aux-event-status-code	string
x-aux-third-party-bus-id	string
x-aux-transport-retry-count	string
x-aux-SyncResponse	string
x-aux-IntelligibleCheckRequired	string
x-aux-TimeToAccept	string
x-aux-role	string
x-aux-role-schema	string
x-aux-role-schema-version	string

Tabelle 34. Geschäftsobjekt für Paketheader - Attribute (Forts.)

Attribut	Typ
x-aux-description	string
content-type	string
content-length	string

## Geschäftsobjekte für RosettaNet Service Content erstellen

Die Datenbindung des Back-End-Integrationspakets erfordert ein Objekt der höchsten Ebene. Das Objekt der höchsten Ebene kann wie im Abschnitt „Geschäftsobjekt der höchsten Ebene“ auf Seite 68 beschrieben erstellt werden. WebSphere Partner Gateway erwartet, dass die RosettaNet Service Content-XML den Nutzdaten der Paketnachricht der Back-End-Integration entspricht. Zur Erstellung des Objekts für Nutzdatencontainer benötigen Sie ein Geschäftsobjekt für Nutzdaten, das die RosettaNet Service Content-XML darstellt. Zur Erstellung des Geschäftsobjekts für RosettaNet Service Content können Sie eine der folgenden Methoden verwenden:

- Wenn die Struktur des von RosettaNet bereitgestellten PIP-Service-Content im XML-Schemaformat vorhanden ist, kann dieses XML-Schema als Geschäftsobjekt für Nutzdaten verwendet werden.
- Wenn die Struktur des von RosettaNet bereitgestellten PIP-Service-Content im DTD-Format (DTD - Document Type Definition) vorhanden ist, muss es in das XML-Schema konvertiert werden. Befolgen Sie zum Konvertieren der PIP-Service-Content-DTD in das XML-Schema die Anweisungen im Handbuch *WebSphere Partner Gateway Hubkonfiguration*. Dieses XML-Schema kann als Geschäftsobjekt für Nutzdaten verwendet werden.
- Wenn der PIP das DTD-Format aufweist und für den PIP ein von WebSphere Partner Gateway bereitgestelltes PIP-Paket vorhanden ist, können Sie alternativ die folgende Prozedur befolgen, um das Geschäftsobjekt für den PIP-Service-Content zu erstellen.
  1. Navigieren Sie zu dem RosettaNet Service Content-PIP-Paket für die entsprechende RNIF-Version. Wenn Sie zum Beispiel das Geschäftsobjekt PIP3A4PurchaseOrderRequest erstellen möchten, können Sie die Datei BCG\_Package\_RNSC1.0\_RNIFV02.02.zip im Produktverzeichnis von WebSphere Partner Gateway verwenden.
  2. Importieren Sie die komprimierte Datei mit WebSphere Integration Developer in die WebSphere Process Server-Module, die Sie für den PIP konfiguriert haben.
  3. Erweitern Sie den Ordner für die Datentypen, und klicken Sie mit der rechten Maustaste auf das Geschäftsobjekt, das dem Root-Element des Service-Content entspricht. Klicken Sie im Falle einer PIP-3A4-Anforderung im Ordner für die Datentypen mit der rechten Maustaste auf Pip3A4PurchaseOrder, und wählen Sie dann die Option zum Öffnen mit dem Texteditor aus.
  4. Ändern Sie im Texteditor die Anzeigeeoptionen wie folgt:
 

```

          ../../common/

          in
          ./
          
```
  5. Speichern Sie die Änderungen, und schließen Sie den Texteditor.

## Funktionsweise der Datenbindung des Back-End-Integrationspakets

In diesem Abschnitt wird beschrieben, wie die Datenbindung des Back-End-Integrationspakets eine Paketnachricht der Back-End-Integration aus einem Geschäftsobjekt bzw. ein Geschäftsobjekt aus einer eingehenden Nachricht erstellt.

### Paketnachricht der Back-End-Integration in Objekt der höchsten Ebene konvertieren

In diesem Abschnitt wird beschrieben, wie die Paketnachricht der Back-End-Integration in ein Geschäftsobjekt konvertiert wird, das von den Services in WebSphere Process Server verwendet werden kann.

Wie in Abschnitt „Geschäftsobjekt der höchsten Ebene und untergeordnete Geschäftsobjekte“ auf Seite 68 beschrieben, funktioniert die Datenbindung des Back-End-Integrationspakets mit einem bestimmten Typ von Geschäftsobjekt der höchsten Ebene. Sie müssen dieses Geschäftsobjekt erstellen, bevor Sie die Datenbindung verwenden können.

Das Geschäftsobjekt der höchsten Ebene wird im Abschnitt „Geschäftsobjekt der höchsten Ebene“ auf Seite 68 beschrieben.

**Nachrichtentypen:** Die Art und Weise, wie die Paketnachricht der Back-End-Integration von WebSphere Partner Gateway konvertiert wird, hängt vom Typ der Nachricht ab. Es gibt drei Typen von Nachrichten:

- Eine XML-Nachricht mit dem Root-Tag <transport-envelope> und der URI „http://www.ibm.com/websphere/bcg/2003/v1.1/wbipackaging“ oder „http://www.ibm.com/websphere/bcg/2003/v1.0/wbipackaging“.

Beispiel für eine Nachricht innerhalb eines Transportumschlags (transport-envelope):

```
<transport-envelope
  xmlns="http://www.ibm.com/websphere/bcg/2003/v1.1/wbipackaging">
  <payload encoding="base64" contentType="application/xml"
contentId="111111111111">
    ...base64 encoded XML message...
  </payload>
  <attachment encoding="base64" contentType="text/xml"
contentId="222222222222">
    ...base64 encoded XML attachment...
  </attachment>
  <attachment encoding="base64" contentType="application/pdf"
contentId="333333333333">
    ...base64 encoded PDF attachment...
  </attachment>
</transport-envelope>
```

- Andere XML-Nachrichten.
- Nicht-XML-Nachrichten.

**Methoden zum Konvertieren der Nachricht:** Zum Konvertieren einer JMS-Nachricht des Back-End-Integrationspakets in ein Objekt der höchsten Ebene müssen Sie die Methode 'read' von BCGBackEndIntegrationJMSDataBindingImpl verwenden.

Zum Konvertieren einer Paketnachricht der Back-End-Integration in ein Objekt der höchsten Ebene können Sie die folgenden Methoden von BCGBackEndIntegrationDataBindingUtil verwenden:

- read():

Wenn die Paketnachricht der Back-End-Integration in einem Eingabedatenstrom enthalten ist, können Sie diese Methode verwenden.

- `setFromArray`:

Wenn die Paketnachricht der Back-End-Integration in einer Bytefeldgruppe enthalten ist, können Sie diese Methode verwenden.

- `setFromString`

Wenn die Paketnachricht der Back-End-Integration in einer Zeichenfolge enthalten ist, können Sie diese Methode verwenden.

Bevor Sie eine dieser Methoden aufrufen, können Sie die Verarbeitung der Datenbindung durch Aufruf der folgenden Methoden anpassen:

- `setTLOTypeName()`

Verwenden Sie `setTLOTypeName` für das Objekt, um es zu benennen und die zugehörige URI anzugeben.

- `setBOPrefix()`

Verwenden Sie `setBOPrefix`, um das Präfix für das Geschäftsobjekt anzugeben. Wenn Sie kein Präfix angeben, wird das Standardpräfix `TLO_` verwendet.

- Kann der Typ des Geschäftsobjekts für Nutzdaten ermittelt werden, lautet das Objekt der höchsten Ebene `<BOPrefix><PayloadBOPrefixName>`.
- Kann das Geschäftsobjekt für Nutzdaten nicht ermittelt werden (für Nicht-XML-Nutzdaten), lautet der Name `BCG_TLO_BackendPackaging` (Standardname).

Wenn Sie `setTLOTypeName` und `setBOPrefix` angeben, wird `setBOPrefix` ignoriert.

Für Nicht-XML-Nutzdaten: Wenn Sie den Namen des Objekts der höchsten Ebene durch Aufruf von `setTLOTypeName` nicht angeben, bevor Sie die Methode 'read' aufrufen, wird das Standardobjekt verwendet. Die Nutzdaten werden nicht in ein Geschäftsobjekt für Nutzdaten konvertiert. Stattdessen werden die Datenbyte im Standardobjekt der höchsten Ebene unverändert gesetzt. In Abb. 19 auf Seite 75 wird das Standardobjekt der höchsten Ebene dargestellt.

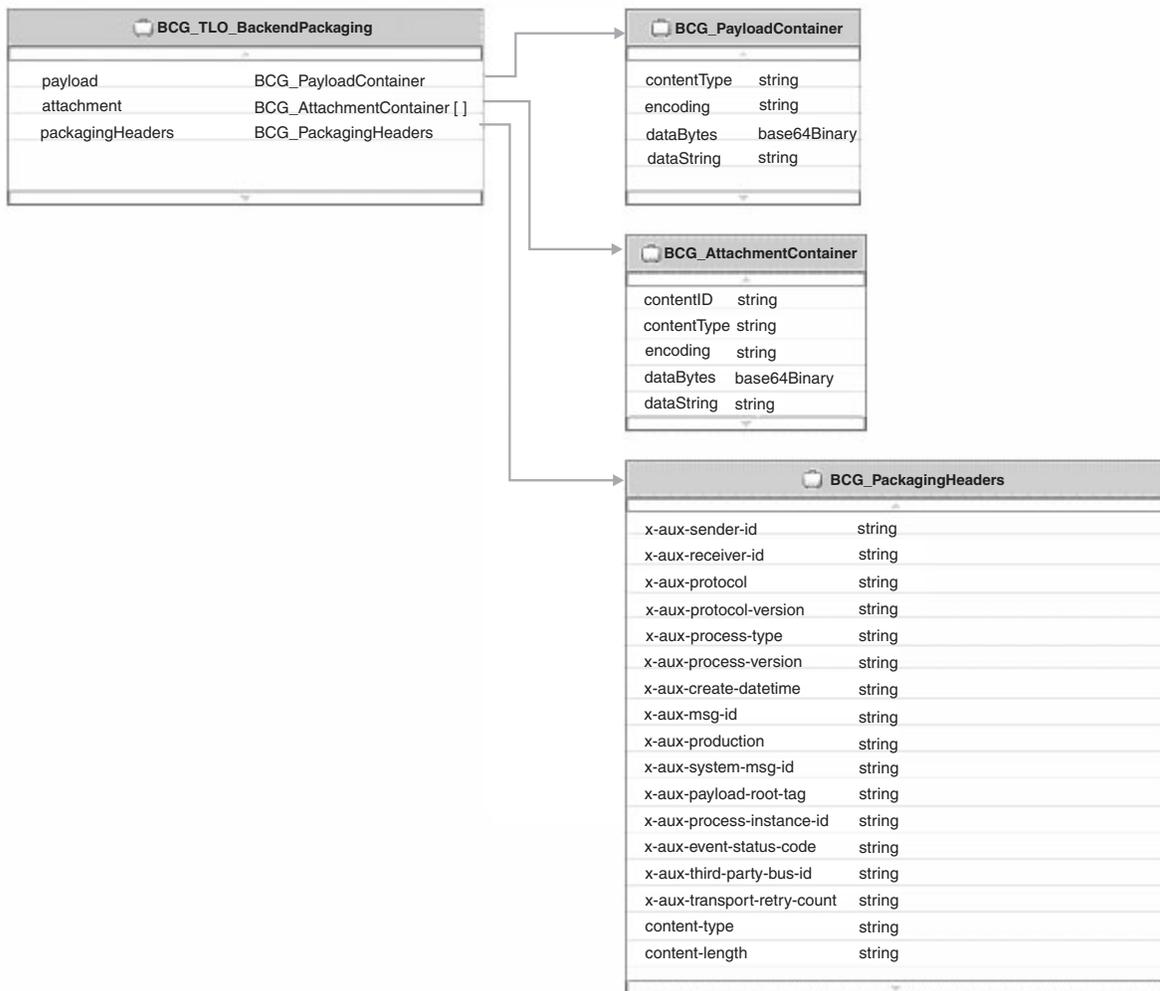


Abbildung 19. Standardobjekt der höchsten Ebene

Wenn die Nachricht über Transportheader verfügt, müssen diese gelesen und mit `setxAuxHeaders()` gesetzt werden. Die Methode `setxAuxHeaders` muss vor der Methode `read()` aufgerufen werden. Die Methode `setxAuxHeaders` wird nicht auf die Datenbindung `BCGBackEndIntegrationJMSDataBindingImpl` angewendet, da die JMS-Transportheader aus der jeweiligen JMS-Nachricht in der Methode 'read' gelesen werden.

**Nachrichten konvertieren:** In den folgenden Abschnitten wird die Funktionsweise der Methoden `read`, `setFromByteArray` und `setFromString` von `BCGBackEndIntegrationDataBindingUtil` sowie der Methode 'read' von `BCGBackEndIntegrationJMSDataBindingImpl` beschrieben. Funktionsweise der Methode:

1. Der XML-Inhaltstyp wird aus dem Header 'Content-Type' (sofern in der Nachricht vorhanden) ermittelt. Beispiel:

Content-Type: application/xml

Wenn der Inhaltstyp nicht verfügbar ist, überprüft die Methode die ersten Byte der Nachricht, um den Typ der Nachricht festzustellen.

Die Methode ermittelt die Nutzdaten (payload) auf der Basis des Inhaltstyps wie folgt:

- Bei einer XML-Nachricht wird die Nachricht entserialisiert, um das Geschäftsobjekt für Nutzdaten zu ermitteln.
- Bei einer XML-Nachricht mit dem Root-Tag <transport-envelope> werden die Nutzdaten mit Base64 decodiert, um die Nutzdatenbyte zu ermitteln; dann wird die Nachricht geparkt, um festzustellen, ob es sich um eine XML- oder eine Nicht-XML-Nachricht handelt.
  - Im Falle einer XML-Nachricht werden die Nutzdatenbyte entserialisiert, um das Geschäftsobjekt für Nutzdaten zu ermitteln.
  - Im Falle einer Nicht-XML-Nachricht werden die Nutzdatenbyte nicht entserialisiert. Die Nutzdatenbyte werden unverändert verwendet.

Jeder im Transportumschlag enthaltene Anhang wird mit Base64 decodiert, um die Anhangsbyte zu ermitteln. Die Anhangsbyte werden unverändert verwendet und nicht entserialisiert.

- Bei einer XML-Nachricht mit dem Root-Tag <EventNotification> wird die Nachricht entserialisiert, um das Geschäftsobjekt für Ereignisbenachrichtigung zu ermitteln.
- Im Falle einer Nicht-XML-Nachricht wird die Nachricht nicht entserialisiert. Die Byte werden unverändert verwendet.

2. Der Name des Geschäftsobjekts der höchsten Ebene wird ermittelt.

- Bei XML-Nutzdaten:
  - Wenn setTLOTypeName aufgerufen wurde, bevor ein Aufruf von read, setFromArray oder setFromString erfolgt ist, wird der in der Methode angegebene Name direkt zur Erstellung des Namens der höchsten Ebene verwendet.
  - Wenn Sie setBOPrefix verwenden, wird der Name für das Objekt der höchsten Ebene aus dem angegebenen Präfix und dem XML-Root-Element der Nutzdaten zusammengesetzt. Die URI wird ebenfalls aus den Nutzdaten ermittelt.
  - Wenn Sie setTLOTypeName oder setBOPrefix nicht verwenden, wird der Name für das Objekt der höchsten Ebene aus dem Standardpräfix TLO\_ und dem XML-Root-Element der Nutzdaten zusammengesetzt.

- Bei Nicht-XML-Nutzdaten:
  - Das Standardobjekt der höchsten Ebene kann verwendet werden.
  - Die Methode setTLOTypeName kann verwendet werden, sie muss jedoch mit dem von Ihnen erstellten Objekt der höchsten Ebene bzw. mit dem Standardobjekt der höchsten Ebene (BCG\_TLO\_BackendPackaging) konform sein.

3. Das Objekt der höchsten Ebene wird instanziiert.

4. Das Geschäftsobjekt für Nutzdatencontainer wird instanziiert, und die zugehörigen Werte werden festgelegt. Die Art und Weise, wie die Werte festgelegt werden, hängt vom Typ der Nachricht ab.

- Bei einer XML-Nachricht mit dem Root-Tag <transport-envelope> werden die Attribute wie folgt gesetzt:
  - contentType  
Der Wert des Attributs contentType für den Tag <payload>.
  - encoding  
Der Wert des Attributs encoding für den Tag <payload>.
  - payload

- Der Name des Geschäftsobjekt für Nutzdaten.
- Bei einer XML-Nachricht bzw. einer XML-Nachricht mit dem Root-Tag <event-notification> werden die Attribute wie folgt gesetzt:
  - contentType wird nicht gesetzt.
  - encoding wird nicht gesetzt.
  - payload
    - Der Name des Geschäftsobjekt für Nutzdaten.
- Bei einer Nicht-XML-Nachricht werden die Attribute wie folgt gesetzt:
  - contentType wird nicht gesetzt.
  - encoding wird nicht gesetzt.
  - dataBytes
    - Wenn die eingehende Nachricht in Byte empfangen wird, wird die Gesamtmenge an Byte als Wert für dieses Attribut gesetzt.
  - dataString
    - Wenn die eingehende Nachricht als Zeichenfolge empfangen wird, wird die gesamte Zeichenfolge als Wert für dieses Attribut gesetzt.
  - payload wird nicht gesetzt.
- 5. Die Anhangsgeschäftsobjekte werden instanziiert und die zugehörigen Werte wie folgt festgelegt:
  - contentType
    - Der Wert des Attributs contentType für den Tag <attachment>.
  - encoding
    - Der Wert des Attributs encoding für den Tag <attachment>.
  - contentId
    - Der Wert des Attributs contentId für den Tag <attachment>.
  - dataBytes
    - Die mit Base64 decodierten Anhangsbyte.
- 6. Das Geschäftsobjekt für die Transportheader wird instanziiert, und die aus der Nachricht gelesenen Transportheader werden gesetzt. Der Name für dieses Geschäftsobjekt wird als Wert des Attributs packagingHeaders im Objekt der höchsten Ebene festgelegt.

Mustercode zeigt, wie die Methoden verwendet werden.

### **Objekt der höchsten Ebene in Paketnachricht der Back-End-Integration konvertieren**

In diesem Abschnitt wird beschrieben, wie ein Geschäftsobjekt von WebSphere Process Server in eine Paketnachricht der Back-End-Integration serialisiert wird. Die Datenbindung stellt den Inhalt sowie eventuelle Anhänge in einen Transportumschlag im Base64-Format. Die Transportheader aus dem Objekt der höchsten Ebene werden als Zeichenfolge in den Umschlag eingefügt.

Bevor die Datenbindung des Back-End-Integrationspakets aufgerufen werden kann, muss ein Objekt der höchsten Ebene von einem Service erstellt werden (siehe „Geschäftsobjekt der höchsten Ebene“ auf Seite 68).

**Methoden zum Konvertieren des Objekts:** Ein WebSphere Process Server-Service ruft die Datenbindung des Back-End-Integrationspakets auf und sendet diese an das Objekt der höchsten Ebene. Zum Konvertieren des Objekts in eine JMS-Nachricht des Back-End-Integrationspakets verwenden Sie die Methode 'write' von

BCGBackEndIntegrationJMSDataBindingImpl. Zum Konvertieren des Objekts in eine Paketnachricht der Back-End-Integration verwenden Sie die Methode write, getAsByteArray oder getAsString von BCGBackEndIntegrationDataBindingUtil.

Vor Aufruf einer dieser Methoden können Sie die folgenden Methoden aufrufen:

- Methode setPackagingSchema  
Diese Methode gibt an, welches Paketschema bei der Erstellung der Paketnachricht der Back-End-Integration verwendet werden soll.
- Methode setOptions  
Diese Methode stellt Informationen (wie z. B. den Root-Tag der Nutzdaten) bereit, die zum Serialisieren der Nachricht verwendet werden können. Informationen zu den Optionen, die Sie festlegen können, finden Sie im Abschnitt „Methode setOptions“ auf Seite 83.

**Objekte konvertieren:** Im folgenden Abschnitt wird die Funktionsweise der Methoden write, getAsByteArray und getAsString von BCGBackEndIntegrationDataBindingUtil sowie der Methode 'write' von BCGBackEndIntegrationJMSDataBindingImpl beschrieben. Funktionsweise der Methode:

1. Der Inhaltstyp der Nutzdaten wird ermittelt. Diese Informationen werden aus dem Geschäftsobjekt für Nutzdatencontainer abgerufen. Die Art und Weise, wie die Nutzdaten verarbeitet werden, basiert auf dem zugehörigen Inhaltstyp wie folgt:
  - XML-Nutzdaten  
Das Attribut dataBytes bzw. das Attribut dataString des Nutzdatencontainers (sofern vorhanden) muss null sein, und das Geschäftsobjekt für Nutzdatencontainer sollte mindestens ein Attribut eines komplexen Typs aufweisen. Dies ist ein Verweis auf das Geschäftsobjekt für Nutzdaten. Das Geschäftsobjekt wird serialisiert und als XML-Nutzdaten verwendet. Wurden mehrere Attribute mit einem komplexen Typ definiert, stellt das erste dieser Attribute die XML-Nutzdaten dar.
  - EventNotification  
Das erste Attribut, das nicht null ist, sollte den Typ 'EventNotification' aufweisen. Dies ist ein Verweis auf das Geschäftsobjekt für Ereignisbenachrichtigungen. Das Geschäftsobjekt wird serialisiert und als XML-Nutzdaten verwendet.
  - Nicht-XML  
Wenn das Attribut dataBytes nicht null ist, wird der Wert des Attributs als Nutzdaten verwendet.  
Wenn das Attribut dataString nicht null ist, wird der Wert des Attributs als Nutzdaten verwendet.
2. Das Geschäftsobjekt für Nutzdaten wird serialisiert, wenn der Typ XML oder EventNotification ist.
3. Base64 codiert die Nutzdaten basierend auf dem zugehörigen Typ:  
Wenn das Geschäftsobjekt für Nutzdatencontainer für das Attribut dataByte einen Wert aufweist, der nicht null ist, wird dieser Wert (Nutzdatenbyte) mit Base64 codiert.  
Wenn das Geschäftsobjekt für Nutzdatencontainer für das Attribut dataString einen Wert aufweist, der nicht null ist, werden die Byte basierend auf dem Attribut contentType des Geschäftsobjekts extrahiert. Die Byte werden dann mit Base64 codiert.

Wenn die Nutzdaten aus einem Geschäftsobjekt serialisiert wurden, werden die Byte basierend auf dem Attribut contentType des Geschäftsobjekts extrahiert. Die Byte werden dann mit Base64 codiert.

4. Ein XML-Dokument wird basierend auf dem angegebenen Paketschema erstellt.
  - a. Der Root-Tag <transport-envelope> wird hinzugefügt.
  - b. Der Tag <payload>, der ein untergeordnetes Element des Tags <transport-envelope> ist, wird mit der im Base64-Format codierten Zeichenfolge aus Schritt 3 auf Seite 78 gesetzt.
5. Die Anhänge werden verarbeitet.

Wenn das Attribut dataByte des Objekts für Anhangscontainer nicht null ist, werden die Byte mit Base64 codiert, und die Zeichenfolge wird als Wert des Tags <attachment> festgelegt.

Wenn das Objekt für Anhangscontainer für das Attribut dataString einen Wert aufweist, der nicht null ist, werden die Byte basierend auf dem Attribut contentType des Geschäftsobjekts extrahiert. Die Byte werden dann mit Base64 codiert, und die Zeichenfolge wird als Wert des Tags <attachment> festgelegt.

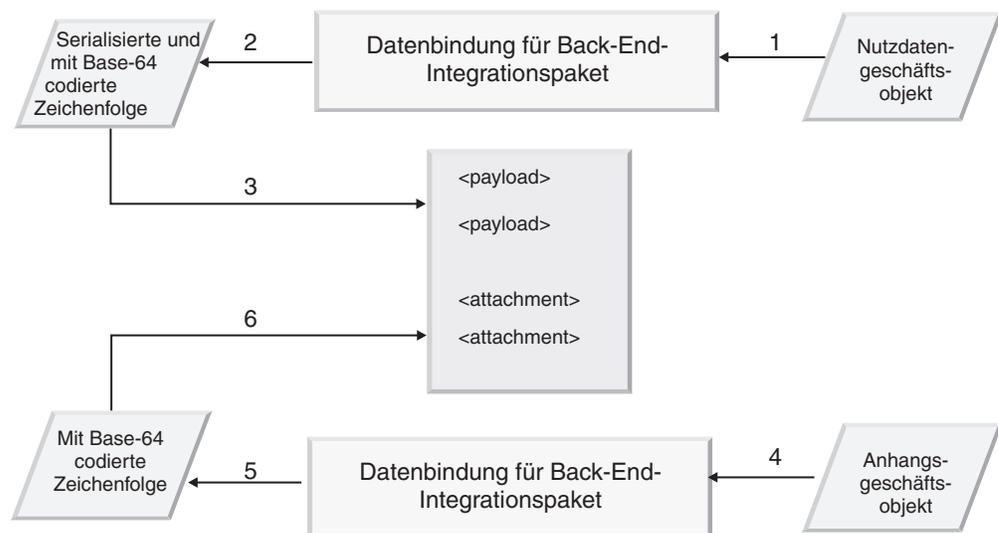


Abbildung 20. Wie die Datenbindung konvertierte Geschäftsobjekte der Paketschicht der Back-End-Integration hinzufügt

Mustercode zeigt, wie die Methoden verwendet werden.

## Klasse BCGBackEndIntegrationDataBindingUtil

In diesem Abschnitt werden die Klasse BCGBackEndIntegrationDataBindingUtil sowie die Methoden dieser Klasse beschrieben. Sie können diese Klasse unverändert verwenden oder eine neue Datenbindungsklasse erstellen, wenn Sie die Verarbeitung der Datenbindung anpassen möchten.

### DataBinding-Schnittstelle

BCGBackEndIntegrationDataBindingUtil implementiert die folgende DataBinding-Schnittstelle:

```
DataBinding
public interface comonj.connector.runtime.DataBinding extends Serializable {
    public DataObject getDataObject()throws
```

```

commonj.connector.runtime.DataBindingException;
    public void setDataObject(DataObject dataObject) throws
commonj.connector.runtime.DataBindingException;
}

```

## Methoden

In diesem Abschnitt werden die Methoden von `BCGBackEndIntegrationDataBindingUtil` beschrieben.

Die Klasse `BCGBackEndIntegrationDataBindingUtil` verfügt über zwei Typen von Methoden:

- Methoden, die ein Datenobjekt aus einer Bytefeldgruppe, einer Zeichenfolge oder einem Datenstrom erstellen:
  - `read(InputStream)`
  - `setFromArray(byte[ ])`
  - `setFromString(String)`
  - `setxAuxHeaders(HashMap)`
  - `setBOPrefix(String)`
  - `setTLOTypeName(String, String)`
  - `getDataObject()`

### Anmerkungen:

1. Zur Erstellung des Datenobjekts kann eine der folgenden Methoden verwendet werden:

- `read(InputStream)`
- `setFromArray(byte[ ])`
- `setFromString(String)`

Vor Aufruf der Methode zur Erstellung des Datenobjekts kann das Programm eine der folgenden Methoden aufrufen:

- `setxAuxHeaders`
- `setBOPrefix`
- `setTLOTypeName`

2. Bei Verwendung der Methode `setTLOTypeName` ist die Methode `setBOPrefix` bedeutungslos und wird ignoriert.
3. Verwenden Sie die Methode `getDataObject`, um das Geschäftsobjekt der höchsten Ebene zu ermitteln.

- Methoden, die ein Datenobjekt in eine Bytefeldgruppe, eine Zeichenfolge oder einen Datenstrom konvertieren:
  - `setDataObject(DataObject)`
  - `setPackagingSchema(String)`
  - `setOptions(HashMap)`
  - `getAsString()`
  - `getAsByteArray()`
  - `write(OutputStream)`
  - `getxAuxHeaders()`

### Methode `getAsByteArray`

Diese Methode gibt die aus dem Objekt der höchsten Ebene ermittelte Bytefeldgruppe (`byte[ ]`) zurück.

**Syntax:**

```
public byte[ ] getAsByteArray()
```

**Methode getAsString**

Diese Methode gibt ein Zeichenfolgeformat des Objekts der höchsten Ebene zurück.

**Syntax:**

```
protected java.lang.String getAsString()
```

**Methode getDataObject**

Diese Methode gibt das Datenobjekt zurück, sofern es bereits instanziiert wurde.

**Syntax:**

```
public commonj.sdo.DataObject getDataObject()
```

**Methode getXAuxHeader**

Diese Methode gibt die x-aux-Header aus dem Transportumschlag zurück. Informationen zu x-aux-Headern finden Sie im Abschnitt „Inhalt des Headers der Transportebene“ auf Seite 26.

**Syntax:**

```
public java.util.HashMap getXAuxHeaders()
```

**Methode read**

Diese Methode greift auf einen Eingabedatenstrom zu, liest diesen Eingabedatenstrom und konvertiert ihn in ein Geschäftsobjekt der höchsten Ebene. Eine detaillierte Beschreibung der Aktionen, die von der Methode 'read' ausgeführt werden, finden Sie im Abschnitt „Paketnachricht der Back-End-Integration in Objekt der höchsten Ebene konvertieren“ auf Seite 73.

**Syntax:**

```
public void read(java.io.InputStream inputStream)
    throws commonj.connector.runtime.DataBindingException
```

**Parameter:****inputStream**

Der Eingabedatenstrom, aus dem die Daten gelesen werden.

**Methode setBOPrefix**

Wenn das Objekt der höchsten Ebene nicht angegeben wurde, können Sie ein Präfix angeben, indem Sie setBOPrefix () aufrufen. Wenn Sie kein Präfix angeben, wird TLO\_ als Standardpräfix verwendet.

**Anmerkungen:**

1. Im Falle von XML-Nutzdaten wird der Namespace der XML-Nutzdaten als Namespace des Objekts der höchsten Ebene verwendet.
2. Im Falle von Nicht-XML-Nutzdaten wird der Standardnamespace "http://ibm.com/websphere/bcg/2005/wbi/bo" als Namespace für das Objekt der höchsten Ebene verwendet.

**Syntax:**

```
public void setBOPrefix(java.lang.String prefix)
```

## Methode setDataObject

Diese Methode legt das Datenobjekt fest.

### Syntax:

```
public void setDataObject(commonj.sdo.DataObject topLevelbo)
    throws commonj.connector.runtime.DataBindingException
```

### Parameter:

#### topLevelbo

Das Geschäftsobjekt der höchsten Ebene. Informationen hierzu finden Sie im Abschnitt „Geschäftsobjekt der höchsten Ebene“ auf Seite 68.

## Methode setDebugLevel

Die Debugstufe ist standardmäßig auf 'error' (Fehler) gesetzt. Verwenden Sie diese Methode, um die Stufe zu ändern.

### Syntax:

```
public static void setDebugLevel(int debugLevel)
```

### Parameter:

#### debugLevel

Gültige Werte für die Debugstufe:

- BCG\_LOG\_DEBUG
- BCG\_LOG\_ERROR
- BCG\_LOG\_WARNING
- BCG\_LOG\_INFO

## Methode setFromArray

Diese Methode greift auf die Datenbyte der Nutzdaten (payload) oder des Transportumschlags (transport-envelope) zu und erstellt ein Objekt der höchsten Ebene. Vor Aufruf dieser Methode können Sie eine oder mehrere der folgenden Methoden aufrufen:

- Methode setTLOTypeName
- Methode setBOPrefix
- Methode setxAuxHeader

### Syntax:

```
public void setFromArray(byte[ ] dataBytes)
    throws commonj.connector.runtime.DataBindingException
```

### Parameter:

#### dataBytes

Die Nutzdaten in Form von Byte.

## Methode setFromString

Diese Methode nimmt eine Zeichenfolge als Parameter und konvertiert diese in ein Objekt der höchsten Ebene. Vor Aufruf dieser Methode können Sie eine oder mehrere der folgenden Methoden aufrufen:

- Methode setTLOTypeName
- Methode setBOPrefix
- Methode setOptions

**Syntax:**

```
public void setFromString(java.lang.String fromString)
    throws commonj.connector.runtime.DataBindingException
```

**Parameter:****fromString**

Das Zeichenfolgeformat des Datenobjekts der höchsten Ebene oder der Nutzdaten.

**Methode setOptions**

Diese Methode legt die erforderlichen Optionen fest.

**Syntax:**

```
public void setOptions(java.util.HashMap options)
    throws commonj.connector.runtime.DataBindingException
```

**Parameter:****Hashmap**

Hashmap kann die folgenden Schlüsselwerte aufweisen:

- BCG\_ROOT\_NODE\_NAME

Der Rootname, der zum Generieren der Nutzdaten verwendet wird.

- BCG\_APPEND\_DTD

Dieses Attribut gibt an, ob ein DTD-Tag an die Nutzdaten angehängt werden soll. Der Wert kann BCG\_APPEND\_DTD\_TRUE oder BCG\_APPEND\_DTD\_FALSE sein. Der Standardwert ist BCG\_APPEND\_DTD\_FALSE.

- BCG\_SYSTEM\_ID

Die System-ID für die DTD. Wenn der Wert von BCG\_APPEND\_DTD auf BCG\_APPEND\_DTD\_TRUE gesetzt ist, kann dieser Wert festgelegt werden. Wenn der Wert von BCG\_APPEND\_DTD auf BCG\_APPEND\_DTD\_FALSE gesetzt ist, wird dieser Wert ignoriert.

- BCG\_PUBLIC\_ID

Die öffentliche ID für die DTD. Wenn der Wert von BCG\_APPEND\_DTD auf BCG\_APPEND\_DTD\_TRUE gesetzt ist, ist dieser Wert optional. Wenn der Wert von BCG\_APPEND\_DTD auf BCG\_APPEND\_DTD\_FALSE gesetzt ist, wird diese Option ignoriert.

**Methode setPackagingSchema**

Diese Methode definiert das Paketschema, das beim Generieren des Back-End-Integrationspakets verwendet wird. Sie können eines der folgenden Paketschemata angeben:

- <http://www.ibm.com/websphere/bcg/2003/v1.0/wbipackaging>
- <http://www.ibm.com/websphere/bcg/2003/v1.1/wbipackaging>
- <http://www.ibm.com/websphere/bcg/2004/v1.2/wbipackaging>

Wenn das Paketschema nicht angegeben wird, wird <http://www.ibm.com/websphere/bcg/2003/v1.1/wbipackaging> verwendet.

Das Paketschema wird beim Aufrufen der Methode `write` verwendet.

**Syntax:**

```
public void setPackagingSchema(java.lang.String packagingSchema)
```

**Parameter:****packagingSchema**

Der Schemaname für das Geschäftsobjekt der höchsten Ebene.

**Methode setTLOTypeName**

Diese Methode legt den Namen des Objekts der höchsten Ebene und den zugehörigen Namespace fest.

- Wenn es sich um Nicht-XML-Nutzdaten handelt und setTLOTypeName nicht verwendet wird, wird der Name des Standardobjekts der höchsten Ebene verwendet.
- Wenn die Nutzdaten ein XML-Dokument sind, wird der Name des Objekts der höchsten Ebene wie folgt abgeleitet:

`boPrefix + < Root-Tag der XML-Nutzdaten >`.

Wenn beispielsweise setBOPrefix("IBM") aufgerufen wird und das Root-Element der Nutzdaten "ABC" ist, lautet der Name des Objekts der höchsten Ebene "IBM\_ABC".

Das Präfix kann mit Hilfe der Methode setBOPrefix festgelegt werden. Wenn der Name des Objekts der höchsten Ebene nicht festgelegt wurde und die Nutzdaten ein XML-Dokument sind, muss der Datentyp (DataType) wie folgt aussehen:

`boPrefix + < Root-Tag >`.

**Syntax:**

```
public void setTLOTypeName(java.lang.String tns,  
                           java.lang.String typeName)
```

**Parameter:****tns**

Der Namespace des Geschäftsobjekts der höchsten Ebene.

**typeName**

Der Typ des Objekts der höchsten Ebene.

**Methode setxAuxHeader**

Diese Methode legt die Transportheader fest. Wenn keine Transportheader festgelegt wurden, müssen keine Transportheader gelesen werden. Informationen zu x-aux-Headern finden Sie im Abschnitt „Inhalt des Headers der Transportebene“ auf Seite 26.

**Syntax:**

```
public void setxAuxHeaders(java.util.HashMap xAuxHeaders)
```

**Methode write**

Diese Methode schreibt das Objekt der höchsten Ebene in den angegebenen Ausgabedatenstrom. Eine detaillierte Beschreibung der Aktionen, die von der Methode 'write' ausgeführt werden, finden Sie im Abschnitt „Objekt der höchsten Ebene in Paketnachricht der Back-End-Integration konvertieren“ auf Seite 77.

**Syntax:**

```
public void write(java.io.OutputStream outputStream)  
                throws commonj.connector.runtime.DataBindingException
```

**Parameter:****outputStream**

Der Ausgabedatenstrom, in den das Datenobjekt geschrieben wird.

## Mustercode

Der folgende Code stellt dar, wie aus einem Eingabedatenstrom ein Objekt der höchsten Ebene erstellt werden kann:

```
BCGBackEndIntegrationDataBindingUtil util = BCGBackEndIntegrationDataBindingUtil ();
util.setTLOTypeName ("TLO_URIName", "TLOName");
//util.setBOPrefix ("BO_prefix");
//setBoPrefix is commented because setTLOTypeName () is being used
util.read (inputStream);
DataObject tlo = util.getDataObject ();
```

Der folgende Code stellt dar, wie ein Datenstrom aus einem Datenobjekt ermittelt werden kann:

```
BCGBackEndIntegrationDataBindingUtil util = BCGBackEndIntegrationDataBindingUtil ();
util.setOptions (options);
util.setDataObject (tlo);
byte [ ] tlo_bytes = util.getAsByteArray ();
```

---

## Klasse BCGBackEndIntegrationJMSDataBindingImpl

Diese Klasse erstellt ein Geschäftsobjekt aus einer JMS-Nachricht, die Nutzdaten enthält, oder schreibt das Geschäftsobjekt in eine JMS-Nachricht. Beim Lesen des Geschäftsobjekts aus einer JMS-Nachricht werden die JMS-Transportheader gelesen, bevor das Geschäftsobjekt erstellt wird. Beim Schreiben des Geschäftsobjekts in eine JMS-Nachricht schreibt diese Klasse die JMS-Transportheader, wenn das Geschäftsobjekt über ein untergeordnetes Geschäftsobjekt für Paketheader verfügt.

BCGBackEndIntegrationJMSDataBindingImpl erweitert BCGBackEndIntegrationDataBindingUtil (siehe Abschnitt „Klasse BCGBackEndIntegrationDataBindingUtil“ auf Seite 79).

## JMSDataBinding-Schnittstelle

BCGBackEndIntegrationJMSDataBindingImpl implementiert die JMSDataBinding-Schnittstelle:

```
public interface com.ibm.websphere.sca.jms.data.JMSDataBinding extends DataBinding {

    public void read(javax.jms.Message message) throws javax.jms.JMSEException;
    public void write(javax.jms.Message message) throws javax.jms.JMSEException;
    public int getMessageType();

    static public int OBJECT_MESSAGE = 0;
    static public int TEXT_MESSAGE = 1;
    static public int BYTES_MESSAGE = 2;
    static public int STREAM_MESSAGE = 3;
    static public int MAP_MESSAGE = 4;

}
```

## Methoden

In diesem Abschnitt werden die Methoden von BCGBackEndIntegrationJMSDataBindingImpl beschrieben.

### getMessageType

Diese Methode gibt den Typ der Nachricht zurück.

**Syntax:**

```
public int getMessageType()
```

## **isBusinessException**

Diese Methode gibt einen Hinweis darauf zurück, ob Geschäftsausnahmebedingungen (business exceptions) vorliegen.

### **Syntax:**

```
public boolean isBusinessException()
```

### **read**

Diese Methode liest die Transportheader und erstellt je nach Nachrichtentyp das Geschäftsobjekt der höchsten Ebene. Eine detaillierte Beschreibung der Aktionen, die von der Methode 'read' ausgeführt werden, finden Sie im Abschnitt „Paketnachricht der Back-End-Integration in Objekt der höchsten Ebene konvertieren“ auf Seite 73.

Bei dieser Methode ist es nicht erforderlich, aux-Header zu setzen und aus der Nachricht selbst zu lesen.

### **Syntax:**

```
public void read(javax.jms.Message jmsMessage)
    throws javax.jms.JMSEException
```

### **Parameter:**

#### **jmsMessage**

Die Nachricht mit den Nutzdaten und den Transportheadern.

## **setBusinessException**

Diese Methode definiert einen Anzeiger, der angibt, ob eine Geschäftsausnahmebedingung aufgetreten ist.

### **Syntax:**

```
public void setBusinessException(boolean arg0)
```

### **write**

Diese Methode schreibt das Datenobjekt in die Nachricht und legt je nach Nachrichtentyp die Header in der Nachricht fest. Eine detaillierte Beschreibung der Aktionen, die von der Methode 'write' ausgeführt werden, finden Sie im Abschnitt „Objekt der höchsten Ebene in Paketnachricht der Back-End-Integration konvertieren“ auf Seite 77.

### **Syntax:**

```
public void write(javax.jms.Message jmsMessage)
    throws javax.jms.JMSEException
```

---

## Datenbindungsklassen in Komponentenimplementierung einfügen

Bei der Entwicklung von WebSphere Process Server-Komponenten mit WebSphere Integration Developer können Sie die von WebSphere Partner Gateway bereitgestellten Klassen `BCGBackEndIntegrationJMSDataBindingImpl` und `BCGBackEndIntegrationDataBindingUtil` verwenden. Diese Klassen sind in der Datei `databinding.jar` enthalten, die sich im Verzeichnis `Integration\WBI\WebSphereProcessServer\DataBinding` auf dem Produktimage befindet.

Zur Verwendung dieser Klassen in Ihrem Geschäftsintegrationsprojekt müssen Sie sicherstellen, dass sich `databinding.jar` in der EAR-Datei des Geschäftsintegrationsprojekts befindet, das Sie in WebSphere Process Server implementieren. Lesen Sie die Informationen im Information Center für WebSphere Process Server oder WebSphere Integration Developer, oder befolgen Sie die in diesem Abschnitt beschriebene Prozedur, um die Datei `databinding.jar` in die generierte EAR-Datei einzufügen. Stellen Sie vor Ausführung dieser Prozedur sicher, dass Sie sich in der Geschäftsintegrationsperspektive (Business Integration Perspective) befinden und das Geschäftsintegrationsprojekt erstellt haben, in das Sie die `.jar`-Datei importieren möchten.

1. Importieren Sie in WebSphere Integration Developer die Datei `databinding.jar` in das Projekt. `databinding.jar` muss als J2EE-Dienstprogramm-JAR importiert werden. Nach einem erfolgreichen Import erstellt WebSphere Integration Developer implizit ein Projekt für `databinding.jar`.
2. Fügen Sie dieses Projekt Ihrem Geschäftsintegrationsprojekt als Abhängigkeit hinzu. Gehen Sie wie folgt vor, um das Projekt als Abhängigkeit hinzuzufügen:

**Anmerkung:** Die folgenden Anweisungen stellen nur eine Möglichkeit dar, wie Sie das Projekt als Abhängigkeit hinzufügen können. Informationen zu den anderen Möglichkeiten, wie Sie dieses Ziel erreichen können, finden Sie in der WebSphere Process Server-Dokumentation.

- a. Klicken Sie doppelt auf das Geschäftsintegrationsprojekt.  
Daraufhin wird der Abhängigkeitseditor geöffnet.
  - b. Erweitern Sie den Ordner **Java**.
  - c. Klicken Sie auf **Add**.  
Die Liste mit den Projekten wird angezeigt.
  - d. Wählen Sie das Projekt aus, das in Schritt 1 implizit von WebSphere Integration Developer für `databinding.jar` erstellt wurde.  
Das Projekt, das von WebSphere Integration Developer für `databinding.jar` erstellt wurde, wird als Abhängigkeit unter **Java** angezeigt.
  - e. Wählen Sie das Projekt aus.
  - f. Wählen Sie das Kontrollkästchen **Deploy with Module** aus.
3. Schließen Sie den Abhängigkeitseditor, und speichern Sie den Arbeitsbereich.  
Nach Ausführung dieser Schritte ist `databinding.jar` in der generierten EAR-Datei enthalten.



---

## Kapitel 4. Integration von WebSphere Process Server mit HTTP als Transportprotokoll

In diesem Kapitel wird die Integration von WebSphere Partner Gateway mit WebSphere Process Server über das HTTP-Transportprotokoll beschrieben. Dieses Kapitel enthält Informationen zu den folgenden Themen:

- „Nachrichten an WebSphere Process Server senden“
- „Nachrichten von WebSphere Process Server senden“ auf Seite 91
- „Dokumente an WebSphere Process Server senden“ auf Seite 93
- „Dokumente von WebSphere Process Server senden“ auf Seite 95

In diesem Kapitel wird nicht beschrieben, wie über HTTP gesendete SOAP-Anforderungen verarbeitet werden. Informationen zum Senden und Empfangen von SOAP-Anforderungen über HTTP finden Sie in Kapitel 6, „Integration von WebSphere Process Server mit SOAP/HTTP“, auf Seite 109.

---

### Nachrichten an WebSphere Process Server senden

In diesem Abschnitt wird beschrieben, wie WebSphere Partner Gateway eine Nachricht von einem externen Partner empfängt und die Nachricht über HTTP an WebSphere Process Server sendet.

#### Anmerkungen:

1. Alle Dokumenttypen außer RosettaNet können den Pakettyp 'None' aufweisen, wenn sie von WebSphere Partner Gateway an WebSphere Process Server gesendet werden. RosettaNet-Dokumente müssen den Pakettyp 'Backend Integration' aufweisen.
2. Alle Dokumenttypen außer SOAP (Web-Service-Anforderungen) können den Pakettyp 'Backend Integration' aufweisen, wenn sie von WebSphere Partner Gateway an WebSphere Process Server gesendet werden. Für SOAP-Anforderungen muss der Pakettyp 'None' angegeben sein.

Informationen zum Senden und Empfangen von SOAP-Dokumenten über HTTP finden Sie in Kapitel 6, „Integration von WebSphere Process Server mit SOAP/HTTP“, auf Seite 109.

Damit WebSphere Process Server eine von WebSphere Partner Gateway gesendete Nachricht empfangen kann, müssen Sie ein Servlet schreiben, um die Nachricht abzurufen und in ein Geschäftsobjekt zu konvertieren.

WebSphere Partner Gateway sendet Nachrichten an die URL-Adresse, die im HTTP-Ziel für den Empfänger des internen Partners in der Partnerverbindung konfiguriert ist. Das Servlet ist für diese URL-Adresse empfangsbereit und empfängt die Nachricht. Der Service, der das konvertierte Geschäftsdokument empfängt, verfügt über eine SCA-Exportbindung. Abb. 21 auf Seite 90 zeigt, wie eine von einem externen Partner gesendete Nachricht von WebSphere Partner Gateway verarbeitet und anschließend an das HTTP-Servlet gesendet wird, wo das Geschäftsobjekt über die Exportbindung den Service aufruft.

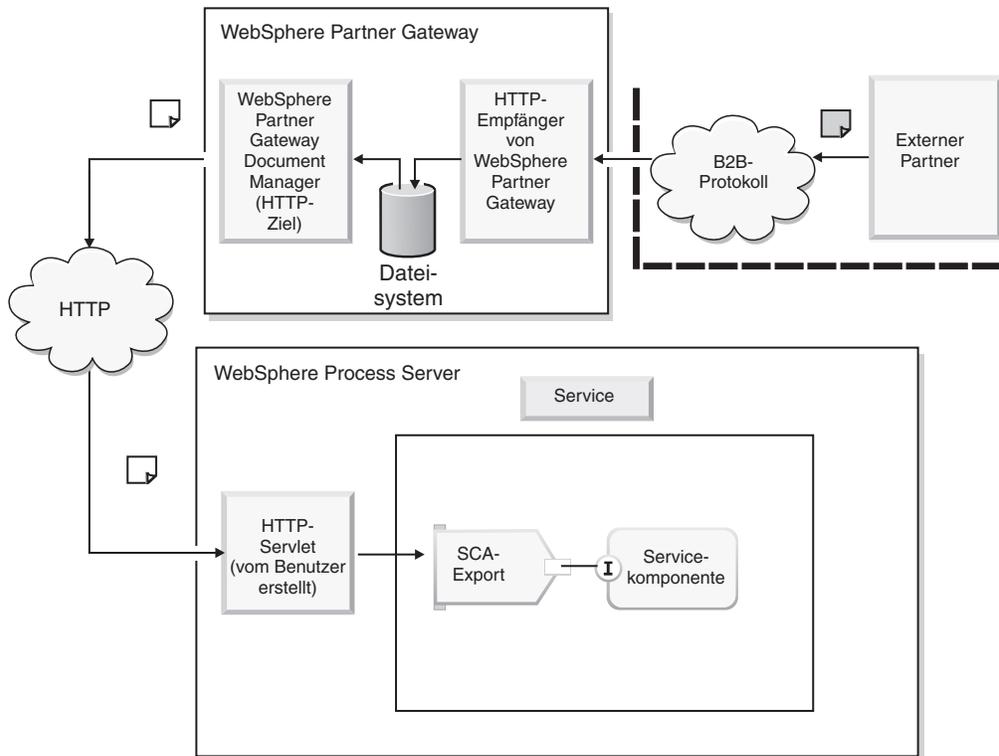


Abbildung 21. Senden einer Nachricht von WebSphere Partner Gateway an das HTTP-Servlet

Das HTTP-Servlet führt die folgenden Aufgaben aus:

1. Anhand der empfangenden Anforderungsnachricht stellt das Servlet fest, welcher Service und welche Methode dieses Service aufgerufen werden muss.
2. Das HTTP-Servlet erstellt auf der Basis der eingehenden Nachricht ein Geschäftsobjekt (siehe „Servlet erstellen“ auf Seite 94).
3. Das HTTP-Servlet ruft den Service unter Verwendung der entsprechenden SCA-Client-API (SCA - Service Component Architecture) auf:
  - a. Wenn die SCA-Service-Methode nur auf Anforderungen (request-only) basiert, wird keine Geschäftsantwort erwartet. Wenn das HTTP-Servlet den SCA-Service aufrufen kann, gibt das Servlet den HTTP-Statuscode 200 an WebSphere Partner Gateway zurück. Wenn das HTTP-Servlet den SCA-Service nicht aufrufen kann, wird der entsprechende HTTP-Fehlerstatuscode zurückgegeben.
  - b. Wenn die SCA-Service-Methode auf Anforderungen/Antworten (request-response) basiert, gibt der SCA-Service ein Antwortgeschäftsobjekt zurück. Das HTTP-Servlet serialisiert dieses Geschäftsobjekt in eine Nachricht. Das HTTP-Servlet gibt diese Nachricht in einer HTTP-Antwort an WebSphere Partner Gateway zurück. Wenn das HTTP-Servlet den SCA-Service aus irgendwelchen Gründen nicht erfolgreich aufrufen kann, wird der entsprechende HTTP-Fehlerstatuscode zurückgegeben.

Beispiel: Bei einer von Partnern empfangenen synchronen cXML-Nachricht sendet WebSphere Partner Gateway die cXML-Nachricht über HTTP an WebSphere Process Server. WebSphere Partner Gateway erwartet, dass WebSphere Process Server über dieselbe HTTP-Verbindung eine synchrone cXML-Antwort bereitstellt.

**Anmerkung:** WebSphere Partner Gateway unterstützt für Back-End-Integrationspakete Aufrufe, die nur auf Anforderungen (request-only) basieren. Mit der Datenbindung des Back-End-Integrationspakets können jedoch auf Anforderungen und Antworten (request-response) basierende Datenflüsse entwickelt werden (über WebSphere Partner Gateway-Benutzerexits). Die Klasse BCGBackEndIntegrationDataBindingUtil kann als Basis für die Konvertierung eines Antwortgeschäftsobjekts in einen Antwortdatenstrom verwendet werden.

---

## Nachrichten von WebSphere Process Server senden

In diesem Abschnitt wird beschrieben, wie WebSphere Partner Gateway eine Nachricht von WebSphere Process Server über HTTP empfängt und diese an einen externen Partner sendet.

### Anmerkungen:

1. Alle Dokumenttypen außer RosettaNet- und Binärdokumente können den Pakettyp 'None' aufweisen, wenn sie von WebSphere Process Server an WebSphere Partner Gateway gesendet werden. RosettaNet- und Binärdokumente müssen den Pakettyp 'Backend Integration' aufweisen.
2. Alle Dokumenttypen außer SOAP (Web-Service-Anforderungen) können den Pakettyp 'Backend Integration' aufweisen, wenn sie von WebSphere Process Server an WebSphere Partner Gateway gesendet werden. Für SOAP-Anforderungen muss der Pakettyp 'None' angegeben sein.

Informationen zum Senden und Empfangen von SOAP-Dokumenten über HTTP finden Sie in Kapitel 6, „Integration von WebSphere Process Server mit SOAP/HTTP“, auf Seite 109.

Wenn WebSphere Process Server eine Nachricht an WebSphere Partner Gateway sendet, wird eine Komponente verwendet, die eine HTTP POST-Operation für einen HTTP-Empfänger von WebSphere Partner Gateway ausführt. Diese Komponente wird von Ihnen entwickelt und als Service bereitgestellt.

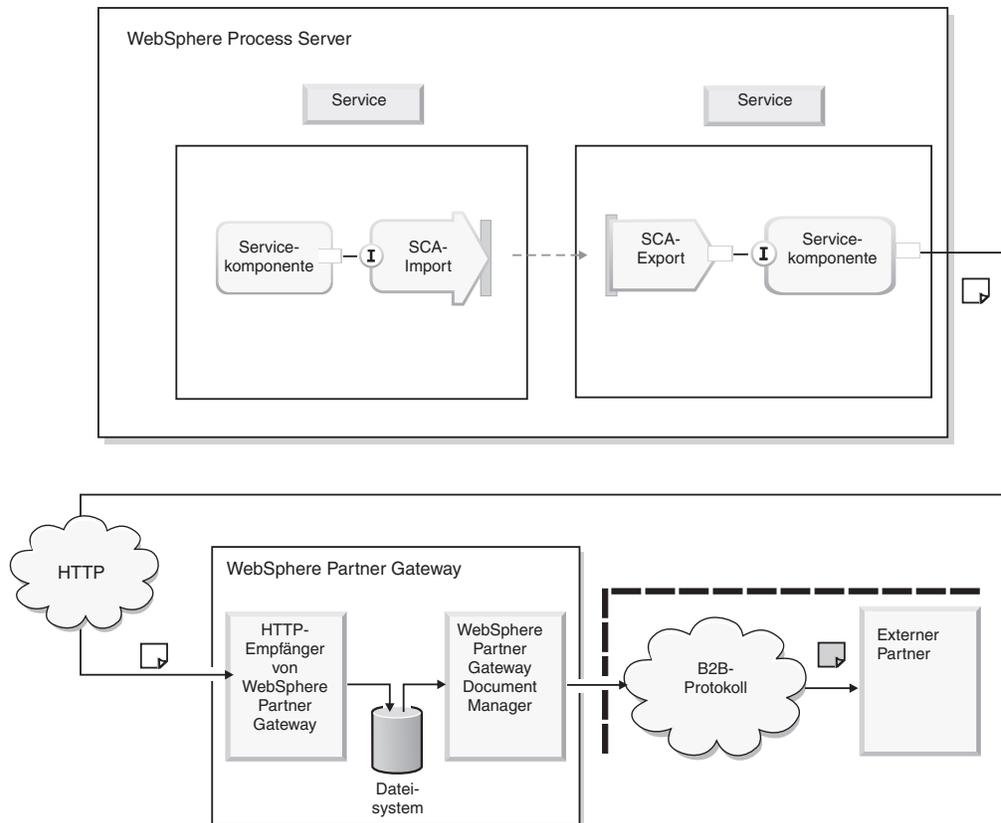


Abbildung 22. Senden einer Nachricht von WebSphere Process Server an den HTTP-Empfänger in WebSphere Partner Gateway

Nachrichten werden von den WebSphere Process Server-Services wie folgt an WebSphere Partner Gateway gesendet:

1. Ein SCA-Service von WebSphere Process Server, der Geschäftsdokumente über HTTP an WebSphere Partner Gateway senden muss, verwendet die zugehörige SCA-Importbindung, um die Komponente aufzurufen, die die HTTP POST-Operation ausführt. Die SCA-Importbindung des Service ruft die Komponente mit einem Geschäftsobjekt auf.
2. Die Komponente empfängt das Geschäftsobjekt in der zugehörigen Exportbindung. Dann serialisiert die Komponente das Geschäftsobjekt in ein Geschäftsdokument. Informationen dazu finden Sie im Abschnitt „Komponente zum Konvertieren und Senden der Nachricht erstellen“ auf Seite 96.
3. Die Komponente sendet die Nachricht mittels einer HTTP POST-Operation an die URL-Adresse des HTTP-Empfängers von WebSphere Partner Gateway.
4. WebSphere Partner Gateway empfängt diese Nachricht auf dem HTTP-Empfänger.

Die Art und Weise, wie WebSphere Partner Gateway antwortet, hängt davon ab, ob die Nachricht eine synchrone Antwort oder lediglich eine Antwort auf HTTP-Transportebene erfordert. Der HTTP-Empfänger verfügt über einen Synchronprüfungs-Konfigurationspunkt, der diese Feststellung trifft. Sie können die Synchronprüfung bei der Erstellung des Empfängers konfigurieren oder den Empfänger im Nachhinein entsprechend ändern. Informationen zum Synchronprüfungs-Konfigurationspunkt finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

- a. Wenn WebSphere Partner Gateway feststellt, dass die von WebSphere Process Server empfangene Nachricht lediglich eine Antwort auf Transportebene erfordert, antwortet der WebSphere Partner Gateway-Empfänger mit dem entsprechenden HTTP-Statuscode. Dann wird das Geschäftsdokument an den externen Partner weitergeleitet.
  - b. Wenn WebSphere Partner Gateway feststellt, dass die von WebSphere Process Server empfangene Nachricht ein Antwortgeschäftsdokument erfordert, bleibt die HTTP-Transportverbindung geöffnet. WebSphere Partner Gateway leitet das Geschäftsdokument an einen externen Partner weiter. Der externe Partner empfängt das Anforderungsdokument und sendet ein Antwortdokument an WebSphere Partner Gateway. WebSphere Partner Gateway gibt das Antwortgeschäftsdokument als HTTP-Antwort an WebSphere Process Server zurück.
5. Die Komponente empfängt die HTTP-Antwort. Sie stellt fest, ob die Antwort nur ein HTTP-Statuscode oder auch ein Geschäftsdokument ist:
- a. Wenn die Antwort ein HTTP-Statuscode ist, der auf einen Erfolg hinweist, gibt die Komponente den Aufruf zurück. Der SCA-Service von WebSphere Process Server, der die Komponente aufgerufen hat, setzt die Verarbeitung fort. Weist der HTTP-Statuscode jedoch auf einen Fehler hin, gibt die Komponente den entsprechenden Fehler zurück. Der SCA-Service von WebSphere Process Server führt die entsprechende Fehlerbehandlung durch.
  - b. Wenn die Antwort ein Geschäftsdokument ist, konvertiert die Komponente dieses Geschäftsdokument in ein Geschäftsobjekt. Dann gibt sie das Geschäftsobjekt an den SCA-Service von WebSphere Process Server zurück, der das Antwortgeschäftsobjekt verarbeitet.

Beispiel: Bei synchronen cXML-Nachrichten, die von WebSphere Process Server empfangen werden und für einen externen Partner vorgesehen sind, sendet WebSphere Partner Gateway die Nachricht an den Partner und empfängt synchron die Antwort. WebSphere Partner Gateway gibt die Antwort dann als HTTP-Antwort derselben HTTP-Anforderung an WebSphere Process Server zurück, die ursprünglich von WebSphere Process Server gesendet wurde.

**Anmerkung:** WebSphere Partner Gateway unterstützt für Back-End-Integrationspakete Aufrufe, die nur auf Anforderungen (request-only) basieren. Mit der Datenbindung des Back-End-Integrationspakets können jedoch auf Anforderungen und Antworten (request-response) basierende Datenflüsse entwickelt werden (über WebSphere Partner Gateway-Benutzerexits). Die Klasse `BCGBackEndIntegrationDataBindingUtil` kann als Basis für die Konvertierung eines Antwortgeschäftsobjekts in einen Antwortdatenstrom verwendet werden.

---

## Dokumente an WebSphere Process Server senden

In diesem Abschnitt werden die Schritte beschrieben, die erforderlich sind, damit WebSphere Process Server von WebSphere Partner Gateway gesendete Dokumente akzeptiert.

### WebSphere Partner Gateway konfigurieren

Befolgen Sie bei der Konfiguration von WebSphere Partner Gateway die Anweisungen im Handbuch *WebSphere Partner Gateway Hubkonfiguration*. Gehen Sie bei der Konfiguration von WebSphere Partner Gateway wie folgt vor:

1. Fügen Sie dem Synchronprüfungs-Konfigurationspunkt des HTTP-Empfängers den Handler `com.ibm.bcg.server.sync.SoapSyncHdlr` hinzu (bei der Verarbeitung von Zweiweg-anforderungen).

2. Stellen Sie sicher, dass das HTTP-Ziel des internen Partners die URL-Adresse angibt, an die WebSphere Partner Gateway Nachrichten für WebSphere Process Server sendet.

Das Ziel sollte auf das HTTP-Servlet verweisen, das in WebSphere Process Server implementiert wurde.

## WebSphere Process Server konfigurieren

In diesem Abschnitt wird beschrieben, wie Sie WebSphere Process Server für den Empfang von Nachrichten von WebSphere Partner Gateway konfigurieren.

### Geschäftsobjekt erstellen

Bei Verwendung der Klasse `BCGBackendIntegrationDataBindingUtil` für die Konvertierung einer Nachricht in ein Geschäftsobjekt muss das im Abschnitt „Geschäftsobjekt der höchsten Ebene und untergeordnete Geschäftsobjekte“ auf Seite 68 beschriebene Geschäftsobjekt der höchsten Ebene verwendet werden. Sie können zur Erstellung des Geschäftsobjekts den Editor für Geschäftsobjekte verwenden oder die `.xsd`-Datei (`BCG_TLO_BackendPackaging.xsd`) importieren, die sich auf dem Installationsdatenträger befindet.

Wenn Sie nicht das Back-End-Integrationspaket verwenden, sondern für das Dokument den Pakettyp 'None' angegeben haben, erstellen Sie das Geschäftsobjekt gemäß den Anforderungen des Dokumentaustauschs.

### Servlet erstellen

In diesem Abschnitt wird das Servlet beschrieben, das Sie erstellen müssen, damit WebSphere Partner Gateway über das HTTP-Transportprotokoll Nachrichten an WebSphere Process Server senden kann. Definieren Sie das Servlet für die folgenden Aufgaben:

- Empfang von Nachrichten von WebSphere Partner Gateway
- Konvertieren des Geschäftsdokuments in ein Geschäftsobjekt
- Aufruf des entsprechenden Service in WebSphere Process Server

Wenn Sie das Back-End-Integrationspaket verwenden, können Sie mit Hilfe der Klasse `BCGBackendIntegrationDataBindingUtil` eine HTTP-Anforderungsnachricht in ein Anforderungsgeschäftsobjekt konvertieren, das von Ihrem Service erwartet wird. Sie können `BCGBackendIntegrationDataBindingUtil` auch verwenden, um eine HTTP-Antwortnachricht auf der Basis des Antwortgeschäftsobjekts zu erstellen, das von Ihrem Service zurückgegeben wird. Es folgt ein Beispiel für die Verwendung der Klasse `BCGBackendIntegrationDataBindingUtil`, um eine HTTP-Anforderungsnachricht in ein Anforderungsgeschäftsobjekt zu konvertieren. In diesem Beispiel wird die Methode `read` und die Methode `getDataObject` verwendet. Wenn der Name des Objekts der höchsten Ebene bereits bekannt ist, können Sie auch die Methode `setTLOTypeName` aufrufen.

```
try{
    DataObject rootBO = null;
    BCGBackendIntegrationDataBindingUtil dataBinding = new
BCGBackendIntegrationDataBindingUtil();

    // request.getInputStream() gives the
// backend integration packaging input stream received by servlet
    dataBinding.read(request.getInputStream());
    rootBO = dataBinding.getDataObject();
}
```

```
catch(Exception exp){
    System.out.println("Error occurred while creating request business
object: " + exp);
}
```

Implementieren Sie das Servlet, nachdem Sie es erstellt haben. Die URL-Adresse, für die das Servlet empfangsbereit ist, sollte der URL-Adresse des Empfängerziels entsprechen, das für die Partnerverbindung in WebSphere Partner Gateway konfiguriert wurde.

---

## Dokumente von WebSphere Process Server senden

In diesem Abschnitt werden die Schritte beschrieben, die erforderlich sind, damit WebSphere Process Server Dokumente an WebSphere Partner Gateway senden kann.

### WebSphere Partner Gateway konfigurieren

Befolgen Sie bei der Konfiguration von WebSphere Partner Gateway die Anweisungen im Handbuch *WebSphere Partner Gateway Hubkonfiguration*. Gehen Sie bei der Konfiguration von WebSphere Partner Gateway wie folgt vor:

1. Erstellen Sie einen HTTP/S-Empfänger auf dem Hub, um die Dokumente zu empfangen, die von WebSphere Process Server gesendet werden (sofern dieser noch nicht vorhanden ist). Dieser Empfänger gibt die URL-Adresse an, an der die Empfängerkomponente von WebSphere Partner Gateway für Dokumente von WebSphere Process Server empfangsbereit ist.
2. Fügen Sie dem Synchronprüfungs-Konfigurationspunkt des HTTP-Empfängers den Handler `com.ibm.bcg.server.sync.SoapSyncHdlr` hinzu (bei der Verarbeitung von Zweiweganforderungen).

### WebSphere Process Server konfigurieren

In diesem Abschnitt wird beschrieben, wie Sie WebSphere Process Server für das Senden von Nachrichten an WebSphere Partner Gateway konfigurieren. Um einem WebSphere Process Server-Service das Senden einer Nachricht an WebSphere Partner Gateway über HTTP zu ermöglichen, benötigen Sie eine Komponente, die eine HTTP POST-Operation für die URL-Adresse ausführen kann, die der HTTP-Empfänger von WebSphere Partner Gateway für den Empfang von Dokumenten verwendet. Immer wenn ein WebSphere Process Server-Service ein Geschäftsdokument an einen externen Partner sendet, wird diese Komponente aufgerufen.

#### Geschäftsobjekt erstellen

Bei Verwendung der Klasse `BCGBackendIntegrationDataBindingUtil` für die Konvertierung eines Geschäftsdokuments in ein Geschäftsobjekt muss das im Abschnitt „Geschäftsobjekt der höchsten Ebene und untergeordnete Geschäftsobjekte“ auf Seite 68 beschriebene Geschäftsobjekt der höchsten Ebene verwendet werden. Sie können zur Erstellung des Geschäftsobjekts den Editor für Geschäftsobjekte verwenden oder die `.xsd`-Datei (`BCG_TLO_BackendPackaging.xsd`) importieren, die sich auf dem Installationsdatenträger befindet.

Wenn Sie nicht das Back-End-Integrationspaket verwenden, sondern den Pakettyp 'None' angegeben haben, erstellen Sie das Geschäftsobjekt gemäß den Anforderungen des Dokumentaustauschs.

## Komponente zum Konvertieren und Senden der Nachricht erstellen

Erstellen Sie eine Komponente, die das Geschäftsobjekt in einen Eingabedatenstrom konvertiert und dann die HTTP-Nachricht an WebSphere Partner Gateway sendet. Diese Komponente serialisiert das aus dem Service abgerufene Geschäftsobjekt in ein Geschäftsdokument. Anschließend kann mit Hilfe der JAVA-API `java.net.HTTPURLConnection` eine HTTP POST-Operation für die URL-Adresse ausgeführt werden, die der HTTP-Empfänger von WebSphere Partner Gateway für den Empfang von Dokumenten verwendet.

Das folgende Beispiel zeigt, wie die Komponente eine Nachricht mit POST an WebSphere Partner Gateway senden kann, wenn als Pakettyp 'None' verwendet wird.

```
String wpgHTTPTargetURL;    // URL of WebSphere Partner Gateway HTTP Target
byte payload[];            // message that needs to be sent to WebSphere Partner Gateway
// Set wpgHTTPTargetURL
// Set payload bytes. Convert business object received from service into bytes.
...
try{
    java.net.HttpURLConnection uc = (java.net.HttpURLConnection)
        new URL(wpgHTTPTargetURL).openConnection();
    uc.setDoInput(true);
    uc.setDoOutput(true);
    uc.setRequestMethod("POST");
    // Set the content type you want to send
    uc.setRequestProperty("Content-Type", "text/xml");
    uc.connect();
    BufferedOutputStream os = new BufferedOutputStream (uc.getOutputStream());
    os.write( payload);
    os.close();
    // If you are expecting response business document from WebSphere Partner
    // Gateway, then you have to read the response
    InputStream is = uc.getInputStream();
    if(is != null){
        BufferedInputStream bis = new BufferedInputStream
        (uc.getInputStream());
        // Read bytes from BufferedInputStream obtained above
        // Convert response business document received from WebSphere
        // Partner Gateway into business object
        ...
        bis.close();
    }
}
catch(Throwable ex){
    System.out.println("Cannot post: " + ex.getMessage());
    ex.printStackTrace();
}
}
```

**Wichtig:** Wenn Sie zum Senden von Dokumenten an WebSphere Process Server das Back-End-Integrationspaket verwenden, können Sie die Klasse `BCGBackEndIntegrationDataBindingUtil` verwenden, um die von dem Service (oder der Komponente) empfangenen Geschäftsobjekte in Geschäftsdokumente zu konvertieren, die von WebSphere Partner Gateway erwartet werden. Wenn Sie ein Antwortgeschäftsdokument von WebSphere Partner Gateway erwarten, können Sie mit Hilfe von `BCGBackEndIntegrationDataBindingUtil` auch die HTTP-Antwortnachricht auf der Basis des Antwortgeschäftsbereichs erstellen, das von Ihrem Service zurückgegeben wird.

---

## Kapitel 5. Integration von WebSphere Process Server mit JMS als Transportprotokoll

In diesem Kapitel wird die Integration von WebSphere Partner Gateway mit WebSphere Process Server über das JMS-Transportprotokoll beschrieben. Es enthält Prozeduren für die Konfiguration von WebSphere Partner Gateway für JMS, wenn WebSphere Platform Messaging (Standard-Messaging-Provider) und ein Service-Integration-Bus als JMS-Provider verwendet werden und WebSphere Partner Gateway auf WebSphere Application Server installiert wird.

Dieses Kapitel enthält Informationen zum Senden und Empfangen von Dokumenten zwischen WebSphere Partner Gateway und WebSphere Process Server unter Verwendung des JMS-Transportprotokolls. Es behandelt die folgenden Themen:

- „Dokumentenversand mit dem JMS-Transportprotokoll“
- „Dokumentenempfang mit dem JMS-Transportprotokoll“ auf Seite 99
- „WebSphere Process Server-Umgebung in WebSphere Application Server einrichten“ auf Seite 100
- „JMS konfigurieren, wenn WebSphere Partner Gateway auf WebSphere Application Server installiert ist“ auf Seite 103

---

### Dokumentenversand mit dem JMS-Transportprotokoll

Beim Senden eines Dokuments von WebSphere Partner Gateway an WebSphere Process Server über das JMS-Transportprotokoll müssen Sie die JMS-Exportbindung verwenden, um die WebSphere Process Server-Komponente über JMS aufzurufen. SCA-Komponenten (SCA - Service Component Architecture) können JMS-Nachrichten aus der JMS-Warteschlange empfangen, die in der entsprechenden JMS-Exportbindung konfiguriert ist. Abb. 23 auf Seite 98 bietet eine Übersicht darüber, wie WebSphere Partner Gateway über das JMS-Transportprotokoll ein Dokument an WebSphere Process Server sendet.

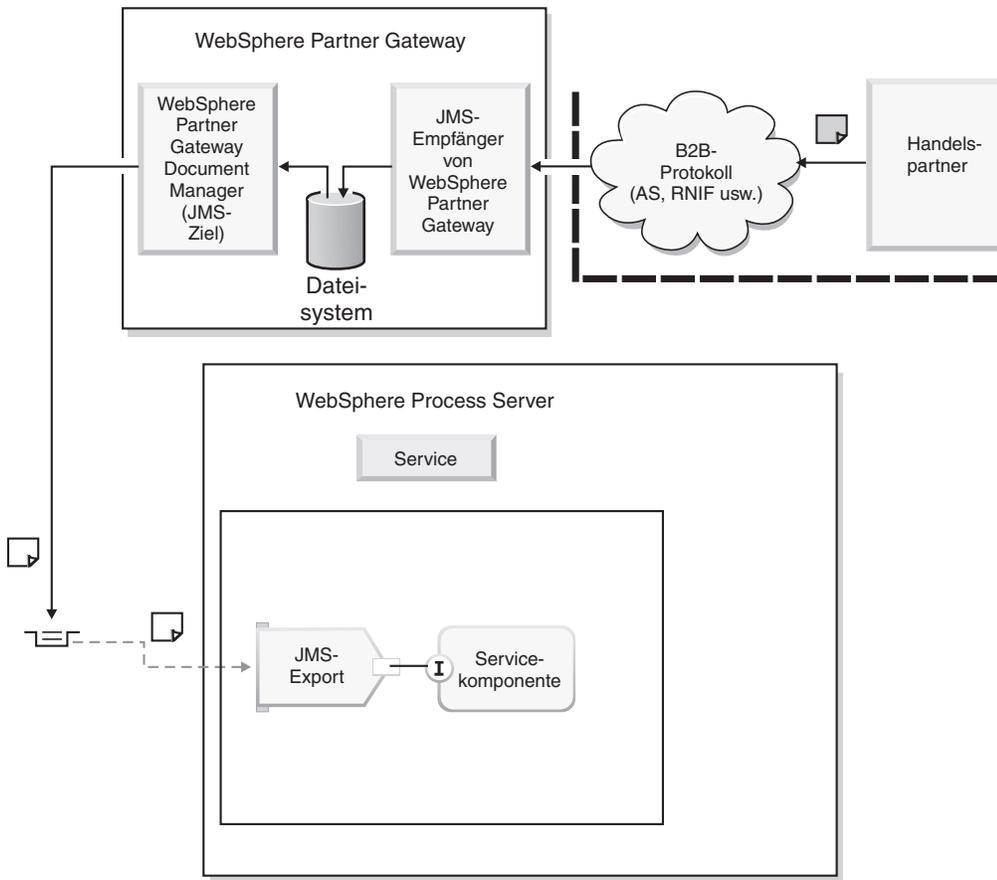


Abbildung 23. JMS-Export zum Aufrufen von SCA-Services über JMS

In den folgenden Schritten wird beschrieben, wie Sie mit SCA-JMS-Exportbindungen SCA-Services über JMS aufrufen können:

1. Ein Handelspartner sendet ein Geschäftsdokument an WebSphere Partner Gateway und verwendet dazu ein B2B-Protokoll.
2. WebSphere Partner Gateway empfängt das Geschäftsdokument von dem Handelspartner.
3. Unter Verwendung der konfigurierten Partnerverbindung für dieses Geschäftsdokument leitet WebSphere Partner Gateway das Dokument an WebSphere Process Server weiter.

Bei JMS-basierter Integration wird erwartet, dass das Empfängerziel der Partnerverbindung ein JMS-Ziel ist.

Dieses JMS-Ziel ist für das Senden von JMS-Nachrichten an das SI-Bus-Ziel konfiguriert.

Das SI-Bus-Ziel ist die JMS-Warteschlange, die in der SCA-JMS-Exportbindung der SCA-Komponente konfiguriert ist.

## Dokumentenempfang mit dem JMS-Transportprotokoll

Damit WebSphere Partner Gateway ein Dokument von WebSphere Process Server über das JMS-Transportprotokoll empfangen kann, muss die JMS-Importbindung verwendet werden, um externe Services über JMS aufzurufen. SCA-Komponenten können JMS-Nachrichten an die JMS-Warteschlange senden, die in der entsprechenden JMS-Importbindung konfiguriert ist. Abb. 24 bietet eine Übersicht darüber, wie WebSphere Partner Gateway über das JMS-Transportprotokoll ein Dokument von WebSphere Process Server empfängt.

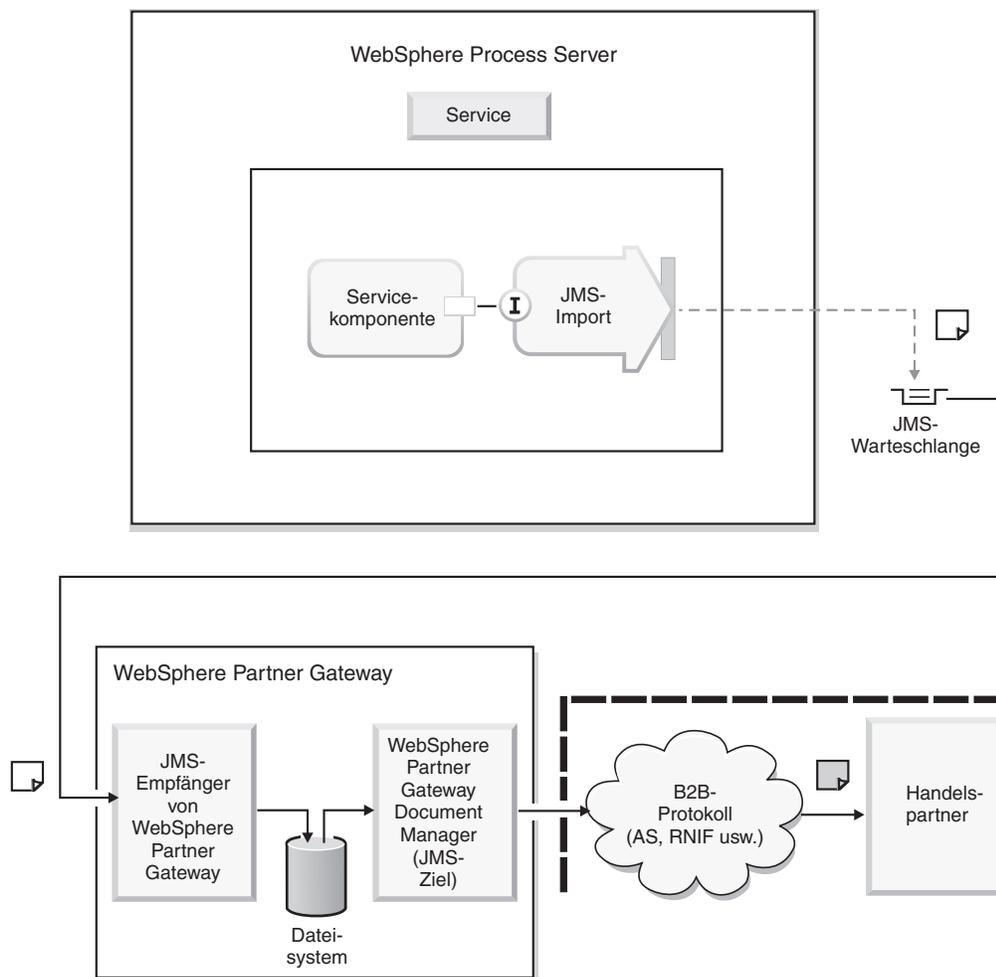


Abbildung 24. JMS-Import zum Empfangen eines Dokuments über JMS

In den folgenden Schritten wird beschrieben, wie mit SCA-JMS-Importbindungen ein Dokument von WebSphere Process Server empfangen wird:

1. Der JMS-Empfänger von WebSphere Partner Gateway empfängt ein Geschäftsdokument von WebSphere Process Server.
2. Unter Verwendung der konfigurierten Partnerverbindung für dieses Geschäftsdokument leitet WebSphere Partner Gateway das Dokument an den Handelspartner weiter.
3. Der Handelspartner empfängt das Geschäftsdokument von WebSphere Partner Gateway über das vereinbarte B2B-Protokoll.

Bei JMS-basierter Integration wird erwartet, dass die im JMS-Empfänger von WebSphere Partner Gateway konfigurierte JMS-Warteschlange das JMS-Ziel ist, an das die WebSphere Process Server-Services JMS-Nachrichten senden.

Dieses JMS-Warteschlangenziel ist in der SCA-JMS-Importbindung der SCA-Komponente konfiguriert.

---

## WebSphere Process Server-Umgebung in WebSphere Application Server einrichten

In diesem Abschnitt werden die erforderlichen Schritte zur Einrichtung der WebSphere Process Server-Umgebung für den JMS-Transport in WebSphere Application Server beschrieben. Dazu gehört das Erstellen und Konfigurieren der WebSphere Process Server-Artefakte.

Dieser Abschnitt enthält Informationen zu den folgenden Themen:

- „SCA-Service mit WSDL erstellen“
- „JMS-Import- und -Exportbindungen anpassen“
- „JMS-Datenbindung implementieren“ auf Seite 101
- „Funktionsselektor anpassen“ auf Seite 102

### SCA-Service mit WSDL erstellen

In Verbindung mit Export- und Importbindungen wird ein SCA-Service verwendet, um die B2B-Funktionalität sowie die Funktionen für die Interaktion von Handelspartnern von WebSphere Partner Gateway zu nutzen. Anweisungen dazu, wie Sie mit WSDL (Web Services Description Language) einen SCA-Service erstellen und definieren, finden Sie im Information Center für WebSphere Process Server.

### JMS-Import- und -Exportbindungen anpassen

Die JMS-Importbindung einer Komponente wird verwendet, um externe Services über JMS aufzurufen. SCA-Komponenten können JMS-Nachrichten an die JMS-Warteschlange senden, die in der entsprechenden JMS-Importbindung konfiguriert ist. Gehen Sie wie folgt vor, damit SCA-Komponenten Nachrichten an die Warteschlange senden können, die im JMS-Empfänger von WebSphere Partner Gateway konfiguriert ist:

1. Stellen Sie die von der SCA-JMS-Importbindung der Komponente geforderte JMS-Datenbindung bereit. Informationen dazu finden Sie im Abschnitt „JMS-Datenbindung implementieren“ auf Seite 101.
2. Geben Sie die Attribute der JMS-Importbindung an, die sich auf Ihre Umgebung beziehen. Der JMS-Import der Komponente stellt Nachrichten in das JMS-Ziel, das in der JMS-Importbindung angegeben ist. Das JMS-Ziel, das Sie in der JMS-Importbindung angeben, sollte auf die JMS-Warteschlange verweisen, die im JMS-Empfänger von WebSphere Partner Gateway konfiguriert ist. Detaillierte Informationen zu den Attributen der JMS-Importbindung finden Sie im Information Center für WebSphere Process Server.

Die JMS-Exportbindung einer Komponente wird verwendet, um die WebSphere Process Server-Komponente über JMS aufzurufen. Gehen Sie wie folgt vor, damit SCA-Komponenten JMS-Nachrichten aus dem JMS-Ziel empfangen können, das in der entsprechenden JMS-Exportbindung konfiguriert ist:

1. Stellen Sie die von der SCA-JMS-Exportbindung der Komponente geforderte JMS-Datenbindung bereit. Informationen dazu finden Sie im Abschnitt „JMS-Datenbindung implementieren“ auf Seite 101.

2. Stellen Sie den von der SCA-JMS-Exportbindung der Komponente geforderten Funktionsselektor bereit. Informationen dazu finden Sie im Abschnitt „Funktionsselektor anpassen“ auf Seite 102.
3. Geben Sie die Attribute für die JMS-Exportbindung an, die sich auf Ihre Umgebung beziehen. Der JMS-Export einer Komponente ruft die JMS-Nachricht aus dem JMS-Ziel ab, das in der JMS-Exportbindung angegeben wurde. Das JMS-Ziel, das Sie in der JMS-Exportbindung angeben, sollte auf die JMS-Warteschlange verweisen, die im JMS-Ziel von WebSphere Partner Gateway konfiguriert ist. Detaillierte Informationen zu den Attributen der JMS-Exportbindung finden Sie im Information Center für WebSphere Process Server.

## JMS-Datenbindung implementieren

Die WebSphere Process Server-Komponenten für SCA-JMS-Import- und -Export stellen Konfigurationsdaten für die JMS-Datenbindung bereit. Die JMS-Datenbindung wird vom SCA-JMS-Import und vom SCA-JMS-Export zum Konvertieren von Geschäftsobjekten in JMS-Nachrichten (und umgekehrt) verwendet.

- SCA-JMS-Export: Der SCA-JMS-Export eines Service nutzt eine konfigurierte JMS-Datenbindung, um eine von WebSphere Partner Gateway empfangene JMS-Nachricht wie von der Methode des Service erwartet in ein Geschäftsobjekt zu konvertieren. Dieses Geschäftsobjekt wird verwendet, um die Methode eines Service aufzurufen.
- SCA-JMS-Import: Der SCA-JMS-Import eines Service nutzt eine konfigurierte JMS-Datenbindung, um ein Geschäftsobjekt wie von WebSphere Partner Gateway erwartet in eine JMS-Nachricht zu konvertieren.

Informationen zur Entwicklung und Implementierung der JMS-Datenbindung finden Sie im Information Center für WebSphere Process Server.

Bei Verwendung von JMS für die Back-End-Integration unterstützt WebSphere Partner Gateway die Pakettypen 'None' und 'Backend Integration'.

Bei Verwendung von 'None' können Sie die von WebSphere Process Server bereitgestellte Standard-JMS-Datenbindung verwenden oder eine eigene Datenbindung implementieren. In der Regel müssen Sie eine eigene Datenbindung schreiben, wenn das Format der zwischen WebSphere Partner Gateway und WebSphere Process Server ausgetauschten JMS-Nachrichten nicht XML ist.

Bei Verwendung des Back-End-Integrationspakets kann die von WebSphere Partner Gateway bereitgestellte Klasse `com.ibm.bcg.dataBinding.BCGBackendIntegrationJMSSDataBindingImpl` genutzt werden. Zur Anpassung der Verarbeitung können Sie die Datenbindung mit Hilfe der Klasse `com.ibm.bcg.dataBinding.BCGBackendIntegrationDataBindingUtil` oder der Klasse `com.ibm.bcg.dataBinding.BCGBackendIntegrationJMSSDataBindingImpl` von WebSphere Partner Gateway implementieren. Weitere Informationen zu diesen Klassen finden Sie in den Abschnitten „Klasse `BCGBackendIntegrationJMSSDataBindingImpl`“ auf Seite 85 und „Klasse `BCGBackendIntegrationDataBindingUtil`“ auf Seite 79.

Zur Implementierung der JMS-Datenbindungsschnittstelle für das Back-End-Integrationspaket müssen benutzerdefinierte Lese- und Schreibmethoden (`read/write`) erstellt werden.

## Funktionsselektor anpassen

Der Funktionsselektor ist für die SCA-JMS-Exportbindung erforderlich. Mit Hilfe des Funktionsselektors wird festgestellt, welche Servicemethode für ein Geschäftsdokument aufgerufen werden muss, das am JMS-Ziel empfangen wird.

Verwenden Sie eines der folgenden Verfahren, um einen Funktionsselektor zu implementieren:

- Paket 'None': Wenn WebSphere Partner Gateway eine JMS-Nachricht mit dem Paket 'None' an WebSphere Process Server sendet, muss der Benutzer festlegen, welche Methode des SCA-Service aufgerufen wird. Die einzige Möglichkeit, bei Verwendung des Pakets 'None' den Typ eines Geschäftsdokuments zu ermitteln, besteht darin, das Dokument zu parsen. Verwenden Sie eines der folgenden Verfahren, um das Dokument zu parsen:
  - Dispatcher: Bei Verwendung des Pakets 'None' empfiehlt IBM, das Dokument nicht mit dem Funktionsselektor zu parsen. Diese Vorgehensweise ist redundant und kostenintensiv, da das Dokument in der Datenbindung ebenfalls geparkt werden muss. IBM empfiehlt die Entwicklung eines Dispatchers oder einer Funktionsselektor-Komponente, der bzw. die als Front-End für die eigentliche WebSphere Process Server-Komponente dient, die an dem Geschäftsdokument interessiert ist. Die JMS-Exportbindung dieser Dispatcherkomponente empfängt das Geschäftsdokument von WebSphere Partner Gateway. Die Dispatcherkomponente verfügt über nur eine Methode, über die alle möglichen Geschäftsdokumente empfangen werden können. Die JMS-Exportbindung zerlegt (un-marshall) die JMS-Nachrichten in Geschäftsobjekte. Dann stellt das Dispatchermodul fest, welche Servicekomponente an diesem Geschäftsobjekt interessiert ist und ruft die Komponente mit diesem Geschäftsobjekt auf.
  - Minimales Parsing: Wenn eine SCA-JMS-Exportbindung über mehrere Methodenbindungen verfügt, kann der Funktionsselektor implementiert werden, um zur Ermittlung des Geschäftsdokumententyps (z. B. Bestellung, Auftragsbestätigung, Rechnung usw.) ein minimales Parsing für das Geschäftsdokument auszuführen. Der Funktionsselektor kann dann den Typ des Geschäftsdokuments als nativen Methodennamen zurückgeben. Bei der Konfiguration kann der native Methodename in der Methodenbindung der SCA-JMS-Exportbindung als Geschäftsdokumententyp angegeben werden. Bei Verwendung der Methodenbindung kann der SCA-JMS-Export feststellen, welche Servicemethode aufgerufen werden muss.

**Anmerkung:** Die Datenbindung führt zur Erstellung des Geschäftsobjekts ein umfassendes Parsing für das Geschäftsdokument aus. Deshalb sollte das Parsing des Funktionsselektors für das Geschäftsdokument absolut minimal sein. Ist das Parsing des Funktionsselektors zu kostenintensiv, sollten Sie dieses Verfahren nicht verwenden.

- Einfacher Funktionsselektor: Erwartet der Benutzer nur einen Geschäftsdokumententyp, können Sie einen einfachen Funktionsselektor entwickeln, der das Dokument nicht parst und immer denselben nativen Methodennamen zurückgibt. Bei der Konfiguration kann der native Methodename in der Methodenbindung der SCA-JMS-Exportbindung als Methodename angegeben werden. Bei Verwendung der Methodenbindung kann der SCA-JMS-Export feststellen, welche Servicemethode aufgerufen werden muss. Ein Vorteil dieses Verfahrens besteht darin, dass eingehende Geschäftsdokumente nicht geparkt werden müssen. Der Nachteil ist allerdings, dass sich in der JMS-Warteschlange nur ein Typ von Geschäftsdokument befinden kann.

- Paket 'Backend Integration': Wenn WebSphere Partner Gateway mit dem Back-End-Integrationspaket eine JMS-Nachricht an WebSphere Process Server sendet, kann der Benutzer mit Hilfe der JMS-Header dieses Pakets einen Funktionsselektor entwickeln. Die JMS-Eigenschaften können vom Funktionsselektor aus der JMS-Nachrichteneingabe gelesen werden. Der Funktionsselektor kann den Wert eines der JMS-Header des Back-End-Integrationspakets verwenden, um einen nativen Methodennamen zu erstellen. Im Falle einer RosettaNet-Nachricht kann beispielsweise die JMS-Eigenschaft x-aux-payload-root-tag gelesen werden, um den Typ des Geschäftsdokuments zu ermitteln. Der Funktionsselektor kann dann den nativen Methodennamen erstellen, der mit dem nativen Methodennamen in der Methodenbindung des JMS-Exports übereinstimmt.

Mit WebSphere Integration Developer können Sie einen angepassten Funktionsselektor erstellen, indem Sie die Schnittstelle `commonj.connector.runtime.FunctionSelector` implementieren. Beispiel:

```
public class PurchaseOrderSelector implements FunctionSelector {
    public String generateEISFunctionName(Object[] arg0)
        throws SelectorException {
        return "receiveMessage";
    }
}
```

---

## JMS konfigurieren, wenn WebSphere Partner Gateway auf WebSphere Application Server installiert ist

In diesem Abschnitt werden die erforderlichen Schritte zur Einrichtung der WebSphere Partner Server-Umgebung für den JMS-Transport in WebSphere Application Server beschrieben, wenn WebSphere Platform Messaging als Messaging-Provider verwendet wird. Es wird davon ausgegangen, dass WebSphere Partner Gateway auf WebSphere Application Server installiert ist.

Die Informationen in diesem Abschnitt setzen voraus, dass der Benutzer mit WebSphere Platform Messaging und Service-Integration-Buses vertraut ist.

Dieser Abschnitt enthält Informationen zu den folgenden Themen:

- „Busse, JMS-Warteschlangen und Verbindungsfactorys erstellen und konfigurieren“ auf Seite 104
- „JMS-Empfänger erstellen“ auf Seite 106
- „JMS-Ziel erstellen“ auf Seite 107
- „Zielwarteschlange erstellen“ auf Seite 108

Bei Verwendung der Messaging-Steuerkomponente und der JNDI (Java Naming and Directory Interface) des Anwendungsservers der WebSphere Partner Gateway-Komponente kann es für den Kunden zu Einschränkungen bei den Bustopologien kommen. Deshalb empfiehlt IBM, dass die WebSphere Partner Gateway-Komponenten (JMS-Empfänger und JMS-Ziel) eine Verbindung zur Messaging-Steuerkomponente auf einem anderen Anwendungsserver herstellen, der WebSphere Process Server ausführen kann oder auch nicht. Eine Übersicht über diese Topologie finden Sie im Abschnitt Abb. 25 auf Seite 104.

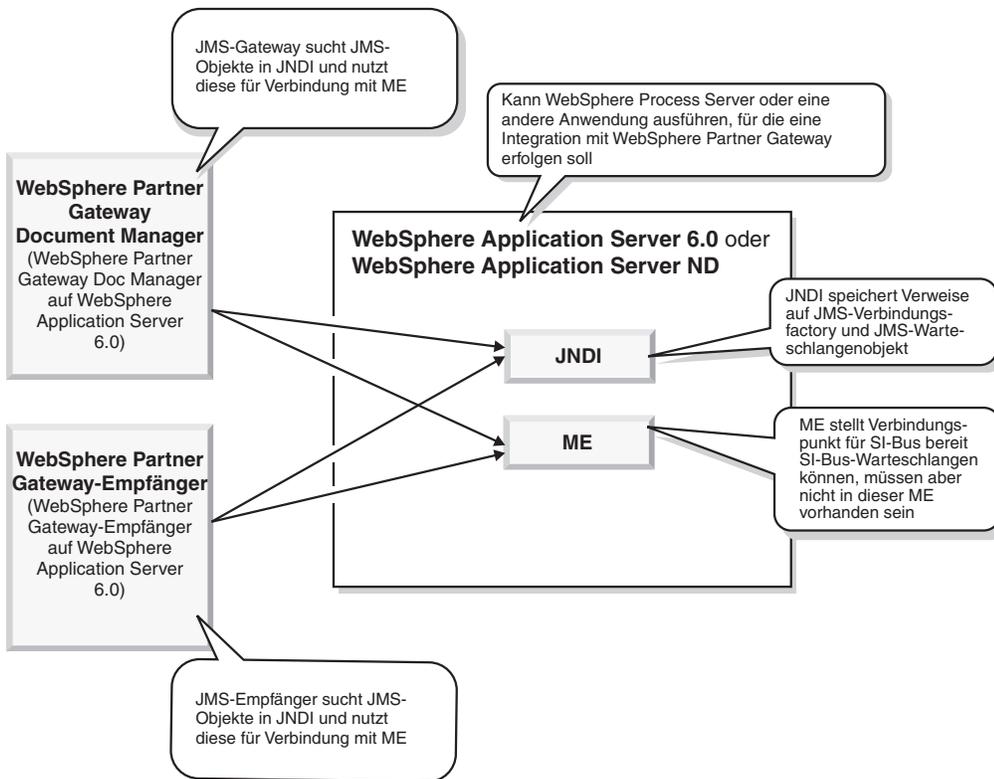


Abbildung 25. Zugriff auf Messaging-Steuerkomponente und JNDI von einer anderen WebSphere Application Server-Instanz

## Busse, JMS-Warteschlangen und Verbindungs-factorys erstellen und konfigurieren

In diesem Abschnitt wird beschrieben, wie Busse, JMS-Warteschlangen und Verbindungs-factorys erstellt und konfiguriert werden, die von WebSphere Partner Gateway zum Senden und Empfangen von Nachrichten verwendet werden.

### Service-Integration-Bus erstellen

Der Service-Integration-Bus unterstützt Anwendungen, die nachrichtenbasierte und serviceorientierte Architekturen verwenden. Ein Bus besteht aus einer Gruppe von miteinander verbundenen Servern und Clustern, die als Bus-Member hinzugefügt wurden. Die Verbindung zwischen einer Anwendung und einem Bus erfolgt über eine der Messaging-Steuerkomponenten, die den jeweiligen Bus-Membnern zugeordnet sind.

Verwendung der WebSphere-Administrationskonsole:

1. Erstellen Sie einen Bus, und geben Sie diesem Bus einen Namen:
  - a. Klicken Sie auf **Serviceintegration** > **Busse**.
  - b. Klicken Sie auf **Neu**, und geben Sie einen Busnamen an. Beispiel: SIBUS.
  - c. Klicken Sie auf **Anwenden**.
  - d. Speichern Sie die Konfiguration, indem Sie im angezeigten Nachrichtenfenster auf **Speichern** klicken. Mit dieser Nachricht wird bestätigt, dass Sie die Änderungen auf die Hauptkonfiguration anwenden möchten.

- e. Klicken Sie erneut auf **Speichern**, wenn Sie aufgefordert werden, das Master-Repository mit Ihren Änderungen zu aktualisieren.
2. Fügen Sie dem Bus die Bus-Member hinzu:
  - a. Klicken Sie auf den Namen für den neu erstellten Bus.
  - b. Klicken Sie im Teilfenster **Weitere Merkmale** auf **Bus-Member**.
  - c. Klicken Sie auf **Hinzufügen**, und wählen Sie den gewünschten Server oder Cluster aus.
  - d. Klicken Sie auf **Weiter**, und klicken Sie dann auf **Fertig stellen**, um das Hinzufügen der neuen Bus-Member zu bestätigen.
  - e. Speichern Sie die Konfiguration, indem Sie im angezeigten Nachrichtenfenster auf **Speichern** klicken. Mit dieser Nachricht wird bestätigt, dass Sie die Änderungen auf die Hauptkonfiguration anwenden möchten.
  - f. Klicken Sie erneut auf **Speichern**, wenn Sie aufgefordert werden, das Master-Repository mit Ihren Änderungen zu aktualisieren.
3. Erstellen Sie eine Zielwarteschlange:
  - a. Klicken Sie in der Standardkonsole von WebSphere Application Server in der linken Anzeige auf **Systemintegration**.
  - b. Klicken Sie auf **Busse < SIBUS** (oder auf den Namen für den in Schritt 1 erstellten Bus).
  - c. Klicken Sie im Teilfenster **Weitere Merkmale** auf **Ziele**. Klicken Sie auf **Neu**.
  - d. Wählen Sie das Optionsfeld **Warteschlange** für den Zieltyp aus, und klicken Sie auf **Weiter**.
  - e. Geben Sie unter **ID** einen Wert ein. Beispiel: **Request**. Daraufhin wird die Zielwarteschlange auf dem Bus erstellt.
4. Speichern Sie die Konfiguration, indem Sie im angezeigten Nachrichtenfenster auf **Speichern** klicken. Mit dieser Nachricht wird bestätigt, dass Sie die Änderungen auf die Hauptkonfiguration anwenden möchten.
5. Klicken Sie erneut auf **Speichern**, wenn Sie aufgefordert werden, das Master-Repository mit Ihren Änderungen zu aktualisieren.

### JMS-Warteschlangenverbindungsfactory erstellen

Mit einer JMS-Warteschlangenverbindungsfactory können Verbindungen zum zugeordneten JMS-Provider von JMS-Warteschlangen für Punkt-zu-Punkt-Messaging erstellt werden.

Verwendung der WebSphere-Administrationskonsole:

1. Erstellen Sie eine Warteschlangenverbindungsfactory, indem Sie die Felder **Name** und **JNDI-Name** ausfüllen und dabei die folgende Syntax verwenden:
  - **Name:** SIBUS.JMSTargetQCF
  - **JNDI-Name:** SIBUS/JMSTargetQCF

Dabei ist SIBUS der Name für den Bus, den Sie im vorherigen Schritt erstellt haben.
2. Wählen Sie den Bus aus. Beispiel: **SIBUS**.
3. Klicken Sie im daraufhin angezeigten Fenster auf die soeben erstellte Warteschlangenverbindungsfactory, und geben Sie unter **Provider-Endpoints** Folgendes ein:
 

```
IP-adresse/Name:7276:BootstrapBasicMessaging
```

Dabei ist 'IP-adresse' die IP-Adresse oder der Name der Maschine, auf der WebSphere Application Server ausgeführt wird. Es wird erwartet, dass die Messaging-Steuerkomponente für diesen Service-Integration-Bus auf dieser

Maschine ausgeführt wird. 7276 ist die Portnummer, die für SIB\_ENDPOINT\_ADDRESS für die WebSphere-Instanz angegeben wurde. Wenn die Messaging-Steuerkomponente auf dem System mit der IP-Adresse 9.26.234.100 ausgeführt wird und SIB\_ENDPOINT\_ADDRESS für die auf diesem Server ausgeführte WebSphere-Instanz den Wert 7276 aufweist, werden die Provider-Endpoints wie folgt angegeben:

```
9.26.234.100:7276:BootstrapBasicMessaging
```

4. Speichern Sie die Konfiguration, indem Sie im angezeigten Nachrichtenfenster auf **Speichern** klicken. Mit dieser Nachricht wird bestätigt, dass Sie die Änderungen auf die Hauptkonfiguration anwenden möchten.
5. Klicken Sie erneut auf **Speichern**, wenn Sie aufgefordert werden, das Master-Repository mit Ihren Änderungen zu aktualisieren.

## JMS-Warteschlange erstellen

JMS-Warteschlangen werden als Ziel für Punkt-zu-Punkt-Messaging verwendet.

Verwendung der WebSphere-Administrationskonsole:

1. Erweitern Sie das Menü **Ressourcen**, und klicken Sie auf **JMS-Provider > Standard-Messaging**.
2. Klicken Sie im Abschnitt **Ziele** der daraufhin angezeigten Seite auf **JMS-Warteschlangen**.
3. Klicken Sie auf **Neu**.
4. Geben Sie einen Warteschlangennamen in die Felder **Name** und **JNDI-Name** ein, und verwenden Sie dabei die folgende Syntax:
  - **Name:** Request.JMSTarget
  - **JNDI-Name:** Request/JMSTarget
5. Wählen Sie den Busnamen (z. B. SIBUS) und den Warteschlangennamen aus den Dropdown-Listen aus.
6. Klicken Sie auf **OK**.
7. Speichern Sie die Konfiguration, indem Sie im angezeigten Nachrichtenfenster auf **Speichern** klicken. Mit dieser Nachricht wird bestätigt, dass Sie die Änderungen auf die Hauptkonfiguration anwenden möchten.
8. Klicken Sie erneut auf **Speichern**, wenn Sie aufgefordert werden, das Master-Repository mit Ihren Änderungen zu aktualisieren.

## JMS-Empfänger erstellen

Dieser Abschnitt enthält Anweisungen zur Erstellung eines Empfängers in WebSphere Partner Gateway.

Verwendung der Community Console von WebSphere Partner Gateway:

1. Klicken Sie auf **Hubadmin > Hubkonfiguration > Empfänger**, um die Empfängerliste anzuzeigen.
2. Klicken Sie auf der Seite **Empfängerliste** auf **Empfänger erstellen**.
3. Führen Sie im Abschnitt mit den Empfängerdetails die folgenden Schritte aus:
  - a. Geben Sie einen Namen für den Empfänger ein. Sie könnten den Empfänger beispielsweise JMSTarget nennen. Dies ist ein erforderliches Feld. Der Name, den Sie hier eingeben, wird in der Empfängerliste angezeigt.
  - b. Geben Sie optional den Status des Empfängers an. **Aktiviert** ist die Standardeinstellung. Ein aktivierter Empfänger ist bereit, Dokumente zu empfangen. Ein inaktivierter Empfänger kann keine Dokumente empfangen.
  - c. Geben Sie optional eine Beschreibung für den Empfänger ein.

4. Wählen Sie in der Transportliste **JMS** aus.
5. Geben Sie die URL-Adresse des JMS-Providers ein. Beispiel:  
`iiop://system-ip:2809/`  
 Dabei ist 'system-ip' die IP-Adresse des Systems, auf dem WebSphere Platform Messaging ausgeführt wird, und 2809 ist der Standardport für die Ausführung des Bootstrap-Servers.
6. Geben Sie einen Wert für den Namen der JMS-Warteschlange ein. Dies ist ein erforderliches Feld. Der Name sollte mit dem JNDI-Namen der JMS-Warteschlange übereinstimmen, die im vorherigen Abschnitt erstellt wurde. Beispiel: Request/JMSTarget.
7. Geben Sie einen Wert für den Namen der JMS-Factory ein. Dies ist ein erforderliches Feld. Der Name sollte mit dem Namen der oben erstellten Warteschlangenverbindungsfactory übereinstimmen. Beispiel: SIBUS/JMSTargetQCF.
8. Geben Sie den Namen der JNDI-Factory wie folgt ein:  
`com.ibm.websphere.naming.WsnInitialContextFactory`
9. Klicken Sie auf **Speichern**.

## JMS-Ziel erstellen

Dieser Abschnitt enthält Anweisungen zur Erstellung des Ziels in WebSphere Partner Gateway.

Erstellen Sie unter Verwendung der Community Console von WebSphere Partner Gateway ein JMS-Ziel wie folgt:

1. Klicken Sie auf **Kontenadmin > Profile > Partner**.
2. Geben Sie die Suchkriterien ein, und klicken Sie auf **Suchen**. Sie können auch auf **Suchen** klicken, ohne Suchkriterien einzugeben, um eine Liste aller Partner anzuzeigen.
3. Klicken Sie auf das Symbol **Details anzeigen**, um das Partnerprofil anzuzeigen.
4. Klicken Sie auf **Ziele**.
5. Klicken Sie auf **Erstellen**.
6. Geben Sie auf der Seite **Liste der Ziele** einen Namen ein, um das Ziel anzugeben. Dies ist ein erforderliches Feld.
7. Wählen Sie **JMS** in der Liste mit den Transporttypen aus.
8. Geben Sie in das Feld **Adresse** die URI ein, an die das Dokument gesendet werden soll. Beispiel:  
`iiop://system-ip:2809`  
 Dabei ist 'systemip' die IP-Adresse des Systems, auf dem WebSphere Platform Messaging ausgeführt wird.
9. Geben Sie in das Feld **JMS-Factory-Name** den Namen der Java-Klasse ein, den der JMS-Provider für die Verbindung mit der JMS-Warteschlange verwendet. Dabei handelt es sich um den JNDI-Namen für die JMS-Factory, die zuvor erstellt wurde. Beispiel:  
`SIBUS/JMSTargetQCF`
10. Geben Sie in das Feld **JMS-Warteschlangenname** den Namen der JMS-Warteschlange ein, an die Dokumente gesendet werden sollen. Dabei handelt es sich um den JNDI-Namen für die Warteschlange, in die Nachrichten gestellt werden müssen. Beispiel:  
`Response/JMSTarget`

11. Geben Sie in das Feld **JMS-JNDI-Factory-Name** die folgende Factory ein:  
`com.ibm.websphere.naming.WsnInitialContextFactory`
12. Klicken Sie auf **Speichern**.

## Zielwarteschlange erstellen

Erstellen Sie unter Verwendung der WebSphere-Administrationskonsole eine Zielwarteschlange in WebSphere Application Server:

1. Klicken Sie in der linken Anzeige auf **Systemintegration**.
2. Klicken Sie auf **Busse > SIBUS** (oder auf den Namen für den im vorherigen Abschnitt erstellten Bus).
3. Klicken Sie im Teilfenster **Weitere Merkmale** auf **Ziele**. Klicken Sie auf **Neu**.
4. Wählen Sie das Optionsfeld **Warteschlange** für den Zieltyp aus, und klicken Sie auf **Weiter**.
5. Geben Sie unter **ID** einen Wert ein. Beispiel: Response. Daraufhin wird die Zielwarteschlange auf dem Bus erstellt.

---

## Kapitel 6. Integration von WebSphere Process Server mit SOAP/HTTP

In diesem Kapitel wird die Integration von WebSphere Partner Gateway mit WebSphere Process Server unter Verwendung von SOAP über das HTTP-Transportprotokoll beschrieben. Es enthält Informationen zu den folgenden Themen:

- „SOAP-Nachrichten über das HTTP-Protokoll austauschen“
- „In WebSphere Process Server enthaltene Web-Services aufrufen“ auf Seite 111
- „In externen Partnern enthaltene Web-Services aufrufen“ auf Seite 112

---

### SOAP-Nachrichten über das HTTP-Protokoll austauschen

Wenn ein externer Partner einen Web-Service bei WebSphere Process Server (oder umgekehrt) anfordert, wird die zugeordnete SOAP-Nachricht an einen Empfänger in WebSphere Partner Gateway gesendet. WebSphere Partner Gateway agiert als Proxy und leitet die Anforderung an die URL-Adresse für den Service weiter.

### In WebSphere Process Server enthaltene Web-Services aufrufen

Wenn ein externer Partner einen in WebSphere Process Server enthaltenen Web-Service aufruft, sieht der Ablauf wie folgt aus:

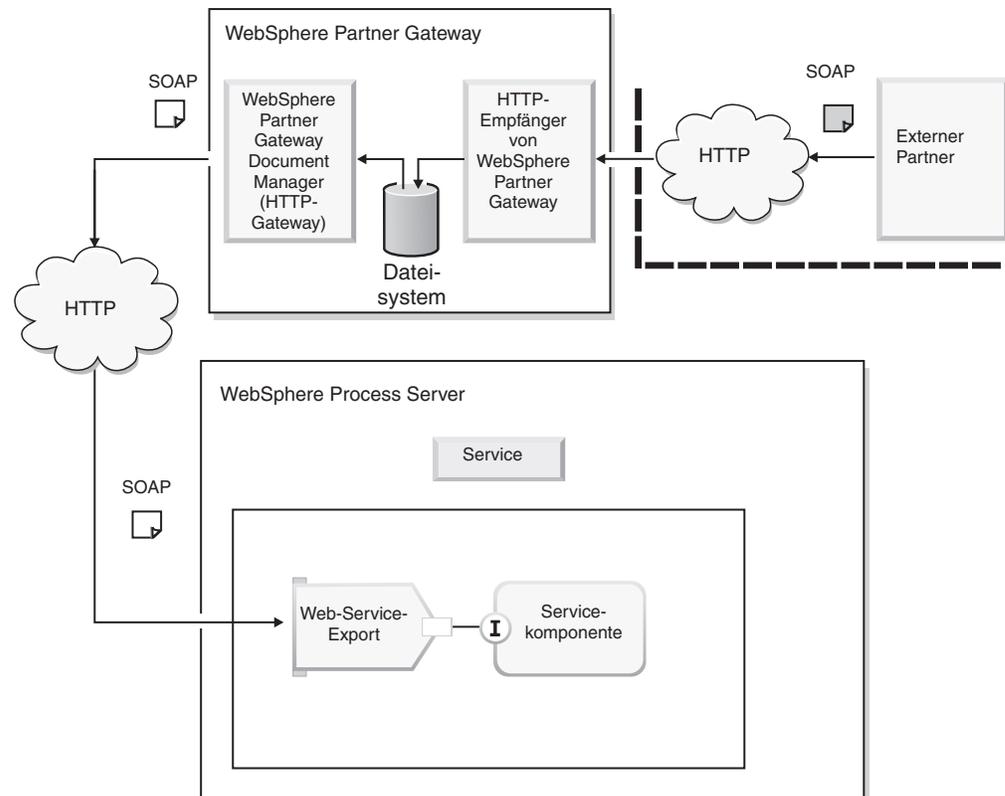


Abbildung 26. Aufruf eines Web-Service durch den externen Partner

WebSphere Partner Gateway leitet die Web-Service-Anforderung an WebSphere Process Server weiter, wo die Web-Service-Exportbindung zum Aufrufen des Service verwendet wird.

Sie müssen Ihrem externen Partner die öffentliche WSDL zur Verfügung stellen, für die als Endpunkt die URL-Adresse von WebSphere Partner Gateway angegeben ist. Informationen zum Angeben dieser URL-Adresse finden Sie im Abschnitt „WebSphere Process Server konfigurieren“ auf Seite 112. WebSphere Partner Gateway agiert als Proxy. Es empfängt eine SOAP-Nachricht vom Partner und ermittelt den entsprechenden privaten Web-Service. Dann ruft es mit Hilfe derselben SOAP-Nachricht den privaten Web-Service in WebSphere Process Server auf. Die von WebSphere Process Server zurückgegebene HTTP-Antwort (sowohl die Antwort auf Transportebene als auch die Web-Service-Antwort) wird dann an den Partner zurückgegeben.

## In Partnern enthaltene Web-Services aufrufen

Wenn WebSphere Process Server einen Web-Service von einem externen Partner aufruft, sieht der Ablauf wie folgt aus:

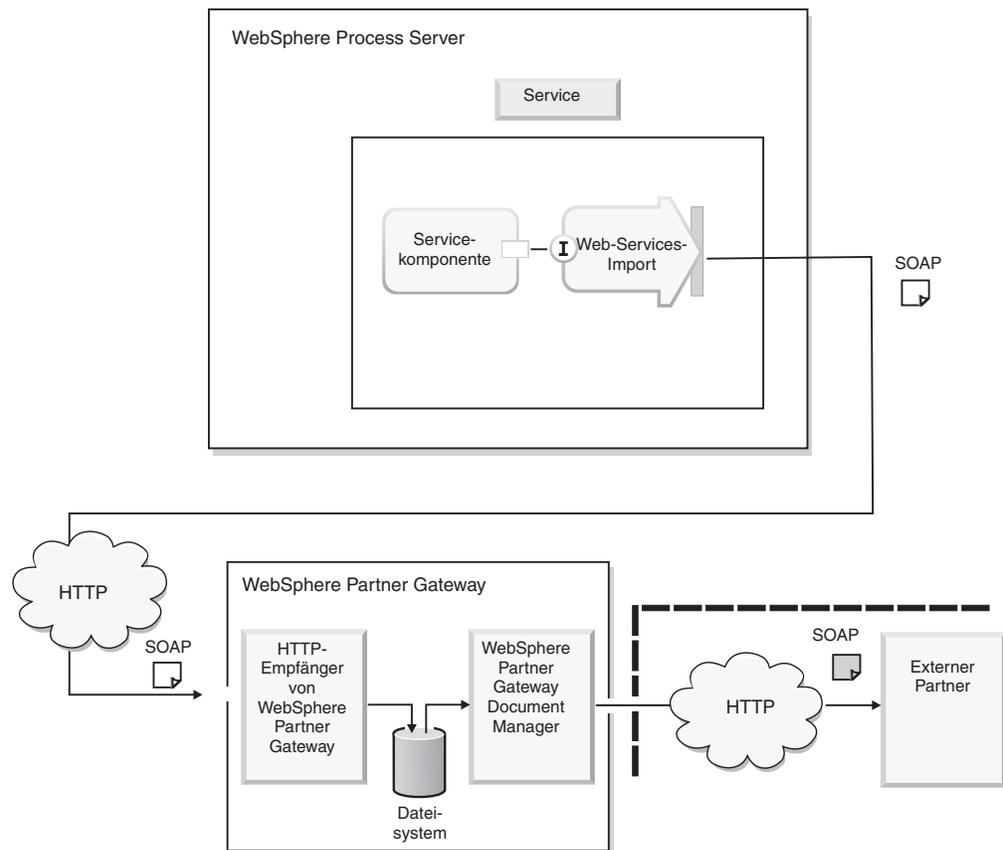


Abbildung 27. Aufruf eines Web-Service durch einen Service von WebSphere Process Server

WebSphere Process Server verwendet zum Aufrufen des Web-Service die Web-Service-Importbindung. WebSphere Partner Gateway leitet die Web-Service-Anforderung von WebSphere Process Server den entsprechenden externen Partner weiter.

WebSphere Partner Gateway macht den Web-Service für WebSphere Process Server über die Web-Service-URL verfügbar, die beim Hochladen des Web-Service in die Community Console angegeben wurde. Darüber hinaus muss der Service in

WebSphere Process Server die URL-Parameter zur Identifizierung des Empfängerpartners bereitstellen. Weitere Informationen finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*. WebSphere Partner Gateway agiert als Proxy. Es empfängt eine SOAP-Nachricht von WebSphere Process Server und ermittelt den entsprechenden Web-Service sowie den Empfängerpartner. Dann ruft es den vom externen Partner bereitgestellten Web-Service mit Hilfe derselben SOAP-Nachricht auf. Die vom externen Partner zurückgegebene HTTP-Antwort (sowohl die Antwort auf Transportebene als auch die Web-Service-Antwort) wird an WebSphere Process Server zurückgegeben.

Die vom externen Partner empfangene Antwort wird über dieselbe HTTP-Verbindung wie die Anforderung an WebSphere Process Server zurückgegeben. Das Verhalten ist bei Web-Services, die nur Anforderungen (request-only) bzw. Anforderungen und Antworten (request-response) verwenden, identisch.

---

## In WebSphere Process Server enthaltene Web-Services aufrufen

In diesem Abschnitt wird beschrieben, wie ein externer Partner einen in WebSphere Process Server enthaltenen Web-Service aufruft.

Ein externer Partner sendet eine Anforderung an einen HTTP-Empfänger auf dem WebSphere Partner Gateway-Hub. WebSphere Partner Gateway authentifiziert den Benutzer, sucht die Verbindung zwischen dem externen und dem internen Partner und leitet die Anforderung an ein beim internen Partner definiertes Ziel weiter.

**Anmerkung:** Wenn der externe Partner die SOAP-Nachricht sendet, muss er sich bei WebSphere Partner Gateway authentifizieren. Der externe Partner kann die HTTP-Basisauthentifizierung verwenden; dazu muss die Geschäfts-ID des Partners sowie Name und Kennwort der Konsole angegeben werden. Alternativ kann der Partner ein SSL-Clientzertifikat vorlegen, das zuvor in WebSphere Partner Gateway definiert wurde. Weitere Informationen zu Zertifikaten finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

## WebSphere Partner Gateway konfigurieren

Befolgen Sie bei der Konfiguration von WebSphere Partner Gateway für Web-Services die Anweisungen im Handbuch *WebSphere Partner Gateway Hubkonfiguration*. Gehen Sie bei der Konfiguration von WebSphere Partner Gateway wie folgt vor:

1. Fügen Sie dem Synchronprüfungs-Konfigurationspunkt des HTTP-Empfängers den Handler `com.ibm.bcg.server.sync.SoapSyncHdlr` hinzu (bei der Verarbeitung von Zweiweganforderungen).
2. Rufen Sie von WebSphere Process Server die WSDL ab, die von der Web-Service-Exportbindung Ihrer Komponente generiert wurde.
3. Stellen Sie sicher, dass das HTTP-Ziel des internen Partners die URL-Adresse angibt, an die WebSphere Partner Gateway Nachrichten für WebSphere Process Server sendet. Das Ziel verweist auf die Endpunkt-URL in der WSDL, die Sie in Schritt 2 hochgeladen haben.

## WebSphere Process Server konfigurieren

Informationen zum Konfigurieren der Komponenten in WebSphere Process Server finden Sie im Information Center für WebSphere Process Server. Wenn Sie einen Export für eine Komponente erstellen, müssen Sie die Option für Web-Service-Bindung (Web-Service-Binding) auswählen.

---

## In externen Partnern enthaltene Web-Services aufrufen

Wenn ein Service in WebSphere Process Server eine SOAP-Anforderung an eine URL-Adresse in WebSphere Partner Gateway stellt, identifiziert WebSphere Partner Gateway den Partner, der die SOAP-Anforderung verarbeiten kann. Dazu wird die von WebSphere Process Server angegebene Basisauthentifizierung für das Partnerprofil verwendet. Wenn die SOAP-Anforderung in zwei Richtungen erfolgt, empfängt WebSphere Partner Gateway die Antwort vom externen Partner und sendet diese an WebSphere Process Server zurück.

### WebSphere Partner Gateway konfigurieren

Befolgen Sie bei der Konfiguration von WebSphere Partner Gateway für Web-Services die Anweisungen im Handbuch *WebSphere Partner Gateway Hubkonfiguration*. Gehen Sie bei der Konfiguration von WebSphere Partner Gateway wie folgt vor:

1. Fügen Sie dem Synchronprüfungs-Konfigurationspunkt des HTTP-Empfängers den Handler `com.ibm.bcg.server.sync.SoapSyncHdlr` hinzu (bei der Verarbeitung von Zweiweganforderungen).
2. Stellen Sie sicher, dass das HTTP-Ziel des externen Partners die URL-Adresse angibt, an die WebSphere Partner Gateway Nachrichten senden soll. Das Ziel verweist auf die Endpunkt-URL in der WSDL, die Sie im Rahmen des Konfigurationsprozesses hochgeladen haben.

### WebSphere Process Server konfigurieren

Informationen zum Konfigurieren der Komponenten in WebSphere Process Server finden Sie im Information Center für WebSphere Process Server. Denken Sie daran, dass Sie die WSDL-Datei, die den Web-Service beschreibt, beim entsprechenden Service-Provider anfordern müssen. Die WSDL-Datei wird in das Modul für die Komponenten importiert.

Der Zugriff auf die von einem externen Partner bereitgestellten Web-Services erfolgt über die URL von WebSphere Partner Gateway. Aus diesem Grund muss WebSphere Process Server den Service über die URL-Adresse des HTTP-Empfängers von WebSphere Partner Gateway aufrufen. Darüber hinaus setzt WebSphere Partner Gateway voraus, dass der interne Partner im Rahmen der HTTP-Basisauthentifizierung einen Benutzernamen und ein Kennwort angibt. Informationen zu Benutzernamen und Kennwörtern finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*. Führen Sie die folgenden Schritte aus, damit WebSphere Process Server die HTTP-Basisauthentifizierung angeben und außerdem den Endpunkt des Web-Service ändern kann, der von WebSphere Process Server aufgerufen wird:

Gehen Sie wie folgt vor, um die Endpunktadresse zu ändern:

1. Melden Sie sich an der Administrationskonsole von WebSphere Process Server an.
2. Navigieren Sie zu den Web-Service-Clientbindungen des Service, den Sie über WebSphere Partner Gateway aufrufen. Verwenden Sie dazu die folgende Pfadangabe: **Enterprise-Anwendungen** > *<ihre\_anwendung>* > **EJB-Module** > *<ihre\_JAR-datei>* > **Web-Services-Clientbindungen**.
3. Geben Sie die überschriebene Endpunkt-URL für den Service-Port an, den Sie über WebSphere Partner Gateway aufrufen möchten. Geben Sie die URL-Adresse des HTTP-Empfängers von WebSphere Partner Gateway an, der die Web-Service-Anforderung von WebSphere Process Server empfängt. Die Endpunkt-URL-Adresse hat das folgende Format:  
`<IP-adresse:portnummer>bcgreceiver/<zielname>`

4. Klicken Sie auf **Anwenden**, und klicken Sie dann auf **OK**.

Führen Sie die folgenden Schritte aus, um den Benutzernamen und das Kennwort für die Basisauthentifizierung anzugeben:

1. Melden Sie sich an der Administrationskonsole von WebSphere Process Server an.
2. Navigieren Sie zu den Web-Service-Clientbindungen des Service, den Sie über WebSphere Partner Gateway aufrufen. Verwenden Sie dazu die folgende Pfadangabe: **Enterprise-Anwendungen** > *<ihre\_anwendung>* > **EJB-Module** > *<ihre\_JAR-dateien>* > **Web-Services: Clientsicherheitsbindungen** > **HTTP-Basisauthentifizierung**.
3. Geben Sie die ID und das Kennwort für die Basisauthentifizierung an.

Die Authentifizierung-ID hat das folgende Format:

*Geschäfts-ID/benutzername*

Beispiel: Wenn die Geschäfts-ID des internen Partners 987654321 und der Benutzername 'admin' ist, sieht die Authentifizierung-ID wie folgt aus:

987654321/admin

4. Klicken Sie auf **Anwenden** und dann auf **OK**.



---

## Kapitel 7. Integration von WebSphere Process Server mit Dateisystem als Transportprotokoll

In diesem Kapitel wird die Integration von WebSphere Partner Gateway mit WebSphere Process Server unter Verwendung des Dateisystemprotokolls beschrieben.

Über das Dateisystemprotokoll kann WebSphere Partner Gateway Nachrichten senden, indem es sie in einer definierten Verzeichnisstruktur ablegt. WebSphere Partner Gateway empfängt Nachrichten, indem es sie aus der Verzeichnisstruktur liest. Das Dateisystemprotokoll unterstützt den Pakettyp 'None'.

Dieses Kapitel enthält Informationen zu den folgenden Themen:

- „Dokumentenversand mit dem Dateisystemprotokoll“
- „Dokumentenempfang mit dem Dateisystemprotokoll“
- „WebSphere Process Server-Umgebung einrichten“
- „WebSphere Partner Gateway-Umgebung einrichten“ auf Seite 117

---

### Dokumentenversand mit dem Dateisystemprotokoll

Die WebSphere Process Server-Komponenten nutzen die SCA-J2C-Importbindung (SCA - Service Component Architecture) zum Senden von Geschäftsdaten an WebSphere Adapter for Flat Files. Der Adapter schreibt die Geschäftsdaten in ein Verzeichnis auf dem Dateisystem. Der Dateisystemempfänger von WebSphere Partner Gateway liest dieses Geschäftsdokument aus dem Verzeichnis auf dem Dateisystem und leitet es an die entsprechenden Handelspartner weiter.

---

### Dokumentenempfang mit dem Dateisystemprotokoll

WebSphere Partner Gateway empfängt ein Geschäftsdokument vom Handelspartner. Der Document Manager von WebSphere Partner Gateway schreibt das Dokument in das Dateisystemverzeichnis. WebSphere Adapter for Flat Files fragt Ereignisse aus diesem Verzeichnis ab und liest das Geschäftsdokument. Dann wird der SCA-Serviceexport aufgerufen, der Ereignisse von diesem Adapter erwartet. Der SCA-Service empfängt das Geschäftsdokument.

**Anmerkung:** WebSphere Adapter for Flat Files erwartet eine Zeichenfolge oder Rohdaten, die aus einer Datei gelesen bzw. in eine Datei geschrieben werden können. Deshalb muss die WebSphere Process Server-Umgebung das Geschäftsobjekt entweder in eine Zeichenfolge oder in Rohdaten serialisieren. Die WebSphere Process Server-Services müssen eine Komponente erstellen, die ein Geschäftsobjekt in ein Geschäftsdokument (oder umgekehrt) serialisieren.

---

### WebSphere Process Server-Umgebung einrichten

In diesem Abschnitt werden die erforderlichen Schritte zur Einrichtung der WebSphere Process Server-Umgebung für den Dateisystemtransport in WebSphere Application Server beschrieben. Dazu gehört das Erstellen und Konfigurieren der WebSphere Process Server-Artefakte.

Dieser Abschnitt enthält Informationen zu den folgenden Themen:

- „WebSphere Adapter for Flat Files implementieren und konfigurieren“
- „SCA-Komponenten erstellen“ auf Seite 117

## **WebSphere Adapter for Flat Files implementieren und konfigurieren**

WebSphere Process Server nutzt für dateisystembasierte Integration WebSphere Adapter for Flat Files, einen bidirektionalen Adapter, der Geschäftsdaten aus dem Dateisystem lesen und in das Dateisystem schreiben kann.

Für eingehende Kommunikation unterstützt der Adapter die Lesefunktion.

Für ausgehende Kommunikation unterstützt der Adapter die folgenden Funktionen:

- Erstellen
- Anhängen
- Löschen
- Überschreiben
- Abrufen
- Liste
- Vorhanden

Die WebSphere Adapter for Flat Files-Installation beinhaltet eine RAR-Datei (RAR - Resource Adapter Archive), die in WebSphere Process Server implementiert werden kann. Die RAR-Datei enthält die Dateien, die im Lieferumfang des Adapters enthalten sind. Vor der Implementierung von WebSphere Adapter für Flachdateien müssen Sie die RAR-Datei des Adapters importieren und das Projekt in WebSphere Integration Developer erstellen.

Die folgenden Abschnitte enthalten eine Übersicht über die Schritte, die zum Implementieren und Konfigurieren von WebSphere Adapter für Flachdateien erforderlich sind. Detaillierte Informationen hierzu finden Sie in der WebSphere Integration Developer-Dokumentation.

### **RAR-Datei importieren:**

1. Wechseln Sie in die J2EE-Perspektive, und importieren Sie die RAR-Datei.
2. Geben Sie die Position für den Import der RAR-Datei (entspricht der Position, in die Sie die Adapterdateien während der Installation kopiert haben) und einen Projektnamen an.

Daraufhin wird ein neues J2EE-Connectorprojekt im Arbeitsbereich erstellt.

### **Externe Abhängigkeiten zum Projekt hinzufügen:**

Kopieren Sie die externen Abhängigkeiten in das Connectormodul in Ihrem WebSphere Integration Developer-Projekt, und fügen Sie sie dem Projekt als interne Bibliotheken hinzu. Dies ist erforderlich, um die Abhängigkeiten in der EAR-Datei zu bündeln, die exportiert wird. Detaillierte Informationen hierzu finden Sie in der WebSphere Integration Developer-Dokumentation.

### **Adapter konfigurieren:**

Für den Konfigurationsprozess wird der Enterprise Service Discovery-Assistent in WebSphere Integration Developer verwendet. Dieser Prozess ermöglicht die Eingabe aller erforderlichen Informationen für die erstmalige Konfiguration des Adapters. Die Ausgabe des Enterprise Service Discovery-Assistenten wird in einem Geschäftsintegrationsmodul gespeichert, das die Geschäftsobjekte, die Importdatei (beschreibt die ausgehende Verarbeitung gemäß Aktivierungsspezifikation), die Exportdatei (beschreibt die eingehende Ereignisverarbeitung gemäß Interaktionsspezifikation) und die WSDL-Datei enthält. Daraus ergibt sich eine Komponente in Ihrem Modul, mit es Ihnen ermöglicht, aus dem Dateisystem zu lesen bzw. in das Dateisystem zu schreiben.

## **SCA-Komponenten erstellen**

Soll die SCA-Komponente Geschäftsdokumente unter Verwendung des Dateisystemprotokolls von WebSphere Partner Gateway empfangen, können Sie mit Hilfe des Enterprise Service Discovery-Tools die erforderlichen Importbindungen für WebSphere Adapter for Flat Files generieren. Zum Generieren der erforderlichen Importbindungen müssen Sie den Servicetyp im Enterprise Service Discovery-Assistenten als 'Eingehend' definieren. Der Methodenname für eingehende Dokumente ist READ. Verbinden Sie die Exportbindung der SCA-Komponente mit der vom Enterprise Service Discovery-Tool generierten Importbindung.

Soll die SCA-Komponente Geschäftsdokumente unter Verwendung des Dateisystemprotokolls an WebSphere Partner Gateway senden, können Sie mit Hilfe des Enterprise Service Discovery-Tools die erforderlichen Exportbindungen für WebSphere Adapter for Flat Files generieren. Zum Generieren der erforderlichen Importbindungen müssen Sie den Servicetyp im Enterprise Service Discovery-Assistenten als 'Ausgehend' definieren. Verbinden Sie die vom Enterprise Service Discovery-Tool generierte Exportbindung mit der Importbindung der SCA-Komponente.

---

## **WebSphere Partner Gateway-Umgebung einrichten**

In diesem Abschnitt werden die erforderlichen Schritte zur Einrichtung der WebSphere Partner Server-Umgebung für den Dateisystemtransport in WebSphere Application Server beschrieben.

Verwendung der Community Console von WebSphere Partner Gateway:

1. Erstellen Sie einen Dateisystemempfänger auf dem Hub, um Dokumente zu empfangen, die von WebSphere Process Server oder von externen Partnern an den Hub gesendet werden.
2. Erstellen Sie gegebenenfalls ein Profil für den internen Partner, das ein Dateisystemziel enthält, das WebSphere Partner Gateway zum Senden von Dokumenten an WebSphere Process Server verwendet.
3. Erstellen Sie Profile für externe Partner, die Dateisystemziele enthalten, die WebSphere Partner Gateway zum Senden von Dokumenten an die Partner verwendet.
4. Importieren Sie WSDL-Dateien, Transformationszuordnungen, RosettaNet-Pakete oder andere Dokumentdefinitionsmechanismen, damit eine Dokumentdefinition für den ausgetauschten Dokumenttyp auf der Seite für Dokumentenflussdefinitionen der Community Console von WebSphere Partner Gateway angezeigt wird.

5. Erstellen Sie Interaktionen zwischen den Dokumenttypen, die der Hub von WebSphere Process Server oder einem externen Partner empfängt, und den Dokumenttypen, die der Hub an WebSphere Process Server oder einen externen Partner sendet.
6. Erstellen Sie im Profil des internen Partners und in den Profilen der externen Partner B2B-Funktionalität, um die Dokumenttypen anzugeben, die gesendet und empfangen werden können.
7. Erstellen Sie Interaktionen zwischen den Dokumenttypen, die der Hub von WebSphere Process Server oder einem externen Partner empfängt, und den Dokumenttypen, die der Hub an WebSphere Process Server oder einen externen Partner sendet.
8. Erstellen Sie Partnerverbindungen zwischen dem internen Partner und den externen Partnern, um den Quellenpartner (Absender des Dokuments), den Zielpartner (Empfänger des Dokuments) und die Aktion anzugeben, die der Hub gegebenenfalls zum Transformieren des Dokuments ausführen soll.

---

## **Teil 3. Integration mit WebSphere InterChange Server**

In den folgenden Kapiteln wird die Vorgehensweise für die Integration von WebSphere Partner Gateway mit WebSphere InterChange Server beschrieben.



---

## Kapitel 8. Einführung zur InterChange Server-Integration

In diesem Kapitel wird die Integration von WebSphere Partner Gateway mit WebSphere InterChange Server beschrieben.

### Anmerkungen:

1. Eine Beschreibung des allgemeinen Verfahrens zur Integration von WebSphere Partner Gateway mit einem Back-End-System finden Sie in Kapitel 2, „Back-End-Integration planen“, auf Seite 9.
2. Für die Informationen dieses Kapitels wird vorausgesetzt, dass Sie mit WebSphere InterChange Server und den zugehörigen Komponenten, wie Collaborations, Geschäftsobjekten und Adaptern vertraut sind.

Die Integration von WebSphere Partner Gateway mit einem Back-End-System wird häufig von zwei verschiedenen Personen ausgeführt oder in zwei verschiedene Aufgabenbereiche (Rollen) untergliedert. Jede Rolle ist hierbei für die Konfiguration einer bestimmten Komponente zuständig, für die der entsprechende Mitarbeiter Experte ist. Aus diesem Grund wird in diesem Kapitel die Integration mit WebSphere InterChange Server in die Konfiguration von WebSphere Partner Gateway und die Konfiguration von InterChange Server unterteilt. In Tabelle 35 sind diese Konfigurationsrollen sowie die Abschnitte innerhalb des Kapitels aufgeführt, in denen Sie die zugehörigen Konfigurationsinformationen finden.

*Tabelle 35. Rollen bei der InterChange Server-Integration*

Konfigurationsrolle	Weitere Informationen
Konfiguration von WebSphere Partner Gateway	<ol style="list-style-type: none"><li>1. „Integration mit InterChange Server planen“ auf Seite 122.</li><li>2. „WebSphere Partner Gateway für die Verwendung mit InterChange Server konfigurieren“ auf Seite 125.</li></ol>
Konfiguration von WebSphere InterChange Server	<ol style="list-style-type: none"><li>1. „Integration mit InterChange Server planen“ auf Seite 122.</li><li>2. „InterChange Server konfigurieren“ auf Seite 128.</li></ol>

**Anmerkung:** Diese Konfigurationsrollen können getrennt voneinander ausgeführt werden. Allerdings werden für sie auch allgemeine Informationen benötigt, um den Datenaustausch zwischen den beiden Komponenten zu ermöglichen.

Dieses Kapitel enthält Informationen zu den folgenden Themen:

- „Integration mit InterChange Server planen“ auf Seite 122
- „WebSphere Partner Gateway für die Verwendung mit InterChange Server konfigurieren“ auf Seite 125
- „InterChange Server konfigurieren“ auf Seite 128
- „Dokumente mit Anhängen verarbeiten“ auf Seite 133

## Integration mit InterChange Server planen

Zur Planung der Integration mit WebSphere InterChange Server befolgen Sie die in Kapitel 2, „Back-End-Integration planen“, auf Seite 9 aufgeführten Schritte. In Tabelle 36 sind die Schritte zur Integration von WebSphere Partner Gateway mit InterChange Server (ICS) zusammengefasst.

Tabelle 36. Planen der Integration mit WebSphere InterChange Server

Integrationschritt	Weitere Informationen
1. Vergewissern Sie sich, dass Sie eine unterstützte Version von WebSphere InterChange Server installiert und für WebSphere Partner Gateway verfügbar haben.	Kapitel 3: „Von WebSphere Partner Gateway unterstützte InterChange Server-Versionen“
2. Bestimmen Sie das Geschäftsprotokoll des WebSphere Partner Gateway-Dokuments.	Kapitel 2: „Zu verwendendes Geschäftsprotokoll“ auf Seite 9
3. Bestimmen Sie den Pakettyp für das Dokument: 'None' oder 'Backend Integration'.	Kapitel 2: „Zu verwendender Pakettyp“ auf Seite 25
4. Bestimmen Sie das zwischen WebSphere Partner Gateway und WebSphere InterChange Server zu verwendende Transportprotokoll.	Kapitel 3: „Von InterChange Server unterstützte Nachrichtentransporte“ auf Seite 123
5. Konfigurieren Sie WebSphere Partner Gateway.	Kapitel 3: „WebSphere Partner Gateway für die Verwendung mit InterChange Server konfigurieren“ auf Seite 125
6. Konfigurieren Sie WebSphere InterChange Server-Komponenten zur Verwendung über das ausgewählte Transportprotokoll.	Kapitel 3: „InterChange Server konfigurieren“ auf Seite 128

## Von WebSphere Partner Gateway unterstützte InterChange Server-Versionen

WebSphere Partner Gateway Version 6.1 unterstützt die Integration mit folgenden Versionen von InterChange Server:

- 4.3.0

InterChange Server ist für verschiedene Plattformen verfügbar, zu denen Windows 2000 und mehrere UNIX-basierte Plattformen zählen. Weitere Informationen finden Sie im Installationshandbuch für InterChange Server in der Dokumentation zu WebSphere InterChange Server.

Zur Integration von WebSphere Partner Gateway mit Interchange Server werden WBI-Adapter verwendet. Die folgenden Adapterversionen werden unterstützt:

- WBIA HTTP Adapter Version 1.3
- WBIA JMS Adapter Version 2.8.x
- WBIA-Web-Services-Adapter Version 3.4.x
- 

In Verbindung mit den Adaptern werden WBI-Data-Handler verwendet. Zur Verarbeitung von XML-Nachrichten müssen Sie sicherstellen, dass Sie den WebSphere Business Integration-Data-Handler für XML der Version 2.3.1 oder höher verwenden. Für cXML-Nachrichten müssen Sie den Data-Handler für XML der Version 2.4.1 oder höher verwenden.

## Von InterChange Server unterstützte Nachrichtentransporte

Wenn WebSphere Partner Gateway eine Nachricht an InterChange Server über ein bestimmtes Nachrichtentransportprotokoll sendet, sendet es diese Nachricht an den entsprechenden Adapter, der dieses Nachrichtenprotokoll verarbeiten kann und die Nachricht an InterChange Server weiterleitet. Wenn InterChange Server eine Nachricht an WebSphere Partner Gateway sendet, sendet es die Nachricht analog über das entsprechende Transportprotokoll an den Adapter, der die Nachricht an WebSphere Partner Gateway weiterleitet.

Der Adapter arbeitet mit einem Data-Handler, der die eigentliche Konvertierung von serialisierten Daten in Geschäftsobjekte bzw. von Geschäftsobjekten in serialisierte Daten ausführt. Diese Konvertierungen werden von dem für den Typ der Nutzdaten (payload) geeigneten Data-Handler ausgeführt.

Die folgenden beiden Nachrichtentransportprotokolle werden von InterChange Server unterstützt:

- HTTP-Transportprotokoll

**Anmerkung:** Der Austausch von Web-Services über HTTP wird in einem separaten Abschnitt behandelt, da Web-Services in einer Weise ausgetauscht werden, die sich von anderen über HTTP übermittelten Dokumenten unterscheidet. Siehe „SOAP-Dokumente über HTTP/S senden“ auf Seite 169.

- JMS-Transportprotokoll

**Anmerkung:** InterChange Server stellt andere Typen von Integrationsoptionen zur Verfügung, wie zum Beispiel eine dateibasierte Integration. Detaillierte Informationen zur Aktivierung des Dokumentaustauschs durch dateibasierte Integration finden Sie in der Dokumentation zu WebSphere InterChange Server.

Verwenden Sie das am besten für die Anforderungen Ihres Geschäfts geeignete Transportprotokoll. Beachten Sie die folgenden Punkte:

- Vor allem müssen Sie sicherstellen, dass das Transportprotokoll, das Sie zwischen dem externen Partner und WebSphere Partner Gateway verwenden, mit dem genutzten Integrationsmechanismus verfügbar ist. Siehe „Zu verwendendes Nachrichtentransportprotokoll“ auf Seite 37.
- Zum Senden von SOAP-Dokumenten an WebSphere InterChange Server und zum Empfangen von SOAP-Dokumenten von WebSphere InterChange Server ist die Verwendung des HTTP-Transportprotokolls erforderlich. Weitere Informationen hierzu finden Sie im Abschnitt „SOAP-Dokumente über HTTP/S senden“ auf Seite 169.

### HTTP

Welche Adapter zum Senden und Empfangen von Dokumenten zwischen WebSphere Partner Gateway und InterChange Server über HTTP erforderlich sind, hängt von folgenden Faktoren ab:

- Typ des gesendeten Dokuments
- Version von InterChange Server mit der die Integration erfolgt

**Anmerkung:** Alle Verweise auf das HTTP-Transportprotokoll sind ebenso auf HTTPS anwendbar.

Tabelle 37 enthält eine Übersicht, wo Sie Informationen zur Konfiguration von Adaptern zur Verwendung mit InterChange Server finden.

Tabelle 37. Konfigurieren für HTTP-Transport mit InterChange Server

Bedingung	Weitere Informationen
Wenn Sie Nicht-SOAP-Dokumente mit InterChange Server übertragen möchten.	„HTTP-Transportprotokoll mit ICS verwenden“ auf Seite 153
Wenn Sie SOAP-Dokumente senden möchten.	„SOAP-Dokumente über HTTP/S senden“ auf Seite 169

## JMS

Die Komponenten, die zum Senden und Empfangen von Dokumenten zwischen WebSphere Partner Gateway und InterChange Server über JMS benötigt werden, sind in Tabelle 63 auf Seite 174 zusammengefasst. Grundsätzlich basiert die Unterstützung für JMS auf der Verwendung des WebSphere Business Integration Adapter für JMS. Der Adapter für JMS ruft Collaborations in InterChange Server asynchron auf.

## Vorteile der unterschiedlichen Transportmethoden

Beachten Sie bei der Entscheidung, welchen Nachrichtentransport Sie mit InterChange Server verwenden, die folgenden Vorteile der unterschiedlichen Methoden:

- Der Adapter für HTTP unterstützt synchrone Kommunikation. Wenn Sie synchrone Transaktionen benötigen, *müssen* Sie das HTTP-Transportprotokoll verwenden.
- Der Adapter für JMS kann eine *garantierte Ereignisübermittlung* von WebSphere Partner Gateway an WebSphere InterChange Server bereitstellen.  
Die garantierte Ereignisübermittlung stellt sicher, dass keine Ereignisse verloren gehen oder zweimal gesendet werden.

## Unterstützung für die InterChange Server-Integration

WebSphere Partner Gateway stellt Muster bereit, die Sie beim Integrationsprozess mit InterChange Server unterstützen. Diese Muster befinden sich im folgenden Verzeichnis des Produktverzeichnisses von WebSphere Partner Gateway:

Integration/WBI/WICS/samples

In Tabelle 38 sind die Unterverzeichnisse des Verzeichnisses `samples` für die verschiedenen Transportprotokolle aufgeführt, die von InterChange Server unterstützt werden.

Tabelle 38. Muster für die InterChange Server-Integration

Transportprotokoll	InterChange Server-Version	Samples-Unterverzeichnis
HTTP	Alle unterstützten Versionen	Allgemeine Muster: HTTP RosettaNet-spezifische Muster: RosettaNet/HTTP
JMS	Alle unterstützten Versionen	Allgemeine Muster: JMS RosettaNet-spezifische Muster: RosettaNet/JMS

---

## WebSphere Partner Gateway für die Verwendung mit InterChange Server konfigurieren

Eine allgemeine Übersicht zur Konfiguration von WebSphere Partner Gateway für die Kommunikation mit einem Back-End-System finden Sie im Abschnitt „WebSphere Partner Gateway konfigurieren“ auf Seite 47. In diesem Abschnitt werden die Schritte zusammengefasst, die zur Konfiguration von WebSphere Partner Gateway zur Kommunikation mit InterChange Server erforderlich sind.

Die Konfiguration von WebSphere Partner Gateway erfolgt in folgenden Schritten:

- Unterstützung für ausgehende Dokumente konfigurieren  
Informationen zum Senden von Dokumenten von WebSphere Partner Gateway an InterChange Server finden Sie im Abschnitt „Unterstützung für ausgehende Dokumente bereitstellen“.
- Unterstützung für eingehende Dokumente konfigurieren  
Informationen zum Senden von Dokumenten von InterChange Server an WebSphere Partner Gateway finden Sie im Abschnitt „Unterstützung für eingehende Dokumente bereitstellen“ auf Seite 126.

### Unterstützung für ausgehende Dokumente bereitstellen

Damit WebSphere Partner Gateway Dokumente an beliebige Back-End-Systeme senden kann, müssen Sie die im Abschnitt „Zieleinheit für Partnerdokument definieren“ auf Seite 49 beschriebenen Schritte ausführen. Wenn es sich bei Ihrem Back-End-System um InterChange Server handelt, müssen Sie ein Ziel erstellen, dessen Transporttyp mit dem Transportprotokoll übereinstimmt, das für Nachrichten zwischen WebSphere Partner Gateway und InterChange Server verwendet wird. Wenn der Hub ein Dokument an InterChange Server sendet, muss er wissen, wohin das Dokument weiterzuleiten ist. Diese Position muss dem verwendeten Transportprotokoll entsprechen. Das Transportprotokoll muss von InterChange Server unterstützt werden (siehe „Von InterChange Server unterstützte Nachrichtentransporte“ auf Seite 123).

In den folgenden Abschnitten wird die Erstellung von Zielen für folgende, von InterChange Server unterstützte Transportprotokolle zusammengefasst:

- „Dokumentenversand über das HTTP-Transportprotokoll an ICS konfigurieren“
- „Dokumentenversand über das JMS-Transportprotokoll an ICS konfigurieren“ auf Seite 126

### Dokumentenversand über das HTTP-Transportprotokoll an ICS konfigurieren

Wenn der Hub ein Dokument über das HTTP-Protokoll an InterChange Server sendet, leitet der Hub die Nachricht über das definierte Ziel weiter. Dieses Ziel gibt die URL-Adresse an, über die das Dokument von InterChange Server empfangen werden kann. Wenn InterChange Server mit dem HTTP-Protokoll arbeitet, empfängt ein Adapter das Dokument an der entsprechenden URL-Adresse und kann es dann an InterChange Server senden.

Damit der Hub Dokumente über ein Ziel mit dem HTTP-Transportprotokoll senden kann, müssen Sie auf der Seite **Zieldetails** der Community Console ein Ziel erstellen. Dieses Ziel muss zur Verwendung des Transportprotokolls HTTP 1.1 und zum Schreiben an die URL-Adresse, für die der entsprechende Adapter empfangsbereit ist, konfiguriert werden.

**Anmerkung:** Eine Übersicht zur Erstellung eines Ziels finden Sie im Abschnitt „Zieleinheit für Partnerdokument definieren“ auf Seite 49.

## Dokumentenversand über das JMS-Transportprotokoll an ICS konfigurieren

Wenn der Hub ein Dokument über das JMS-Protokoll an InterChange Server sendet, leitet der Hub die Nachricht an die entsprechende JMS-Warteschlange weiter, aus der sie von InterChange Server abgerufen werden kann. Damit der Hub auf diese JMS-Position zugreifen kann, müssen Sie in WebSphere Partner Gateway ein Ziel erstellen, das das JMS-Transportprotokoll verwendet. Dieses Ziel muss zum Schreiben an die Warteschlange konfiguriert werden, die vom Adapter für JMS abgefragt wird.

**Anmerkung:** Eine Übersicht zur Erstellung eines Ziels finden Sie im Abschnitt „Zieleinheit für Partnerdokument definieren“ auf Seite 49.

Damit der Hub Dokumente über ein Ziel mit dem JMS-Transportprotokoll senden kann, erstellen Sie auf der Seite **Zieldetails** der Community Console ein Ziel. Wenn Sie als JMS-Provider WebSphere MQ Version 6.0 mit dem neuesten Fixpack verwenden, sollten Sie die detaillierten Anweisungen zu diesem Thema im Handbuch *WebSphere Partner Gateway Hubkonfiguration* lesen. Geben Sie darüber hinaus die in Tabelle 39 aufgeführten Informationen für das JMS-Protokoll auf der Seite **Zieldetails** ein.

Tabelle 39. JMS-Werte für die Seite 'Zieldetails' zur Kommunikation mit ICS

Feld der Zieldetails	Wert	Anmerkungen und Einschränkungen
JMS-Nachrichtenklasse	TextMessage, BytesMessage oder StreamMessage	
JMS-Warteschlangenname	Der gleiche JMS-Warteschlangenname wie für die Eingabewarteschlange des Adapters für JMS	Diese Warteschlange muss in der Liste der Eingabewarteschlangen des Adapters für JMS enthalten sein. Das heißt, der Adapter muss diese Warteschlange nach eingehenden Ereignissen abfragen. Weitere Informationen hierzu finden Sie im Abschnitt „JMS-Warteschlangen angeben“ auf Seite 181.

## Unterstützung für eingehende Dokumente bereitstellen

Damit WebSphere Partner Gateway Nachrichten von einem beliebigen Back-End-System empfangen kann, müssen Sie die im Abschnitt „Abrufposition für Back-End-Dokumente definieren“ auf Seite 53 beschriebenen Schritte ausführen. Wenn InterChange Server Ihr Back-End-System ist, müssen Sie die folgenden Schritte in Ihrem Hub ausführen:

1. Definieren Sie im Partnerprofil für den internen Partner den Zieltyp, und geben Sie die zugehörige IP-Adresse an, für die der Empfänger empfangsbereit ist.
2. Erstellen Sie einen Empfänger, dessen Transporttyp mit dem Transportprotokoll übereinstimmt, das für Dokumente zwischen WebSphere Partner Gateway und InterChange Server verwendet wird.

Damit der Hub ein Dokument von InterChange Server empfangen kann, muss er die Position kennen, an der die Nachrichten abzurufen sind. Diese Position muss dem zu verwendenden Transportprotokoll entsprechen.

Das Transportprotokoll muss von InterChange Server unterstützt werden (siehe „Von InterChange Server unterstützte Nachrichtentransporte“ auf Seite 123).

In den folgenden Abschnitten wird die Erstellung von Empfängern für Transportprotokolle zusammengefasst, die von InterChange Server unterstützt werden.

### **Dokumentenempfang von ICS über das HTTP-Transportprotokoll konfigurieren**

Wenn der Hub ein Dokument über das HTTP-Transportprotokoll empfängt, ruft der Empfänger das Dokument aus dem definierten Empfänger ab. Dieser Empfänger gibt die URL-Adresse an, an der die Empfängerkomponente für Dokumente von InterChange Server empfangsbereit ist. Wenn InterChange Server mit dem HTTP-Transportprotokoll arbeitet, sendet ein Adapter das Dokument an die entsprechende URL-Adresse, an der es vom Hub empfangen werden kann.

Damit der Hub Dokumente über einen Empfänger mit dem HTTP-Transportprotokoll empfangen kann, müssen Sie auf der Seite **Empfängerliste** der Community Console ein Ziel erstellen. Dieser Empfänger muss das Transportprotokoll HTTP 1.1 verwenden. Der Hub bestimmt diese URL-Adresse als Kombination aus folgenden Informationen:

- Die IP-Adresse des Host-Computers, die aus dem Profil für den internen Partner ermittelt wird
- Die Empfänger-URL-Adresse, die aus dem URL-Feld der Empfängerdefinition ermittelt wird

**Anmerkung:** Eine Übersicht zur Erstellung eines Empfängers finden Sie im Abschnitt „Abrufposition für Back-End-Dokumente definieren“ auf Seite 53.

Damit InterChange Server Dokumente an diesen Empfänger senden kann, muss der zugehörige Adapter zum Senden von Dokumenten an diese URL-Adresse konfiguriert werden. Daher müssen Sie sicherstellen, dass diese Empfänger-URL für die InterChange Server-Konfiguration verfügbar ist.

### **Dokumentenempfang von ICS über das JMS-Transportprotokoll konfigurieren**

Wenn der Hub ein Dokument von InterChange Server über das JMS-Protokoll empfängt, ruft der Hub das Dokument aus der entsprechenden JMS-Warteschlange ab, an die es von InterChange Server gesendet wurde. Damit der Hub auf diese JMS-Position zugreifen kann, müssen Sie in WebSphere Partner Gateway einen Empfänger erstellen, der das JMS-Transportprotokoll verwendet. Über den Empfänger ist der Hub für alle Dokumente in seiner Eingabewarteschlange empfangsbereit und ruft sie ab.

**Anmerkung:** Eine Übersicht zur Erstellung eines Empfängers finden Sie im Abschnitt „Abrufposition für Back-End-Dokumente definieren“ auf Seite 53.

Damit der Hub Dokumente über einen Empfänger mit dem JMS-Transportprotokoll empfangen kann, müssen Sie auf der Seite **Empfängerliste** der Community Console einen Empfänger erstellen. Wenn Sie als JMS-Provider WebSphere MQ Version 6.0 mit dem neuesten Fixpack verwenden, sollten Sie die detaillierten Anweisungen zu diesem Thema im Handbuch *WebSphere Partner Gateway Hub-konfiguration* lesen. Geben Sie darüber hinaus die in Tabelle 40 aufgeführten Informationen für das JMS-Protokoll auf der Seite **Empfängerdetails** ein.

Tabelle 40. JMS-Werte für die Seite 'Empfängerdetails' zur Kommunikation mit ICS

Feld der Empfängerdetails	Wert	Anmerkungen und Einschränkungen
JMS-Nachrichtenklasse	TextMessage	
JMS-Warteschlangenname	Der gleiche JMS-Warteschlangenname wie für die Ausgabewarteschlange des Adapters für JMS	Diese Warteschlange muss als Ausgabewarteschlange des Adapters für JMS aufgelistet sein. Das heißt, der Adapter muss Dokumente an diese Warteschlange senden. Weitere Informationen hierzu finden Sie im Abschnitt „JMS-Warteschlangen angeben“ auf Seite 181.

## InterChange Server konfigurieren

Für die Interaktionen zwischen WebSphere Partner Gateway und InterChange Server müssen Sie eine Integrationskomponentenbibliothek (Integration Component Library - ICL) im Tool 'System Manager' erstellen. Diese ICL enthält die folgenden Artefakte:

- Geschäftsobjektdefinitionen
- Connectorobjekte
- Collaboration-Schablonen und Collaboration-Objekte

Darüber hinaus müssen Sie auch ein Benutzerprodukt erstellen und aus der ICL die Artefakte auswählen, die für Ihre spezielle Interaktion zwischen InterChange Server und WebSphere Partner Gateway erforderlich sind.

**Anmerkung:** Weitere Informationen zur Erstellung von ICLs und zur Konfiguration von InterChange Server finden Sie im Handbuch *System Implementation Guide* in der Dokumentation zu WebSphere InterChange Server.

### Geschäftsobjektdefinitionen erstellen

WebSphere Partner Gateway sendet Ihre Nachricht an einen Adapter, von dem sie in Form von mindestens einem **Geschäftsobjekt** an InterChange Server weitergeleitet wird. Damit InterChange Server ein Geschäftsobjekt erkennen kann, muss er zunächst eine Schablone lokalisieren, die als *Geschäftsobjektdefinition* bezeichnet wird und die Struktur der in diesem Geschäftsobjekt enthaltenen Informationen beschreibt. Jede Einzelinformation in einer Geschäftsobjektdefinition wird in einem *Attribut* untergebracht. Daher müssen Sie Geschäftsobjektdefinitionen erstellen, um die Informationen in Ihrer Nachricht darzustellen. Zur Erstellung von Geschäftsobjektdefinitionen verwenden Sie das Tool 'Business Object Designer'.

**Anmerkung:** Das Tool 'Business Object Designer' ist Teil der beiden Produkte WebSphere InterChange Server und WebSphere Business Integration Adapter. Weitere Informationen zur Verwendung dieses Tools finden Sie im Handbuch *Business Object Development Guide*.

InterChange Server verwendet Geschäftsobjekte für die folgenden Informationen:

- „Geschäftsobjekt für das Dokument“ auf Seite 129
- „Geschäftsobjekte für Konfigurationsdaten“ auf Seite 131

## Geschäftsobjekt für das Dokument

Zur Aufnahme der Nutzdaten des WebSphere Partner Gateway-Dokuments oder der Nachricht müssen Sie eine Geschäftsobjektdefinition definieren, die das **Geschäftsobjekt für Nutzdaten** darstellt. Das Geschäftsobjekt für Nutzdaten ist die Form, in der der Adapter das Dokument an (oder von) InterChange Server überträgt. Dieser Abschnitt enthält die folgenden Informationen zum Geschäftsobjekt für Nutzdaten:

- „Geschäftsobjektstruktur“
- „Geschäftsobjekte konvertieren“ auf Seite 130
- „InterChange Server-Terminologie“ auf Seite 131

**Geschäftsobjektstruktur:** Das Geschäftsobjekt für Nutzdaten muss so entworfen werden, dass für jede Einzelinformation im Dokument, die Sie übertragen möchten, ein Attribut in der zugeordneten Definition für das Geschäftsobjekt für Nutzdaten vorhanden sein muss. Wie Tabelle 41 zeigt, ist der Inhalt des Geschäftsobjekts für Nutzdaten von der Struktur des Dokuments und dem vom Dokument verwendeten Pakettyp abhängig.

Tabelle 41. Beziehung des Pakettyps zur Struktur des Geschäftsobjekts für Nutzdaten

Dokumentstruktur	Pakettyp	Definition des Geschäftsobjekts für Nutzdaten
Nur Nutzdaten	Keine	Enthält die Nutzdaten des Dokuments.
Nur Nutzdaten	Back-End-Integration	Enthält: <ul style="list-style-type: none"> <li>• Die Nutzdaten des Dokuments</li> <li>• Header der Transportebene</li> </ul>
Nutzdaten und Anhänge	Keine	<i>Nicht möglich.</i> Sie müssen das Back-End-Integrationspaket verwenden, wenn Ihr Dokument Anhänge enthält.
Nutzdaten und Anhänge  Das Dokument enthält einen XML-Wrapper, der als Transportumschlag bezeichnet wird, und in den sowohl die Nutzdaten als auch Anhänge eingefügt sind.	Back-End-Integration	Enthält: <ul style="list-style-type: none"> <li>• Die Nutzdaten des Dokuments</li> <li>• Header der Transportebene</li> <li>• Container für Anhänge, der die Anhangsdaten und etwaige Anhangsgeschäftsobjekte enthält</li> </ul> <p>Ein von WebSphere Partner Gateway bereitgestellter Data-Handler, der als Attachment-Data-Handler bezeichnet wird, ist zur Verarbeitung des Transportumschlags erforderlich. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumente mit Anhängen verarbeiten“ auf Seite 133.</p>

Das Geschäftsobjekt für Nutzdaten muss außerdem gemäß den Anforderungen des entsprechenden Adapters erstellt werden, der zur Integration mit WebSphere Partner Gateway verwendet wird. In Tabelle 42 sind die Stellen aufgeführt, an denen Sie detaillierte Informationen zur Erstellung des Geschäftsobjekts für Nutzdaten für die Übertragung über ein bestimmtes Transportprotokoll finden.

Tabelle 42. Erstellen von Geschäftsobjekten für Nutzdaten für verschiedene Transportprotokolle

Transportprotokoll	Anmerkungen und Einschränkungen	Weitere Informationen
HTTP		„Geschäftsobjektdefinitionen für ICS über HTTP erstellen“ auf Seite 160

Tabelle 42. Erstellen von Geschäftsobjekten für Nutzdaten für verschiedene Transportprotokolle (Forts.)

Transportprotokoll	Anmerkungen und Einschränkungen	Weitere Informationen
JMS	Für Dokumente mit Back-End-Integrationspaket	„Geschäftsobjektdefinitionen für JMS erstellen“ auf Seite 182
Alle	Für Dokumente mit Anhängen	„Anhangsbezogene Geschäftsobjektdefinitionen erstellen“ auf Seite 145

**Geschäftsobjekte konvertieren:** In der Regel verwendet der Adapter einen Data-Handler zur Konvertierung zwischen dem Format des Dokuments und seiner Geschäftsobjektdarstellung. Dieser Data-Handler wird als Payload-Data-Handler bezeichnet. Der Adapter muss so konfiguriert werden, dass er den richtigen Data-Handler für den Inhaltstyp (Content-Type) der Nutzdaten aufruft. Gewöhnlich wird der WebSphere Business Integration-Data-Handler für XML als Payload-Data-Handler konfiguriert, weil er eine Konvertierung zwischen XML-Nachrichten und Geschäftsobjekten ausführt. Sie haben jedoch die Möglichkeit, angepasste Data-Handler für beliebige Nachrichtenformate zu erstellen, für die WebSphere Business Integration Server keinen entsprechenden Data-Handler bereitstellt.

**Anmerkung:** Zur Verarbeitung von XML-Nachrichten müssen Sie sicherstellen, dass Sie den WebSphere Business Integration-Data-Handler für XML der Version 2.3.1 oder höher verwenden. Für cXML-Nachrichten müssen Sie den Data-Handler für XML der Version 2.4.1 oder höher verwenden.

Sie müssen sicherstellen, dass die von Ihnen verwendeten Payload-Data-Handler die untergeordneten Metaobjekte ignorieren können, die für das von Ihnen verwendete Transportprotokoll erforderlich sind. Bevor Sie einen Data-Handler (sei es ein von WebSphere Business Integration bereitgestellter oder ein angepasster Data-Handler) verwenden, müssen Sie sicherstellen, dass er Unterstützung für untergeordnete Metaobjekte bietet. Lesen Sie den Abschnitt über den Tag *cw\_mo\_bezeichnung* in den anwendungsspezifischen Informationen des Geschäftsobjekts im entsprechenden Abschnitt für Ihr Transportprotokoll (siehe Tabelle 42 auf Seite 129).

Um festzulegen, welcher Data-Handler zur Konvertierung der Nutzdaten zu verwenden ist, müssen Sie die folgenden Schritte ausführen:

1. Stellen Sie den MIME-Typ fest, den der Data-Handler unterstützen muss, um die Nutzdaten zu konvertieren, und ermitteln Sie einen Data-Handler, der diesen MIME-Typ verarbeiten kann.

Im Handbuch *Data Handler Guide* der Dokumentation zu WebSphere Business Integration Adapter werden die Data-Handler beschrieben, die von IBM zur Verfügung gestellt werden. Wenn keiner dieser Data-Handler geeignet ist, können Sie einen angepassten Data-Handler erstellen.

2. Erstellen Sie in Business Object Designer ein untergeordnetes Metaobjekt (child-Metaobjekt) für den Data-Handler, den Sie verwenden müssen. Wenn Sie einen von IBM zur Verfügung gestellten Data-Handler verwenden, finden Sie Informationen zur Struktur des untergeordneten Metaobjekts im Handbuch *Data Handler Guide*.

3. Aktualisieren Sie in Business Object Designer das Data-Handler-Metaobjekt für Connector der höchsten Ebene, um ein Attribut für den unterstützten MIME-Typ einzufügen. Der Attributtyp für dieses Attribut ist das untergeordnete Metaobjekt (child-Metaobjekt) des Data-Handlers.
4. Definieren Sie in Connector Configurator die entsprechenden Connectorkonfigurationseigenschaften, um den zu verwendenden Data-Handler anzugeben:
  - Definieren Sie die Eigenschaften `DataHandlerConfigM0` und `DataHandlerMimeType` mit dem Namen des Data-Handler-Metaobjekts der höchsten Ebene bzw. dem unterstützten MIME-Typ.
  - Definieren Sie die Eigenschaft `DataHandlerClassName` mit dem Namen der zu instanziiierenden `DataHandler`-Klasse.

**Anmerkung:** Sie definieren *entweder* die Eigenschaften `DataHandlerConfigM0` und `DataHandlerMimeType` *oder* die Eigenschaft `DataHandlerClassName`.

5. Fügen Sie in Connector Configurator das Data-Handler-Metaobjekt der höchsten Ebene in die Liste der unterstützten Geschäftsobjekte ein.

**InterChange Server-Terminologie:** Für InterChange Server hängt der Name des Geschäftsobjekts für Nutzdaten wie folgt von der Kommunikationsrichtung ab:

- Wenn WebSphere Partner Gateway ein Dokument an InterChange Server *sendet*, nimmt es an der *Ereignisbenachrichtigung* von InterChange Server teil.  
In diesem Fall wird das Datengeschäftsobjekt als *Ereignisgeschäftsobjekt* (manchmal auch nur als *Ereignis*) bezeichnet, das InterChange Server über ein Ereignis benachrichtigt, das in einem externen Partner stattgefunden hat.
- Wenn WebSphere Partner Gateway ein Dokument von InterChange Server *empfängt*, nimmt es an der *Anforderungsverarbeitung* von InterChange Server teil.  
In diesem Fall ist das Datengeschäftsobjekt ein Anforderungsgeschäftsobjekt, das von InterChange Server gesendet wurde, um Informationen von einem externen Partner anzufordern. Als Reaktion darauf kann InterChange Server ein Antwortgeschäftsobjekt an die Hub-Community zurücksenden.

## Geschäftsobjekte für Konfigurationsdaten

Für viele der Adapter erstellen Sie Geschäftsobjektdefinitionen, um Konfigurationsdaten darin unterzubringen. Solche Geschäftsobjekte werden häufig als *Metaobjekte* bezeichnet.

In Tabelle 43 sind die Stellen aufgeführt, an denen Sie detaillierte Informationen zur Erstellung des Datengeschäftsobjekts für die Übertragung über ein bestimmtes Transportprotokoll finden.

Tabelle 43. Abschnitte, in denen die Erstellung von Geschäftsobjekten beschrieben werden:

Transportprotokoll	Zugehörige Komponente	Weitere Informationen
HTTP	Adapter für HTTP	„HTTP-Headerinformationen der Transportebene für ICS erstellen“ auf Seite 165
JMS	Adapter für JMS	„JMS-Headerinformationen erstellen“ auf Seite 183
Alle	Attachment-Data-Handler	„Untergeordnetes Metaobjekt für den Attachment-Data-Handler erstellen“ auf Seite 141

## Connector erstellen

Sie müssen ein Connectorobjekt für den Adapter erstellen, den Sie verwenden wollen. Dieses Connectorobjekt stellt eine Instanz des Adapters während der Ausführung dar. Sie können Connectorobjekte im Tool 'System Manager' von InterChange Server erstellen.

**Anmerkung:** Informationen zur Erstellung von Connectorobjekten finden Sie im Handbuch *System Implementation Guide* der Dokumentation zu WebSphere InterChange Server.

Tabelle 44 bietet eine Übersicht zu den Quellen, die Informationen zur Erstellung von Connectorobjekten für das von Ihnen verwendete Transportprotokoll enthalten.

Tabelle 44. Erstellen von Connectorobjekten für verschiedene Transportprotokolle

Transportprotokoll	Adapter	Weitere Informationen
HTTP	Adapter für HTTP	„HTTP-Connectorobjekt erstellen“ auf Seite 168
JMS	Adapter für JMS	„JMS-Connectorobjekt erstellen“ auf Seite 186

## Collaborations erstellen

Der eigentliche Geschäftsprozess, den Sie benötigen, wird innerhalb von InterChange Server durch die Collaboration ausgeführt. Daher muss die entsprechende Collaboration für InterChange Server vorhanden sein, um Ihre WebSphere Partner Gateway-Dokumente korrekt zu verarbeiten. Führen Sie die folgenden Arbeitsschritte aus, um die erforderliche Collaboration während der Ausführung bereitzustellen:

1. Vergewissern Sie sich, dass eine Collaboration-Schablone vorhanden ist, die den benötigten Geschäftsprozess zur Verfügung stellt:
  - Wenn eine solche Collaboration-Schablone momentan *nicht* vorhanden ist, müssen Sie diese erstellen und kompilieren.
  - Wenn eine Collaboration-Schablone vorhanden ist, müssen Sie mit ihrer Verwendungsweise genügend vertraut sein, um das zugehörige Collaboration-Objekt konfigurieren zu können.
2. Erstellen Sie ein Collaboration-Objekt, und binden Sie die zugehörigen Ports wie folgt:
  - Für die Anforderungsverarbeitung stellen Sie den Empfangsport ('To'-Port), der Anforderungen an WebSphere Partner Gateway sendet, auf den Adapter ein.
  - Für die Ereignisbenachrichtigung stellen Sie den Absenderport ('From'-Port), der Ereignisse von WebSphere Partner Gateway empfängt, auf den Adapter ein.

Tabelle 45 enthält eine Übersicht zu den Quellen, die Informationen zur Bindung von Collaboration-Objekten für das von Ihnen verwendete Transportprotokoll enthalten.

Tabelle 45. Collaboration-Bindung für verschiedene Transportprotokolle

Transportprotokoll	Adapter	Weitere Informationen
HTTP	Adapter für HTTP	„Collaborations zur Kommunikation mit dem Adapter für HTTP binden“ auf Seite 168
JMS	Adapter für JMS	„Collaborations zur Kommunikation mit dem Adapter für JMS binden“ auf Seite 187

## Projekt implementieren

Wenn Ihr Benutzerprojekt die Artefakte enthält, die die benötigten Laufzeitkomponenten definieren, müssen Sie es in InterChange Server implementieren. Die Implementierung eines Benutzerprojekts erfolgt über das Tool 'System Manager'.

---

## Dokumente mit Anhängen verarbeiten

WebSphere Partner stellt den Attachment-Data-Handler zur Verarbeitung von Dokumenten zur Verfügung, die zwischen WebSphere Partner Gateway und InterChange Server gesendet werden. Der Attachment-Data-Handler konvertiert ein Dokument im XML-Transportumschlag (mit oder ohne Anhänge) zwischen seinem serialisierten Format und seiner Geschäftsobjektdarstellung. In den beiden folgenden Fällen müssen Sie den Attachment-Data-Handler als Payload-Data-Handler konfigurieren:

- Die Umschlagsmarkierung für das Back-End-Integrationspaket wurde auf 'Ja' gesetzt.

Wenn diese Markierung auf 'Ja' gesetzt ist, fügt WebSphere Partner Gateway ein Dokument unabhängig davon, ob es Anhänge enthält, immer in einen XML-Transportumschlag ein. Diese Markierung für das Back-End-Integrationspaket wird über die Seite **B2B-Funktionalität** des Profils auf 'Ja' gesetzt. Weitere Informationen hierzu finden Sie im Abschnitt „Nutzdaten“ auf Seite 34.

- Das zu verarbeitende Dokument kann Anhänge enthalten.

Wenn ein Dokument Anhänge enthält, wird es von WebSphere Partner Gateway in einen XML-Transportumschlag eingefügt. In jedem Dokumentenfluss gibt es einen Teil mit Nutzdaten und optional mehrere Anhänge. Wenn Sie Dokumente mit Anhängen senden oder empfangen, muss Ihr Geschäftsobjekt für Nutzdaten Informationen über die Anhänge enthalten.

**Anmerkung:** Der Attachment-Data-Handler ist für SOAP-Dokumente, die Anhänge enthalten, *nicht* erforderlich. Informationen zur Behandlung von SOAP-Dokumenten finden Sie im Abschnitt „SOAP-Dokumente über HTTP/S senden“ auf Seite 169.

Der Attachment-Data-Handler wird von einem WebSphere Business Integration-Adapter aufgerufen.

- Wenn WebSphere Partner Gateway und InterChange Server das HTTP-Transportprotokoll verwenden, wird der Attachment-Data-Handler vom Adapter für HTTP aufgerufen.
- Wenn WebSphere Partner Gateway und InterChange Server das JMS-Transportprotokoll verwenden, wird der Attachment-Data-Handler vom Adapter für JMS aufgerufen.

Wenn der Adapter ein Dokument mit einem XML-Transportumschlag empfängt, ruft er den Attachment-Data-Handler auf, um dieses Dokument in die geeignete Geschäftsobjektdarstellung zu konvertieren. Als Beispiel zeigt Abb. 35 auf Seite 175, wie der Adapter für JMS den Attachment-Data-Handler aufruft, um das serialisierte Format des Dokuments in die entsprechende Geschäftsobjektdarstellung zu konvertieren. Umgekehrt ruft der Adapter bei Empfang einer Geschäftsobjektdarstellung für ein Dokument in einem XML-Transportumschlag den Attachment-Data-Handler auf, um diese Geschäftsobjektstruktur in das entsprechende Dokumentformat zu konvertieren. Als Beispiel zeigt Abb. 36 auf Seite 177, wie der Adapter für JMS den Attachment-Data-Handler aufruft, um die Geschäftsobjektdarstellung des Dokuments in das entsprechende serialisierte Format zu konvertieren.

Dieser Abschnitt enthält die folgenden Informationen zum Attachment-Data-Handler:

- „Konvertierung mit dem Attachment-Data-Handler“
- „Umgebung für den Attachment-Data-Handler einrichten“ auf Seite 139
- „Attachment-Data-Handler konfigurieren“ auf Seite 141
- „Anhangsbezogene Geschäftsobjektdefinitionen erstellen“ auf Seite 145

## Konvertierung mit dem Attachment-Data-Handler

Der Attachment-Data-Handler kann die Struktur des XML-Transportumschlags interpretieren und die Konvertierung zwischen den enthaltenen Daten und der entsprechenden Geschäftsobjektdarstellung wie in den folgenden Abschnitten beschrieben ausführen:

- „Dokumente in Geschäftsobjekte konvertieren“, um ein Dokument an InterChange Server zu senden
- „Geschäftsobjekte in Dokumente konvertieren“ auf Seite 137, um ein Dokument von InterChange Server zu empfangen

### Dokumente in Geschäftsobjekte konvertieren

Bevor WebSphere Partner Gateway ein Dokument an InterChange Server sendet, muss festgestellt werden, ob der Inhalt in einen XML-Transportumschlag eingefügt werden muss. Wenn WebSphere Partner Gateway den Transportumschlag erstellt, werden die Nutzdaten und alle Anhänge in Base64 codiert. Anschließend sendet WebSphere Partner Gateway den XML-Transportumschlag mit den entsprechenden Headern der Transportebene an den vorgesehenen Adapter. Dieser Adapter kann so konfiguriert werden, dass er den Attachment-Data-Handler zur Ausführung der Konvertierung der Nutzdaten und aller Anhänge in einem Dokument mit XML-Umschlag in die entsprechende Geschäftsobjektdarstellung aufruft.

Zur Konvertierung eines in einen XML-Transportumschlag eingefügten Dokuments in seine Geschäftsobjektdarstellung erstellt die aufrufende Entität eine Instanz des Attachment-Data-Handlers und leitet das Dokument (im Transportumschlag) an diese weiter. Der Attachment-Data-Handler führt daraufhin die folgenden Schritte aus:

1. Er lädt die Inhaltstypzuordnungen, die im untergeordneten Metaobjekt des Data-Handlers definiert sind.

Die Inhaltstypzuordnungen sind in den Konfigurationseigenschaften der Form `ContentTypeMap_x` des untergeordneten Metaobjekts (`child-Metaobjekts`) definiert. Das untergeordnete Metaobjekt ist ein Geschäftsobjekt, das die Konfigurationsdaten für den Attachment-Data-Handler enthält. Attribute in diesem Geschäftsobjekt ordnen Inhaltstypen Inhaltstypzuordnungen (`content-type`

- maps) zu. Weitere Informationen hierzu finden Sie im Abschnitt „Untergeordnetes Metaobjekt für den Attachment-Data-Handler erstellen“ auf Seite 141.
2. Er überprüft das Dokument, um festzustellen, ob es in einen XML-Transportumschlag eingefügt ist.
    - Wenn der Attachment-Data-Handler *keinen* Transportumschlag erkennt, müssen die Nutzdaten nicht aus der Umschlagsstruktur extrahiert werden. Das Dokument enthält nur Nutzdaten, die vom Attachment-Data-Handler in die entsprechende Geschäftsobjektdarstellung konvertiert werden müssen. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumente ohne Transportumschlag verarbeiten“.
    - Wenn der Attachment-Data-Handler den Transportumschlag *erkennt*, muss er die Nutzdaten und alle vorhandenen Anhänge aus dieser Umschlagsstruktur extrahieren. Das Dokument enthält Nutzdaten und möglicherweise mehrere Anhänge. Daher muss der Attachment-Data-Handler die Nutzdaten *und* alle vorhandenen Anhänge in ihre zugeordnete Geschäftsobjektdarstellung konvertieren. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumente in Transportumschlag verarbeiten“.
  3. Er definiert das resultierende Geschäftsobjekt für Nutzdaten und gibt dieses Geschäftsobjekt an die aufrufende Entität zurück.

**Dokumente ohne Transportumschlag verarbeiten:** Wenn der Attachment-Data-Handler feststellt, dass das Dokument *nicht* in einem XML-Transportumschlag enthalten ist, muss er die Nutzdaten nicht aus der Umschlagsstruktur extrahieren. Daher verwendet der Data-Handler die (im untergeordneten Metaobjekt definierte) Konfigurationseigenschaft `PayloadDataHandlerMimeType`, um den MIME-Typ zu ermitteln, der den standardmäßig verwendeten Payload-Data-Handler angibt, der für die Nutzdaten des Dokuments zu instanziiert ist. Dieser Data-Handler konvertiert die Nutzdaten in das entsprechende Geschäftsobjekt für Nutzdaten und gibt das resultierende Geschäftsobjekt an die aufrufende Entität zurück.

**Dokumente in Transportumschlag verarbeiten:** Wenn der Attachment-Data-Handler feststellt, dass das Dokument in einem XML-Transportumschlag enthalten ist, muss er die Nutzdaten und alle vorhandenen Anhänge aus dieser Umschlagsstruktur extrahieren, bevor er sie verarbeiten kann. Daher führt der Data-Handler die folgenden Schritte zur Verarbeitung und Konvertierung des Dokuments aus:

1. Er extrahiert die Nutzdaten und alle vorhandenen Anhänge aus dem Transportumschlag und decodiert die Nutzdaten. Die Nutzdaten sind im XML-Tag `<payload>` enthalten. Jeder Anhang ist in einem XML-Tag `<attachment>` enthalten.
2. Er durchsucht die Inhaltstypzuordnungen nach einem Inhaltstyp, der mit dem der Nutzdaten übereinstimmt. Der Attachment-Data-Handler verwendet den in der gefundenen Inhaltstypzuordnung angegebenen MIME-Typ, um eine Instanz eines Data-Handlers zu erstellen. Dieser Data-Handler konvertiert die Nutzdaten in das entsprechende Geschäftsobjekt für Nutzdaten und gibt das resultierende Geschäftsobjekt an den Attachment-Data-Handler zurück.
3. Er erstellt das Geschäftsobjekt für Inhaltsinformationen für die Nutzdaten. Der Attachment-Data-Handler untersucht die anwendungsspezifischen Informationen auf Geschäftsobjektebene der Nutzinformationsdefinition des Geschäftsobjekts und bestimmt den Namen des Geschäftsobjekts für Inhaltsinformationen, dessen Attributname durch den Tag `cw_mo_bcg_content_info` angegeben wird. Er erstellt dann eine Instanz dieses Geschäftsobjekts für Inhalts-

informationen und definiert die Werte für den Inhaltstyp (content-type) und die Codierung (encoding) für die Nutzdaten.

4. Er erstellt das Geschäftsobjekt für Anhangscontainer für die Nutzdaten.

Der Attachment-Data-Handler untersucht die anwendungsspezifischen Informationen auf Geschäftsobjektebene des Geschäftsobjekts für Nutzdaten und bestimmt den Namen des Geschäftsobjekts für Anhangscontainer, dessen Attributname durch den Tag `cw_mo_bcg_attachment` angegeben wird. Er erstellt dann eine Instanz des Geschäftsobjekts für Anhangscontainer und speichert es im entsprechenden Attribut des Geschäftsobjekts für Nutzdaten.

Wenn der Tag `cw_mo_bcg_attachment` nicht vorhanden (oder leer) ist, wird angenommen, dass im Dokument keine Anhänge enthalten sind. In diesem Fall sind keine weiteren Verarbeitungsschritte erforderlich. Der Attachment-Data-Handler gibt das konvertierte Geschäftsobjekt für Nutzdaten zurück.

5. Er erstellt das Standardanhangsgeschäftsobjekt für den Anhangscontainer.

Der Attachment-Data-Handler untersucht die anwendungsspezifischen Informationen auf Geschäftsobjektebene des Geschäftsobjekts für Anhangscontainer und bestimmt den Namen des Standardanhangsgeschäftsobjekts, dessen Attributname durch den Tag `cw_mo_bcg_default_attribute` angegeben wird. Er erstellt dann eine Instanz des Standardanhangsgeschäftsobjekts und speichert es im entsprechenden Attribut des Geschäftsobjekts für Anhangscontainer.

6. Er stellt fest, ob der Anhang in ein Geschäftsobjekt zu konvertieren ist, indem er die Inhaltstypzuordnungen nach einem Inhaltstyp durchsucht, der mit dem des Anhangs übereinstimmt.

Der Attachment-Data-Handler überprüft den Inhaltstyp (content-type) und die Zeichensatzcodierung (character-set encoding) des Anhangs und überprüft, ob es einen entsprechenden Eintrag in einer Inhaltstypzuordnung gibt.

- Wenn *keine* entsprechende Inhaltstypzuordnung gefunden wird, erstellt der Attachment-Data-Handler kein Geschäftsobjekt für die Anhangsdaten.

In diesem Fall erstellt der Data-Handler eine Instanz des standardmäßig verwendeten Anhangsgeschäftsobjekts, definiert die Werte für den Inhaltstyp und die Codierung innerhalb des Geschäftsobjekts für Inhaltsinformationen und definiert die im Base64-Format codierten Anhangsdaten (als Zeichenfolge) im Attribut `attachment`.

Anschließend füllt der Attachment-Data-Handler das Geschäftsobjekt für Anhangscontainer mit dem Standardanhangsgeschäftsobjekt.

- Wenn eine Inhaltstypzuordnung *gefunden* wird, prüft der Attachment-Data-Handler, ob der Anhang in ein Geschäftsobjekt zu konvertieren ist:
  - Wenn die Konfigurationseigenschaft `ConvertAttachment` in der übereinstimmenden Inhaltstypzuordnung den Wert `'false'` hat, erstellt der Attachment-Data-Handler eine Instanz des standardmäßig verwendeten Anhangsgeschäftsobjekts, definiert die Werte für den Inhaltstyp und die Codierung innerhalb des Geschäftsobjekts für Inhaltsinformationen und definiert die im Base64-Format codierten Anhangsdaten (als Zeichenfolge) im Attribut `attachment`.

Anschließend füllt der Attachment-Data-Handler das Geschäftsobjekt für Anhangscontainer mit dem Standardanhangsgeschäftsobjekt.

- Wenn die Konfigurationseigenschaft `ConvertAttachment` in der übereinstimmenden Inhaltstypzuordnung den Wert `'true'` hat, decodiert der Attachment-Data-Handler die Anhangsdaten und erstellt eine Instanz eines Data-Handlers zur Verarbeitung der Anhangsdaten. Dieser Data-Handler verarbeitet die decodierten Byte und gibt das entsprechende Anhangsgeschäftsobjekt zurück.

Der Attachment-Data-Handler untersucht anschließend die anwendungsspezifischen Informationen auf Geschäftsobjektebene der Definition des Anhangsgeschäftsobjekts und bestimmt den Namen des Geschäftsobjekts für Inhaltsinformationen, dessen Attributname durch den Tag `cw_mo_bcg_content_info` angegeben wird. Wenn dieser Tag vorhanden ist, erstellt der Data-Handler das Geschäftsobjekt für Inhaltsinformationen für den Anhang und legt die Werte für den Inhaltstyp und die Codierung des Anhangs fest.

Schließlich füllt der Attachment-Data-Handler das Geschäftsobjekt für Anhangscontainer mit dem Anhangsgeschäftsobjekt.

## Geschäftsobjekte in Dokumente konvertieren

Bevor WebSphere Partner Gateway ein Dokument von InterChange Server empfängt, muss ein Adapter feststellen, ob die Geschäftsobjektdarstellung der Nutzdaten und aller vorhandenen Anhänge in einen XML-Transportumschlag eingefügt werden muss. InterChange Server sendet das Geschäftsobjekt an den vorgesehenen Data-Handler, der die eigentliche Konvertierung ausführt. Dieser Data-Handler kann so konfiguriert werden, dass er den Attachment-Data-Handler zur Ausführung der Konvertierung des Geschäftsobjekts für Nutzdaten und aller Geschäftsobjekte für Anhänge in die entsprechenden Nutzdaten und Anhänge sowie zur Erstellung eines XML-Transportumschlags aufruft.

Zur Konvertierung eines Geschäftsobjekts für Nutzdaten mit Anhängen in die entsprechende Darstellung mit Transportumschlag erstellt die aufrufende Entität eine Instanz des Attachment-Data-Handlers und übergibt dieser das Geschäftsobjekt für Nutzdaten. Der Attachment-Data-Handler führt die folgenden Schritte aus:

1. Er lädt die Inhaltstypzuordnungen, die im zugehörigen Konfigurationsmetaobjekt definiert sind.

Die Inhaltstypzuordnungen sind in den Konfigurationseigenschaften der Form `ContentTypeMap_x` des untergeordneten Metaobjekts (`child-Metaobjekts`) definiert. Das untergeordnete Metaobjekt ist ein Geschäftsobjekt, das die Konfigurationsdaten für den Attachment-Data-Handler enthält. Attribute in diesem Geschäftsobjekt ordnen Inhaltstypen Inhaltstypzuordnungen (`content-type maps`) zu. Weitere Informationen hierzu finden Sie im Abschnitt „Untergeordnetes Metaobjekt für den Attachment-Data-Handler erstellen“ auf Seite 141.

2. Er prüft das Geschäftsobjekt, um festzustellen, ob ein XML-Transportumschlag zu erstellen ist.

- Wenn der Attachment-Data-Handler *nicht* feststellt, dass für das Dokument ein Transportumschlag erforderlich ist, muss er die Nutzdaten nicht in diese Umschlagsstruktur einfügen.

Das Dokument enthält nur Nutzdaten, die vom Attachment-Data-Handler aus der zugeordneten Geschäftsobjektdarstellung erstellt werden müssen. Der Data-Handler muss keinen Transportumschlag für das Dokument erstellen. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumente ohne Transportumschlag erstellen“ auf Seite 138.

- Wenn der Attachment-Data-Handler feststellt, dass für das Dokument ein Transportumschlag erforderlich ist, muss er die Nutzdaten und alle vorhandenen Anhänge in diese Umschlagsstruktur einfügen.

Das Dokument enthält Nutzdaten und möglicherweise mehrere Anhänge. Daher muss der Attachment-Data-Handler die Geschäftsobjektdarstellung der Nutzdaten in Nutzdaten *und* alle vorhandenen Anhänge konvertieren und diese Komponenten in einen Transportumschlag einfügen. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumente mit Transportumschlag erstellen“ auf Seite 138.

3. Er definiert die resultierenden Tags für Nutzdaten und Anhänge (attachment) im WebSphere Partner Gateway-Dokument und gibt dieses Dokument an die aufrufende Entität zurück.

**Dokumente ohne Transportumschlag erstellen:** Wenn der Attachment-Data-Handler feststellt, dass das Geschäftsobjekt für Nutzdaten *keinen* XML-Transportumschlag erfordert, muss er die Nutzdaten nicht in die Umschlagsstruktur einfügen. In diesem Fall verwendet der Data-Handler den Standard-Payload-Data-Handler, um das Geschäftsobjekt für Nutzdaten in das entsprechende Dokument mit Nutzdaten zu konvertieren. Die Konfigurationseigenschaft `PayloadDataHandlerMimeType` (die im untergeordneten Metaobjekt des Attachment-Data-Handlers definiert ist) enthält den MIME-Typ, der den Standard-Payload-Data-Handler angibt, der für das Geschäftsobjekt für Nutzdaten zu instanziiert ist. Dieser Data-Handler empfängt das Geschäftsobjekt für Nutzdaten als Argument und gibt das resultierende Dokument mit Nutzdaten an die aufrufende Entität zurück.

**Dokumente mit Transportumschlag erstellen:** Wenn der Attachment-Data-Handler feststellt, dass das Geschäftsobjekt für Nutzdaten einen XML-Transportumschlag *erfordert*, muss er die Dokumente mit Nutzdaten und die Anhangsdokumente in diese Umschlagsstruktur einfügen. Daher führt der Data-Handler die folgenden Schritte zur Verarbeitung und Konvertierung des Geschäftsobjekts aus:

1. Er ruft den Inhaltstyp (content-type) und die Zeichensatzcodierung (character set encoding) für die Nutzdaten ab.

Der Tag `cw_mo_bcg_content_info` in den anwendungsspezifischen Informationen auf Geschäftsobjektebene des Geschäftsobjekts für Nutzdaten gibt den Namen des Attributs für Inhaltsinformationen an. Dieses Attribut enthält den Inhaltstyp und die Codierung für die Nutzdaten.

**Anmerkung:** Wenn das Attribut für Inhaltsinformationen nicht vorhanden ist, verwendet er den Standard-Data-Handler (der durch den MIME-Typ in der Konfigurationseigenschaft `PayloadDataHandlerMimeType` im untergeordneten Metaobjekt des Attachment-Data-Handlers angegeben ist), um das Geschäftsobjekt für Nutzdaten zu konvertieren.

2. Er durchsucht die Inhaltstypzuordnungen nach einem Inhaltstyp, der mit dem der Nutzdaten übereinstimmt.

Der Attachment-Data-Handler verwendet den in der gefundenen Inhaltstypzuordnung angegebenen MIME-Typ, um eine Instanz eines Payload-Data-Handlers zu erstellen. Dieser Data-Handler konvertiert das Geschäftsobjekt für Nutzdaten in das entsprechende Dokument für Nutzdaten und gibt das resultierende Dokument an den Attachment-Data-Handler zurück. Die Byte der Zeichenfolge, die vom Payload-Data-Handler zurückgegeben wird, codiert der Attachment-Data-Handler mit Base64 und speichert das Ergebnis im Tag `'payload'` des XML-Transportumschlags.

3. Er ruft den Anhangscontainer aus dem Geschäftsobjekt für Nutzdaten ab.

Der Anhangscontainer befindet sich im Attribut für Anhangscontainer (`attachment-container`) des Geschäftsobjekts für Nutzdaten. Die anwendungsspezifischen Informationen auf Geschäftsobjektebene des Geschäftsobjekts für Nutzdaten enthalten den Tag `cw_mo_bcg_attachment`, der das Attribut für Anhangscontainer angibt. Dieses Attribut enthält die Anhänge.

Wenn der Tag `cw_mo_bcg_attachment` nicht vorhanden (oder leer) ist, wird angenommen, dass im Dokument keine Anhänge enthalten sind. In diesem Fall sind

keine weiteren Verarbeitungsschritte erforderlich. Der Attachment-Data-Handler gibt die konvertierten Nutzdaten im entsprechenden Transportumschlag zurück.

4. Er bestimmt für jeden Anhang, ob der Anhang als Geschäftsobjekt oder als reine Daten dargestellt wird.
  - Wenn der Anhang aus reinen Anhangsdaten besteht, enthalten die anwendungsspezifischen Informationen auf Geschäftsobjektebene des Geschäftsobjekts für Anhangscontainer den Tag `cw_mo_bcg_default_attribute`, der das Attribut für Standardanhänge angibt. Dieses Attribut enthält die Anhangsdaten. Der Attachment-Data-Handler ruft diese Daten ab, extrahiert die im Base64-Format codierten Daten und speichert das Ergebnis im Dokument.
  - Wenn der Anhang durch ein Geschäftsobjekt dargestellt wird, enthalten seine anwendungsspezifischen Informationen auf Geschäftsobjektebene den Tag `wbic_type`, der angibt, dass ein Anhangsgeschäftsobjekt enthalten ist. Der Attachment-Data-Handler führt die folgenden Schritte aus, um das Anhangsgeschäftsobjekt zu verarbeiten:
    - a. Er ruft den Inhalt des Anhangsattributs sowie den Inhaltstyp und die Codierung für den Anhang ab.

Die anwendungsspezifischen Informationen auf Geschäftsobjektebene des Anhangsgeschäftsobjekts enthalten den Tag `cw_mo_bcg_content_info`, der das Attribut für Inhaltsinformationen angibt. Dieses Attribut enthält den Inhaltstyp und die Codierung für diesen Anhang. Der Attachment-Data-Handler speichert diese Inhaltsinformationen im Tag `'attachment'` des Dokuments.
    - b. Er durchsucht die Inhaltstypzuordnungen nach einem Inhaltstyp, der mit dem des Anhangs übereinstimmt.

Der Attachment-Data-Handler verwendet den in der passenden Inhaltstypzuordnung angegebenen MIME-Typ, um eine Instanz eines Data-Handlers zu erstellen. Dieser Data-Handler konvertiert das Anhangsgeschäftsobjekt in das entsprechende Anhangsdokument und gibt das resultierende Dokument (als Zeichenfolge) an den Attachment-Data-Handler zurück.
    - c. Er speichert das codierte Ergebnis im Tag `'attachment'` des XML-Wrappers für das Dokument.

Der Attachment-Data-Handler ruft die Byte aus der zurückgegebenen Zeichenfolge (unter Verwendung des Zeichensatzes, falls einer vorhanden war) ab und codiert die Byte mit Base64. Anschließend speichert er das Ergebnis im Tag `'attachment'`.

## Umgebung für den Attachment-Data-Handler einrichten

Für die Verwendung des von WebSphere Partner Gateway zur Verfügung gestellten Attachment-Data-Handlers sind folgende Schritte erforderlich:

- „Zu verwendendes Schema angeben“
- „Attachment-Data-Handler einrichten“ auf Seite 140
- „Attachment-Data-Handler konfigurieren“ auf Seite 141

### Zu verwendendes Schema angeben

Sie können entweder das Standardschema für den Attachment-Data-Handler verwenden oder ein Schema (`wbipackaging_v1.1_ns.xsd`) einsetzen, das es Ihnen erlaubt, die Inhalts-ID (`contentId`) im Back-End-Integrationspaket weiterzugeben.

Wenn Sie das Schema 'wbipackaging\_v1.1\_ns.xsd' verwenden wollen, konfigurieren Sie die Eigenschaft 'wbipackaging\_version' in der Datei 'bcg.properties'. (Die Datei 'bcg.properties' wird im Handbuch *Verwaltung* beschrieben.) Diese Eigenschaft ist folgendermaßen definiert:

```
wbipackaging_version=1.n
```

Dabei hat *n* den Wert 0 oder 1. Der Standardwert für diese Eigenschaft ist 1.0.

Das Metaobjekt des Attachment-Data-Handlers hat das Attribut 'wbipackaging\_version', das den Wert 1.0 oder 1.1 annehmen kann. Wenn Sie den Wert 1.1 angeben, führt der Attachment-Data-Handler eine syntaktische Analyse derXML-Anhänge mit der Inhalts-ID (contentId) des Anhangs durch und generiert sie.

Das codierende Geschäftsobjekt gibt die Inhalts-ID des Anhangs durch das Attribut 'contentId' an. Bei der Erstellung von XML-Code aus dem Geschäftsobjekt verwendet der Attachment-Data-Handler dieses Attribut, um einen contentId-Tag für den Anhang zu erstellen. Bei der Erstellung eines Geschäftsobjekts aus dem XML-Code gibt der Attachment-Data-Handler einen Wert für dieses Attribut an. Dazu verwendet er den Wert, der im Tag 'contentId' der XML-Nachricht angegeben ist.

### Attachment-Data-Handler einrichten

Der Attachment-Data-Handler und die zugeordnete Repository-Datei stehen auf dem Installationsdatenträger von WebSphere Partner Gateway an den in Tabelle 46 aufgelisteten Positionen zur Verfügung.

Tabelle 46. Positionen der Komponenten für den Attachment-Data-Handler

Komponente	Position
Attachment-Data-Handler	Integration/WBI/WICS/Attachment/ bcgwbattachmentdh.jar
Repository-Datei	Integration/WBI/WICS/Attachment/ MO_DataHandler_DefaultAttachmentConfigV1.0.in  oder  Integration/WBI/WICS/Attachment/ MO_DataHandler_DefaultAttachmentConfigV1.1.in

Richten Sie die Dateien im Webserver entsprechend der Dokumentation zu diesem Webserver ein.

### Position des Attachment-Data-Handlers angeben

WebSphere InterChange Server muss die Speicherposition des Attachment-Data-Handlers kennen, um ihn während der Ausführung laden zu können. Zur Angabe der Speicherposition des Attachment-Data-Handlers führen Sie folgende Schritte aus:

1. Bearbeiten Sie das ICS-Startscript `start_server.bat`, das sich im Unterverzeichnis `bin` des Produktverzeichnisses von InterChange Server befindet (auf dem Computer, auf dem sich InterChange Server befindet).
2. Fügen Sie der Variablen `CLASSPATH` in dieser Datei die JAR-Datei für den Attachment-Data-Handler hinzu: `bcgwbattachmentdh.jar`

## Attachment-Data-Handler konfigurieren

Die Konfiguration des Attachment-Data-Handlers besteht aus den folgenden Schritten zur Erstellung der Konfigurationsgeschäftsobjekte:

- „Untergeordnetes Metaobjekt für den Attachment-Data-Handler erstellen“
- „Data-Handler-Metaobjekt der höchsten Ebene aktualisieren“ auf Seite 144

**Anmerkung:** Sie müssen außerdem die anhangsbezogenen Geschäftsobjektdefinitionen für den Attachment-Data-Handler erstellen. Weitere Informationen hierzu finden Sie im Abschnitt „Anhangsbezogene Geschäftsobjektdefinitionen erstellen“ auf Seite 145.

### Untergeordnetes Metaobjekt für den Attachment-Data-Handler erstellen

Zur Konfiguration des Attachment-Data-Handlers müssen Sie ein untergeordnetes Metaobjekt (child-Metaobjekt) erstellen, um den Klassennamen und die Konfigurationseigenschaften, die vom Attachment-Data-Handler benötigt werden, zur Verfügung zu stellen. Zur Erstellung dieses Metaobjekts erstellen Sie eine Geschäftsobjektdefinition, die die in Tabelle 47 aufgeführten Attribute enthält. Verwenden Sie das zum WebSphere Business Integration Toolset gehörende Tool 'Business Object Designer' zur Erstellung dieser Geschäftsobjektdefinition.

Das untergeordnete Metaobjekt stellt den Klassennamen und die Konfigurationseigenschaften zur Verfügung, die vom Attachment-Data-Handler benötigt werden. Erstellen Sie im Business Object Designer ein untergeordnetes Metaobjekt, das die MIME-Typen für die Nutzdaten und für die Typen von Anhängen enthält, deren Empfang Sie erwarten.

Die Attribute des untergeordneten Metaobjekts sind in Tabelle 47 aufgeführt. Ein Beispiel für ein untergeordnetes Metaobjekt für den Attachment-Data-Handler sehen Sie in Abb. 28 auf Seite 144.

**Anmerkung:** Die in diesem Kapitel gezeigten Beispielgeschäftsobjekte enthalten die Standardattribute (z. B. 'ObjectEventId') *nicht*, die für WebSphere InterChange Server erforderlich sind, jedoch vom Attachment-Data-Handler nicht verwendet werden.

Tabelle 47. Konfigurationseigenschaften im untergeordneten Metaobjekt für den Attachment-Data-Handler

Attributname	Beschreibung
ClassName	Der Klassename (erforderlich), der auf die folgende DataHandler-Klasse verweist: <code>com.ibm.bcg.DataHandlers.AttachmentDataHandler</code>
ContentTypeMap_x	Die Inhaltstypzuordnung für die Nutzdaten und für jeden einzelnen Typ von Anhang, dessen Empfang im XML-Wrapper zu erwarten ist.  Weitere Informationen hierzu finden Sie im Abschnitt „Inhaltstypzuordnungen“ auf Seite 142.
PayloadDataHandlerMimeType	Der MIME-Typ, der zur Angabe des Standard-Data-Handlers dient, der Nutzdaten verarbeitet, die <i>keine</i> zugeordneten Anhänge haben.

Tabelle 47. Konfigurationseigenschaften im untergeordneten Metaobjekt für den Attachment-Data-Handler (Forts.)

Attributname	Beschreibung
wbpackaging_version	Dieses Attribut kann den Wert 1.0 oder 1.1 haben. Wenn Sie den Wert 1.1 angeben, führt der Attachment-Data-Handler eine syntaktische Analyse derXML-Anhänge mit der Inhalts-ID (contentId) des Anhangs durch und generiert sie.

**Wichtig:** Um den in Tabelle 47 aufgeführten Attributen einen Wert zuzuordnen, geben Sie den Standardwert für das jeweilige Attribut an. Wenn der Attachment-Data-Handler zum Beispiel den XML-Data-Handler als Standard-Data-Handler verwenden soll, setzen Sie den Standardwert des Attributs `PayloadDataHandlerMimeType` auf `text/xml`.

**Inhaltstypzuordnungen:** Die Inhaltstypzuordnung bestimmt den Data-Handler, der vom Attachment-Data-Handler zum Konvertieren von Informationen aufgerufen wird, die im zugeordneten Inhaltstyp formatiert sind. Wenn als Inhaltstyp (Content-Type) der Nutzdaten zum Beispiel `application/xml` angegeben ist, sucht der Attachment-Data-Handler nach einer Inhaltstypzuordnung, deren Attribut `ContentType` den Wert `application/xml` enthält. Wenn kein übereinstimmender Inhaltstyp gefunden wird, nimmt der Data-Handler an, dass der zugeordnete Anhang *nicht* in ein Geschäftsobjekt konvertiert werden soll.

Sie erstellen eine Inhaltstypzuordnung für jeden dieser Inhaltstypen mit den in Tabelle 49 gezeigten anwendungsspezifischen Informationen auf Attributebene.

Wenn Sie ein Attribut im untergeordneten Metaobjekt erstellen, das eine Inhaltstypzuordnung darstellt, sind folgende Punkte zu beachten:

- Der Name des Attributs für die Inhaltstypzuordnung hat das folgende Format:  
`ContentTypeMap_x`  
Dabei ist *x* eine ganze Zahl, welche die Inhaltstypzuordnung innerhalb der Geschäftsobjektdefinition eindeutig identifiziert.

**Anmerkung:** Sie müssen die Attribute `ContentTypeMap_x` in der richtigen Reihenfolge anordnen. Wenn Sie zum Beispiel drei Inhaltstypzuordnungen haben, müssen die zugehörigen Attribute die Namen `ContentType_1`, `ContentType_2` und `ContentType_3` erhalten.

- Der Standardwert des Attributs für eine Inhaltstypzuordnung muss eine Kombination aus gültigen Tags enthalten.

In Tabelle 48 sind die Tags aufgeführt, die der Standardwert für dieses Attribut enthalten kann.

Tabelle 48. Gültige Tags für den Standardwert des Attributs für Inhaltstypzuordnung

Tagname	Beschreibung	Erforderlich
<code>ContentType</code>	Der tatsächliche Inhaltstyp, der im Transportumschlag ankommt (z. B. <code>text/xml</code> ).	Ja
<code>MimeType</code>	Der MIME-Typ, der zur Angabe des Data-Handlers zur Konvertierung des zugeordneten Inhaltstyps in ein Geschäftsobjekt dient. Wenn Sie <code>MimeType</code> nicht angeben, verwendet der Data-Handler den Wert <code>ContentType</code> , um eine Instanz des Data-Handlers zu erstellen.	Nein

Tabelle 48. Gültige Tags für den Standardwert des Attributs für Inhaltstypzuordnung (Forts.)

Tagname	Beschreibung	Erforderlich
CharSet	<p>Der Name eines Zeichensatzes (z. B. UTF-8), der vom Attachment-Data-Handler zum Konvertieren von Byte in eine Zeichenfolge bzw. einer Zeichenfolge in Byte verwendet wird.</p> <p>Wenn Sie CharSet nicht angeben, führt der Attachment-Data-Handler folgende Aktionen aus:</p> <ul style="list-style-type: none"> <li>• Für eingehende Daten werden die Datenbyte, die sich aus der Decodierung der Nachricht aus der Base64-Codierung ergeben, für die Konvertierung in das Geschäftsobjekt verwendet.</li> <li>• Für ausgehende Daten erfolgen Aufrufe an die Methode des untergeordneten Data-Handlers, der Byte (und keine Zeichenfolge) zurückgibt.</li> </ul>	Nein
ConvertAttachment	Ein Boolescher Wert, der angibt, ob der Anhang in ein Geschäftsobjekt zu konvertieren ist. Der Standardwert ist false.	Nein

Die Inhaltstypzuordnung kann darüber hinaus den Zeichensatz zur Codierung sowie eine Angabe enthalten, ob ein Anhang in ein Geschäftsobjekt konvertiert werden soll. Eine Beschreibung der Attribute des untergeordneten Metaobjekts und ein Beispiel finden Sie im Abschnitt „Untergeordnetes Metaobjekt für den Attachment-Data-Handler erstellen“ auf Seite 141.

Nehmen Sie zum Beispiel an, Ihr Dokument enthält die folgenden Inhaltstypen:

- application/xml
- text/xml
- application/octet-stream

Tabelle 49. Beispiele für Inhaltstypzuordnungen

Inhaltstyp	Attributname	Standardwert
text/xml	ContentType_1	ContentType=text/xml;MimeType=myxml; CharSet=UTF-8;ConvertAttachment=false;
application/xml	ContentType_2	ContentType=application/xml; MimeType=mynewxml;CharSet=UTF-16; ConvertAttachment=true;
application/octet-stream	ContentType_3	ContentType=application/octet-stream; MimeType=myoctet

**Muster für ein untergeordnetes Metaobjekt:** WebSphere Partner Gateway stellt die folgenden Repository-Eingabedateien für InterChange Server zur Verfügung, die ein Muster für ein untergeordnetes Metaobjekt für den Attachment-Data-Handler enthalten:

*Produktverzeichnis*/Integration/WBI/WICS/Attachment/  
MO\_DataHandler\_DefaultAttachmentConfigV1.0.in

*Produktverzeichnis*/Integration/WBI/WICS/Attachment/  
MO\_DataHandler\_DefaultAttachmentConfigV1.1.in

Dabei steht *Produktverzeichnis* für das Verzeichnis, in dem Ihr Produkt WebSphere Partner Gateway installiert ist. Die Repository-Dateien definieren einen einzigen

Attachment-Data-Handler, dessen zugehöriges untergeordnetes Metaobjekt MO\_DataHandler\_DefaultAttachmentConfig ist. Abb. 28 zeigt ein Beispiel eines untergeordneten Metaobjekts für den Attachment-Data-Handler. Dieses Metaobjekt definiert zwei Inhaltstypzuordnungen: ContentTypeMap\_1 und ContentTypeMap\_2.

MO_DataHandler_DefaultAttachmentConfig
Name = ClassName Default Value = com.ibm.bcg.DataHandlers. AttachmentDataHandler
Name = ContentTypeMap_1 Default Value = ContentType=application/xml; MimeType=text/xml;CharSet=UTF-8; ConvertAttachment=true;
Name = ContentTypeMap_2 Default Value = ContentType=text/xml; MimeType=text/xml;CharSet=UTF-8;
Name = PayloadDataHandlerMimeType Default Value = text/xml

Abbildung 28. Beispiel eines untergeordneten Metaobjekts für den Attachment-Data-Handler

MO\_DataHandler\_DefaultAttachmentConfigV1.1.in enthält das zusätzliche Attribut:

```
[Attribute]
  Name = WBIPackaging_Version
  Type = String
  MaxLength = 255
  IsKey = false
  IsForeignKey = false
  IsRequired = true
  DefaultValue = 1.0
  IsRequiredServerBound = false
[End]
```

### Data-Handler-Metaobjekt der höchsten Ebene aktualisieren

Ein WebSphere Business Integration Adapter (z. B. der Adapter für JMS) verwendet das Metaobjekt MO\_DataHandler\_Default, um die Data-Handler anzugeben, die er verwenden kann. Fügen Sie einen Verweis auf den Attachment-Data-Handler zum Metaobjekt hinzu.

Am Metaobjekt MO\_DataHandler\_Default nehmen Sie die folgenden Änderungen vor:

1. Fügen Sie ein Attribut hinzu, dessen Name den MIME-Typ angibt, der der Instanz des Attachment-Data-Handlers zugeordnet ist, so dass ein Dokument, das diesen MIME-Typ enthält, von diesem zugeordneten Data-Handler in ein Geschäftsobjekt konvertiert werden kann.  
Der Attributtyp dieses Attributs ist die Geschäftsobjektdefinition für das untergeordnete Metaobjekt des Attachment-Data-Handlers (siehe „Untergeordnetes Metaobjekt für den Attachment-Data-Handler erstellen“ auf Seite 141).
2. Fügen Sie für jeden der unterstützten MIME-Typen für Anhänge ein Attribut hinzu, falls diese nicht bereits im Data-Handler-Metaobjekt der höchsten Ebene vorhanden sind.

Der Attributtyp dieser Attribute ist jeweils das untergeordnete Metaobjekt des zugeordneten Data-Handlers.

Nehmen Sie zum Beispiel an, Sie haben den Attachment-Data-Handler wie in Abb. 28 auf Seite 144 konfiguriert. Abb. 29 zeigt das Metaobjekt `MO_DataHandler_Default` mit einem Attribut, das den MIME-Typ `wbic_attachment` der Instanz des Attachment-Data-Handlers zuordnet, die durch das untergeordnete Metaobjekt `MO_DataHandler_DefaultAttachmentConfig` konfiguriert wird. Durch dieses Data-Handler-Metaobjekt der höchsten Ebene wird außerdem der MIME-Typ für Dokumente (`text/xml`) dem untergeordneten Metaobjekt des XML-Data-Handlers zugeordnet.

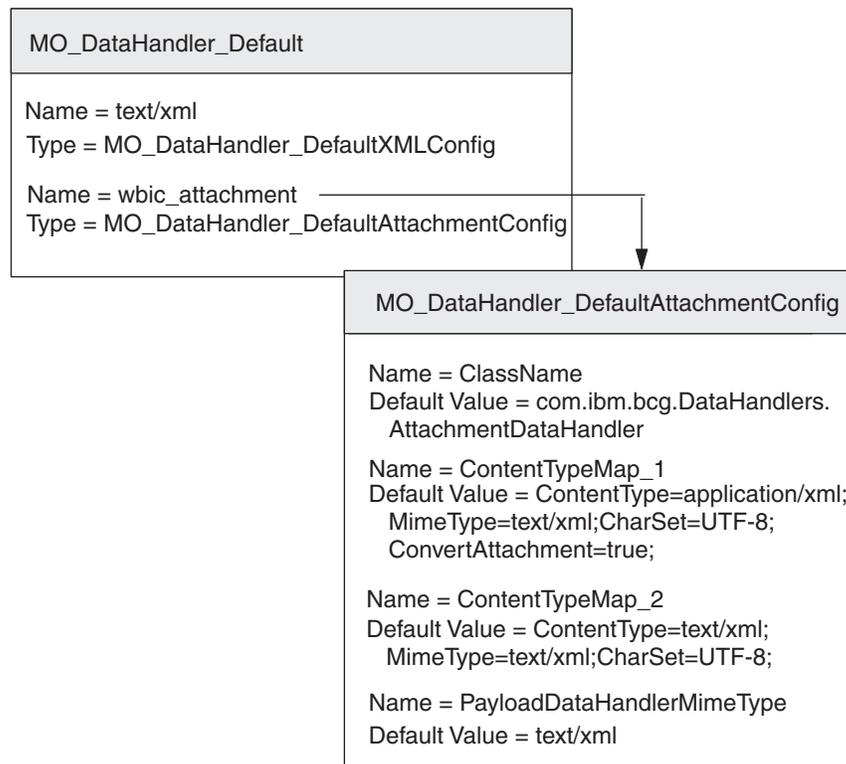


Abbildung 29. Zuordnen des MIME-Typs 'wbic\_attachment' zum Attachment-Data-Handler

Wiederholen Sie den Prozess für jede eindeutige Kombination aus unterstützten Inhaltstypen, die Sie unterstützen müssen, indem Sie ein Attribut im entsprechenden Data-Handler-Metaobjekt der höchsten Ebene hinzufügen, dessen Attributname der der Instanz des Attachment-Data-Handlers zugeordnete MIME-Typ ist und dessen Typ der Name des zugeordneten untergeordneten Metaobjekts ist. Stellen Sie darüber hinaus sicher, dass die konfigurierten MIME-Typen (und ihre untergeordneten Metaobjekte) im Metaobjekt der höchsten Ebene vorhanden sind.

## Anhangsbezogene Geschäftsobjektdefinitionen erstellen

Wenn Sie Dokumente, die in einen XML-Transportumschlag eingefügt sind, senden oder empfangen, muss Ihr Geschäftsobjekt für Nutzdaten Informationen über die Anhänge enthalten. In jedem Dokumentenfluss gibt es einen Teil mit Nutzdaten und optional mehrere Anhänge. Der Attachment-Data-Handler erwartet, dass sich diese Anhangsinformationen in *anhangsbezogenen Geschäftsobjekten* befinden. Daher müssen Sie Geschäftsobjektdefinitionen erstellen, die diese Informationen darstel-

len. Eine Geschäftsobjektdefinition ist die Form von Informationen, die von Inter-Change Server verwendet wird. Geschäftsobjektdefinitionen werden mit Hilfe des Tools 'Business Object Designer' erstellt.

Abb. 30 zeigt die Geschäftsobjektstruktur für einen Teil mit Nutzdaten, der in einen XML-Transportumschlag eingefügt ist.

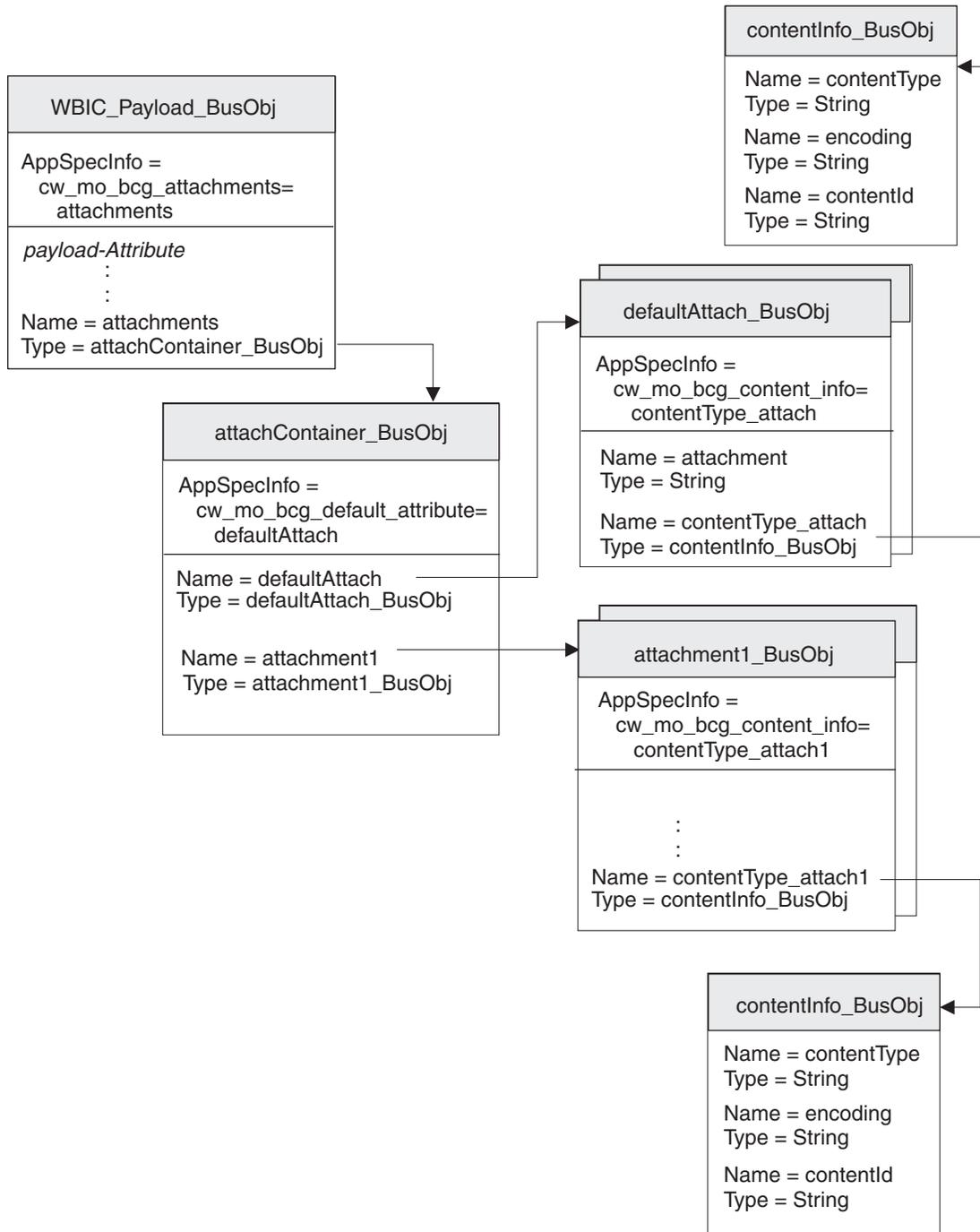


Abbildung 30. Beziehung zwischen dem Geschäftsobjekt für Nutzdaten und den Anhangsgeschäftsobjekten

Wie Abb. 30 zu entnehmen ist, sind alle Anhänge in dem Geschäftsobjekt für Anhangscontainer enthalten. Wenn Anhänge vorhanden sind, hat das Geschäftsobjekt für Nutzdaten ein Attribut, das dem Geschäftsobjekt für Anhangscontainer entspricht.

Stellen Sie sicher, dass Ihre Geschäftsobjektstruktur anhangsbezogene Geschäftsobjekte enthält, indem Sie die folgenden Schritte ausführen:

1. Erstellen Sie eine Geschäftsobjektdefinition, die die Eigenschaften des Inhaltstyps und der Codierung (content-type-encoding) enthält, die für das Back-End-Integrationspaket erforderlich sind.
2. Erstellen Sie für jeden Anhangstyp eine Geschäftsobjektdefinition.
3. Erstellen Sie eine Geschäftsobjektdefinition für das Geschäftsobjekt für Anhangscontainer.
4. Ändern Sie die Geschäftsobjektdefinition für Ihr Geschäftsobjekt für Nutzdaten.

Jeder dieser Schritte wird in den folgenden Abschnitten beschrieben.

### Inhaltsinformationen darstellen

Zum Speichern des Inhaltstyps und der Codierung der zugeordneten Nutzdaten oder des zugeordneten Anhangs, erstellen Sie das *Geschäftsobjekt für Inhaltsinformationen*. Zur Erstellung einer Geschäftsobjektdefinition für Inhaltsinformationen erstellen Sie die in Tabelle 50 aufgeführten Attribute.

Tabelle 50. Attribute des Geschäftsobjekts für Inhaltsinformationen

Attribut	Attributtyp	Beschreibung	Schlüsselattribut?
contentType	Zeichenfolge	Der Inhaltstyp für die zugeordneten Nutzdaten oder den zugeordneten Anhang	Ja
encoding	Zeichenfolge	Die Zeichencodierung für die zugeordneten Nutzdaten oder den zugeordneten Anhang	Nein

In Abb. 30 auf Seite 146 enthält die Geschäftsobjektdefinition `contentInfo_BusObj` Attribute für den Inhaltstyp und die Codierung des Anhangs. Diese Attribute haben alle anwendungsspezifische Informationen auf Attributebene zur Angabe des Namens des zugehörigen Protokollheaders. Zum Beispiel sind für das Attribut `x-aux-sender-id` die anwendungsspezifischen Informationen wie folgt definiert:  
`name=x-aux-sender-id`

Sie können einen beliebigen Namen zur Identifikation der Geschäftsobjektdefinition für Inhaltsinformationen auswählen. Durch die anwendungsspezifischen Informationen des Anhangsgeschäftsobjekts wird bestimmt, ob es sich um einen Geschäftsobjekttyp für Inhaltstyp und Codierung (Content-Type Encoding) handelt. Abb. 30 auf Seite 146 zeigt ein Beispiel für eine Geschäftsobjektdefinition für Inhaltstyp und Codierung mit dem Namen `contentType_BusObj`.

### Anhangsdaten darstellen

Für Anhangsdaten, die nicht in ein Geschäftsobjekt konvertiert werden sollen, erstellen Sie das *Geschäftsobjekt für Standardanhänge*. Dieses Geschäftsobjekt eignet sich zur Aufnahme von im Base64-Format codierten Daten, die aus dem Transportumschlag kommen.

Zur Erstellung einer Geschäftsobjektdefinition für Standardanhänge führen Sie die folgenden Schritte aus:

1. Erstellen Sie die in Tabelle 51 gezeigten Attribute.
2. Wenn Sie ein Geschäftsobjekt für Inhaltsinformationen erstellen, fügen Sie in den anwendungsspezifischen Informationen für die Geschäftsobjektdefinition für Standardanhänge den Tag `cw_mo_bcg_content_info` hinzu, um das Attribut anzugeben, das die Inhaltsinformationen enthält.

Der Tag `cw_mo_bcg_content_info` hat folgendes Format:

```
cw_mo_bcg_content_info=contentInfoAttr
```

Dabei ist *contentInfoAttr* der Name des Attributs, in dem das Geschäftsobjekt für Anhangscontainer enthalten ist.

Tabelle 51. Attribute des Geschäftsobjekts für Standardanhänge

Attribut	Attributtyp	Beschreibung	Schlüsselattribut?
attachment	Zeichenfolge	Die einzelne Anhangsinformation. <b>Anmerkung:</b> Dieses Attribut ist das Schlüsselattribut der Geschäftsobjektdefinition.	Ja
Ein Attribut zur Aufnahme der Inhaltsinformationen	Geschäftsobjekt	Ein optionales Attribut zur Aufnahme des Geschäftsobjekts für Inhaltsinformationen, das den Inhaltstyp und die Codierung für die Anhangsdaten definiert. Dieses Attribut sollte eine einfache Kardinalität haben. <b>Anmerkung:</b> Wenn dieses Attribut <i>nicht</i> vorhanden ist, setzt der Attachment-Data-Handler die Anhangsdaten im Geschäftsobjekt nicht.  Weitere Informationen zum Format des Geschäftsobjekts für Inhaltsinformationen finden Sie im Abschnitt „Inhaltsinformationen darstellen“ auf Seite 147.	Nein

In Abb. 30 auf Seite 146 enthält die Geschäftsobjektdefinition `defaultAttach_BusObj` Attribute für den Teil der Anhangsdaten, einschließlich des Geschäftsobjekts für Inhaltsinformationen zur Angabe des zugehörigen Inhaltstyps und der zugehörigen Codierung. Der Teil der Anhangsdaten, der von diesem Standardanhangsgeschäftsobjekt dargestellt wird, verfügt über einen Inhaltstyp und eine Codierung, die durch das Attribut `contentType_attach` dargestellt werden. Daher enthält die Geschäftsobjektdefinition für Standardanhänge den folgenden Tag in ihren anwendungsspezifischen Informationen auf Geschäftsobjektebene:

```
cw_mo_bcg_content_info=contentType_attach
```

### Anhänge darstellen

Für jede Art von Anhang in Ihrem Dokument, die in ein Geschäftsobjekt konvertiert wird, müssen Sie eine separate *Anhangsgeschäftsobjektdefinition* erstellen. Die Anhangsgeschäftsobjektdefinition stellt die tatsächlichen Daten in einem Dokumentanhang dar. Zur Erstellung der Anhangsgeschäftsobjektdefinition führen Sie die folgenden Schritte aus:

1. Erstellen Sie ein Attribut für jede Einzelinformation der Anhangsdaten.  
Mögliche Attributtypen sind eine Zeichenfolge (für einfache Daten) oder eine Geschäftsobjektdefinition (für komplexe Daten).
2. Wenn für den Anhang ein Inhaltstyp und eine Codierung erforderlich sind:
  - a. Erstellen Sie ein Attribut für Inhaltstyp und Codierung.

Der Attributtyp für dieses Attribut ist die Geschäftsobjektdefinition für Inhaltstyp- und Codierungsinformationen (siehe „Inhaltsinformationen darstellen“ auf Seite 147).

- b. Fügen Sie den anwendungsspezifischen Informationen auf Geschäftsobjektebene der Anhangsgeschäftsobjektdefinition den Tag `cw_mo_bcg_content_info` hinzu, um das Attribut anzugeben, das den Inhaltstyp und die Codierung enthält.

Der Tag `cw_mo_bcg_content_info` hat folgendes Format:

```
cw_mo_bcg_content_info=contentTypeEncodingAttr
```

Dabei ist *InhaltsTypCodierungAttr* der Name des Attributs, in dem das Geschäftsobjekt für Inhaltstyp und Codierung enthalten ist.

In Abb. 30 auf Seite 146 hat das Dokument mit Nutzdaten einen Anhang, der durch die Geschäftsobjektdefinition `attachment1_BusObj` dargestellt wird. Dieser Anhang verfügt über einen Inhaltstyp und eine Codierung, die durch das Attribut `contentTypeEncoding` dargestellt werden. Daher enthält die Anhangsgeschäftsobjektdefinition den folgenden Tag in ihren anwendungsspezifischen Informationen auf Geschäftsobjektebene:

```
cw_mo_bcg_content_info=contentTypeEncoding
```

### Anhangscontainer darstellen

Der Anhangscontainer enthält alle Dokumentanhänge im Transportumschlag. Zur Darstellung des Anhangscontainers für InterChange Server erstellen Sie das Geschäftsobjekt für Anhangscontainer. Jedes Attribut im Geschäftsobjekt für Anhangscontainer stellt einen Anhang dar.

Zur Erstellung einer Geschäftsobjektdefinition für Anhangscontainer führen Sie die folgenden Schritte aus:

1. Fügen Sie für jeden Anhang im Dokument, der in ein Geschäftsobjekt zu konvertieren ist, ein Attribut hinzu.

Der Attributtyp für jedes dieser Attribute ist das zugeordnete Anhangsgeschäftsobjekt (siehe „Anhänge darstellen“ auf Seite 148). Jedes Attribut sollte eine mehrfache Kardinalität haben.

2. Fügen Sie den anwendungsspezifischen Informationen für jedes Attribut den Tag `wbic_type` hinzu, um das Attribut als Anhang zu kennzeichnen.

Der Tag `wbic_type` hat folgendes Format:

```
wbic_type=Attachment
```

**Anmerkung:** Ein Anhangsattribut kann eine mehrfache Kardinalität haben.

3. Wenn die Nutzdaten Anhangsdaten enthalten, die *nicht* in ein Geschäftsobjekt konvertiert werden sollen:

- a. Fügen Sie ein Attribut für das Standardanhangsgeschäftsobjekt hinzu.

Der Attributtyp für dieses Attribut ist die Geschäftsobjektdefinition für Standardanhänge (siehe „Anhangsdaten darstellen“ auf Seite 147). Es ist das Schlüsselattribut für das Geschäftsobjekt für Anhangscontainer. Dieses Attribut benötigt *keinen* Tag `wbic_type` in seinen anwendungsspezifischen Informationen.

**Anmerkung:** Das Geschäftsobjekt für Anhangscontainer darf nur *ein* Standardanhangsattribut enthalten. Dieses Attribut kann jedoch eine mehrfache Kardinalität aufweisen.

- b. Fügen Sie den anwendungsspezifischen Informationen auf Geschäftsobjektebene der Anhangsgeschäftsobjektdefinition den Tag `cw_mo_bcg_default_attribute` hinzu, um das Attribut anzugeben, das die Anhangsdaten enthält.

Der Tag `cw_mo_bcg_default_attribute` hat folgendes Format:

```
cw_mo_bcg_default_attribute=StandardAnhangAttr
```

Dabei ist *StandardAnhangAttr* der Name des Attributs, in dem das Standardanhangsgeschäftsobjekt enthalten ist.

**Wichtig:** Wenn kein Standardanhangsattribut vorhanden ist, kann der Attachment-Data-Handler *keine* Anhänge konvertieren, denen keine Inhaltstypzuordnung zugeordnet ist. Auch die Verarbeitung von Anhängen, die nicht in Geschäftsobjekte konvertiert werden sollen, ist in diesem Fall nicht möglich. Diese Anhänge gehen während der Konvertierung in die Geschäftsobjektdarstellung verloren.

In Abb. 30 auf Seite 146 wird der Anhangscontainer durch die Geschäftsobjektdefinition `attachContainer_BusObj` dargestellt. Diese Geschäftsobjektdefinition für Anhangscontainer verfügt über die folgenden Attribute:

- Das Attribut `attachment1` stellt den einzigen Anhang für das Dokument dar. Daher enthält die Geschäftsobjektdefinition für Anhangscontainer den folgenden Tag in ihren anwendungsspezifischen Informationen auf Attributebene:

```
wbic_type=Attachment
```

Dieser Anhang wird durch die Geschäftsobjektdefinition `attachment1_BusObj` dargestellt.

- Das Attribut `defaultAttach` stellt die Anhangsdaten dar, die *keine* Konvertierung in die Geschäftsobjektdarstellung erfordern. Daher enthält die Geschäftsobjektdefinition für Anhangscontainer den folgenden Tag in ihren anwendungsspezifischen Informationen auf Geschäftsobjektebene:

```
cw_mo_bcg_default_attribute=defaultAttach
```

## Geschäftsobjektdefinition für Nutzdaten ändern

Die Geschäftsobjektdefinition für Nutzdaten stellt die Informationen in Ihrem Dokument dar. Sie enthält ein Attribut für jede Einzelinformation, die Sie zwischen WebSphere Partner Gateway und InterChange Server übertragen. Informationen zur Erstellung der Geschäftsobjektdefinition für Nutzdaten finden Sie im Abschnitt „Geschäftsobjekt für das Dokument“ auf Seite 129.

Wenn Sie Dokumente mit Anhängen senden oder empfangen, muss Ihr Geschäftsobjekt für Nutzdaten Informationen über die Anhänge enthalten. In jedem Dokumentenfluss gibt es einen Teil mit Nutzdaten und optional mehrere Anhänge. Wenn die Nutzdaten Ihres Dokuments Anhänge enthalten, müssen Sie die Geschäftsobjektdefinition für Nutzdaten wie folgt ändern:

1. Erstellen Sie ein Attribut, das die Nutzdaten aufnimmt.

Die Verwendung der Informationen gestaltet sich möglicherweise einfacher, wenn Ihre tatsächlichen Nutzdaten in einer separaten Geschäftsobjektdefinition für Nutzdaten gespeichert werden. In diesem Fall enthält das Geschäftsobjekt für Nutzdaten auf der höchsten Ebene ein Attribut für die Nutzdaten, dessen Attributtyp die Geschäftsobjektdefinition für die tatsächlichen Nutzdaten ist.

2. Fügen Sie einen Anhangscontainer hinzu:

- a. Fügen Sie ein Attribut hinzu, das den Anhangscontainer aufnimmt.

Der Attributtyp dieses Attributs ist die Geschäftsobjektdefinition für Anhangscontainer (siehe „Anhangscontainer darstellen“ auf Seite 149). Dieses Attribut sollte eine einfache Kardinalität haben.

- b. Fügen Sie in den anwendungsspezifischen Informationen für die Geschäftsobjektdefinition für Nutzdaten den Tag `cw_mo_bcg_attachment` hinzu, um das Attribut anzugeben, in dem der Anhangscontainer enthalten ist.

Der Tag `cw_mo_bcg_attachment` hat folgendes Format:

```
cw_mo_bcg_attachment=AnhangsContainerAttr
```

Dabei ist *AnhangsContainerAttr* der Name des Attributs, in dem das Geschäftsobjekt für Anhangscontainer enthalten ist.

3. Optional können Sie den Inhaltstyp für die Nutzdaten angeben. Der Attachment-Data-Handler bestimmt anhand dieses Inhaltstyps, welcher Data-Handler zur Konvertierung der Nutzdaten zu instanziiieren ist. Wenn er einen übereinstimmenden Inhaltstyp in den Inhaltstypzuordnungen findet, instanziiert er den Data-Handler für diesen Inhaltstyp.
  - a. Fügen Sie ein Attribut für Inhaltsinformationen hinzu, wobei es sich um ein optionales Attribut zur Angabe des Inhaltstyps und der Codierung für die Nutzdaten handelt. Dieses Attribut sollte eine einfache Kardinalität haben.

**Anmerkung:** Wenn dieses Attribut *nicht* vorhanden ist, ermittelt der Attachment-Data-Handler den Data-Handler zur Konvertierung der Nutzdaten aus der Einstellung der Konfigurationseigenschaft `PayloadDataHandlerMimeType` in seinem untergeordneten Metaobjekt.

- b. Fügen Sie in den anwendungsspezifischen Informationen für die Geschäftsobjektdefinition für Nutzdaten den Tag `cw_mo_bcg_content_info` hinzu, um das Attribut anzugeben, in dem die Inhaltsinformationen enthalten sind.

Der Tag `cw_mo_bcg_content_info` hat folgendes Format:

```
cw_mo_bcg_content_info=InhaltsInfoAttr
```

Dabei ist *InhaltsInfoAttr* der Name des Attributs, in dem das Geschäftsobjekt für Inhaltsinformationen enthalten ist. Weitere Informationen zum Format des Geschäftsobjekts für Inhaltsinformationen finden Sie im Abschnitt „Inhaltsinformationen darstellen“ auf Seite 147.

4. Fügen Sie alle Konfigurationsattribute hinzu, die für Ihr Transportprotokoll erforderlich sind.

Wenn Sie zum Beispiel das JMS-Transportprotokoll verwenden, muss Ihre Geschäftsobjektdefinition für Nutzdaten das dynamische JMS-Geschäftsobjekt enthalten. Weitere Informationen finden Sie im entsprechenden Abschnitt zur Erstellung von Geschäftsobjektdefinitionen zur Unterstützung Ihres Transportprotokolls.



---

## Kapitel 9. Integration von InterChange Server über HTTP

In diesem Kapitel wird die Integration von WebSphere Partner Gateway mit WebSphere InterChange Server über das HTTP-Transportprotokoll beschrieben. Es enthält Informationen zur Konfiguration von InterChange Server (ICS) und den Adaptern, die für die Kommunikation über HTTP erforderlich sind.

**Anmerkung:** Informationen zur Konfiguration von WebSphere Partner Gateway zur Kommunikation mit InterChange Server über HTTP finden Sie im Abschnitt „WebSphere Partner Gateway für die Verwendung mit InterChange Server konfigurieren“ auf Seite 125. Allgemeine Informationen zur Konfiguration von InterChange Server finden Sie im Abschnitt „InterChange Server konfigurieren“ auf Seite 128.

Dieses Kapitel enthält die folgenden Informationen zum Senden und Empfangen von Dokumenten zwischen WebSphere Partner Gateway und WebSphere InterChange Server unter Verwendung des HTTP-Transportprotokolls:

- „HTTP-Transportprotokoll mit ICS verwenden“
- „SOAP-Dokumente über HTTP/S senden“ auf Seite 169

---

### HTTP-Transportprotokoll mit ICS verwenden

WebSphere Partner Gateway kann Dokumente an WebSphere InterChange Server (ICS) über das HTTP-Transportprotokoll senden und von diesem Produkt empfangen.

**Anmerkung:** Wenn Sie SOAP-Dokumente über das HTTP-Transportprotokoll austauschen möchten, lesen Sie den Abschnitt „SOAP-Dokumente über HTTP/S senden“ auf Seite 169.

Dieser Abschnitt enthält die folgenden Informationen zur Konfiguration von InterChange Server und den entsprechenden Adaptern zur Verwendung mit WebSphere Partner Gateway über HTTP:

- „Erforderliche Komponenten zum Dokumentenversand an ICS über den HTTP-Transport“ auf Seite 154
- „Umgebung für den HTTP-Transport mit ICS einrichten“ auf Seite 157
- „Geschäftsobjektdefinitionen für ICS über HTTP erstellen“ auf Seite 160
- „ICS-Artefakte für HTTP erstellen“ auf Seite 167

## Erforderliche Komponenten zum Dokumentenversand an ICS über den HTTP-Transport

Die Kommunikation von WebSphere Partner Gateway mit InterChange Server über das HTTP-Transportprotokoll setzt voraus, dass diese beiden Komponenten konfiguriert werden. In Tabelle 52 sind diese Konfigurationsschritte zusammengefasst.

Tabelle 52. WebSphere Partner Gateway und InterChange Server konfigurieren

Komponente	Version	Weitere Informationen
WebSphere Partner Gateway	6.1	„Dokumentenversand über das HTTP-Transportprotokoll an ICS konfigurieren“ auf Seite 125  „Dokumentenempfang von ICS über das HTTP-Transportprotokoll konfigurieren“ auf Seite 127
WebSphere InterChange Server	4.3	„ICS-Artefakte für HTTP erstellen“ auf Seite 167

Zum Senden oder Empfangen eines Dokuments zwischen WebSphere Partner Gateway und InterChange Server über das HTTP-Transportprotokoll verwenden Sie außerdem die ICS-kompatiblen Komponenten, die in Tabelle 53 aufgeführt sind.

Tabelle 53. Erforderliche Komponenten für die Übertragung von Dokumenten mit InterChange Server über HTTP

Komponente	Beschreibung	Anmerkungen und Einschränkungen
WebSphere Business Integration Adapter für HTTP (Adapter für HTTP)	Dieser Adapter gibt InterChange Server die Möglichkeit, Geschäftsobjekte mit Anwendungen auszutauschen, die Daten in Form von HTTP-Datenströmen senden oder empfangen.	Version 4.2.1 des Adapters für HTTP verwenden.
Payload-Data-Handler	Dieser Data-Handler konvertiert die Nutzdaten (payload) des Dokuments zwischen dem Dokumentformat (in der Regel XML) und der Geschäftsobjektdarstellung.	Dieser Data-Handler ist erforderlich und muss den MIME-Typ Ihres Dokuments mit Nutzdaten unterstützen.
Attachment-Data-Handler	Dieser Data-Handler behandelt Anhangsdokumente für Ihre Dokumentnachricht.	Dieser Data-Handler ist <i>nur</i> für Dokumente mit Anhängen erforderlich.

In den folgenden Abschnitten wird beschrieben, wie die Komponenten in Tabelle 53 zusammenarbeiten, um Dokumente zwischen WebSphere Partner Gateway und InterChange Server über das HTTP-Transportprotokoll zu senden und zu empfangen.

## Dokumentenversand über HTTP an ICS definieren

Beim Senden eines Dokuments von WebSphere Partner Gateway an InterChange Server über das HTTP-Transportprotokoll verwenden Sie den Adapter für HTTP, um das von WebSphere Partner Gateway als HTTP-Datenstrom gesendete Dokument abzurufen. Der Adapter leitet das Dokument anschließend an InterChange Server weiter. Abb. 31 veranschaulicht, wie WebSphere Partner Gateway Dokumente an InterChange Server über das HTTP-Transportprotokoll sendet.

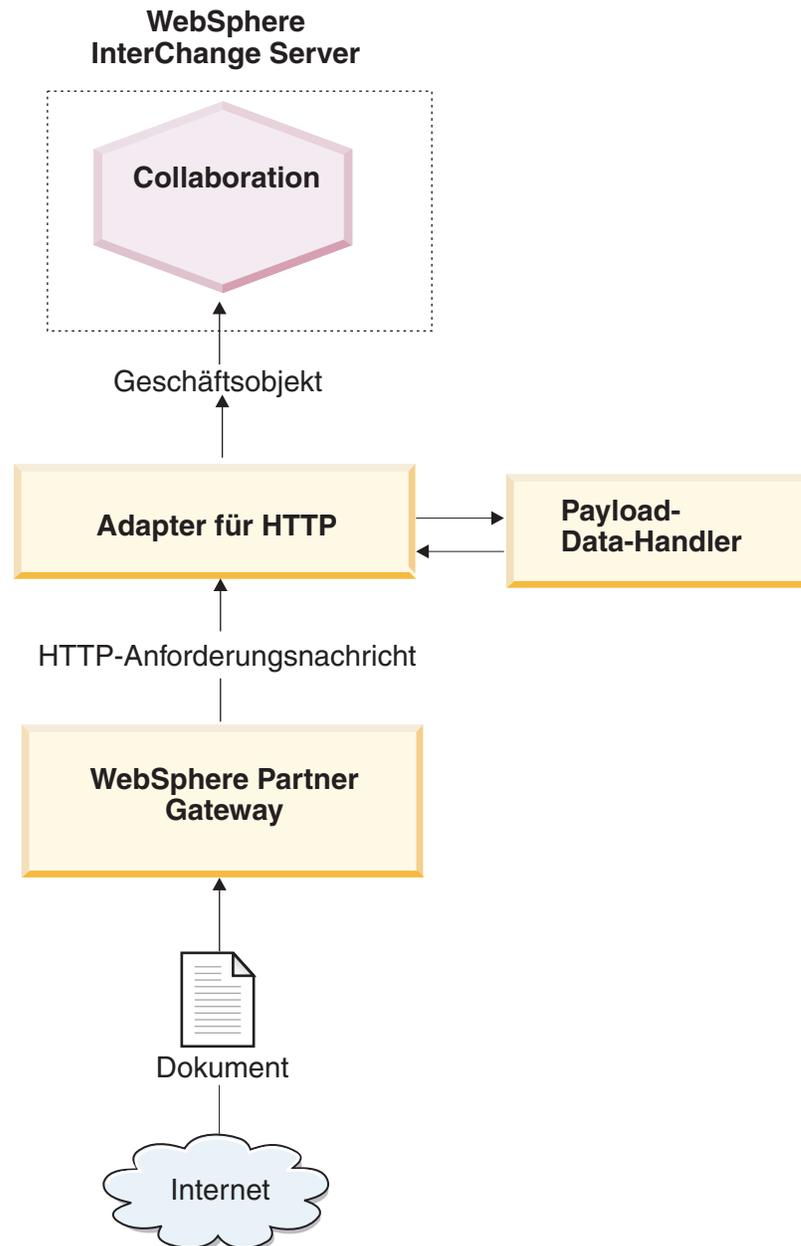


Abbildung 31. Nachrichtenfluss von WebSphere Partner Gateway an eine Collaboration über das HTTP-Transportprotokoll

## Dokumentenempfang von ICS über HTTP definieren

Beim Empfangen eines Dokuments von InterChange Server durch WebSphere Partner Gateway über das HTTP-Transportprotokoll verwenden Sie den Adapter für HTTP, der die Nachricht, die er von InterChange Server empfängt, als HTTP-Datenstrom sendet, so dass sie von WebSphere Partner Gateway abgerufen werden kann. Abb. 32 veranschaulicht, wie WebSphere Partner Gateway Dokumente von InterChange Server über das HTTP-Transportprotokoll empfängt.

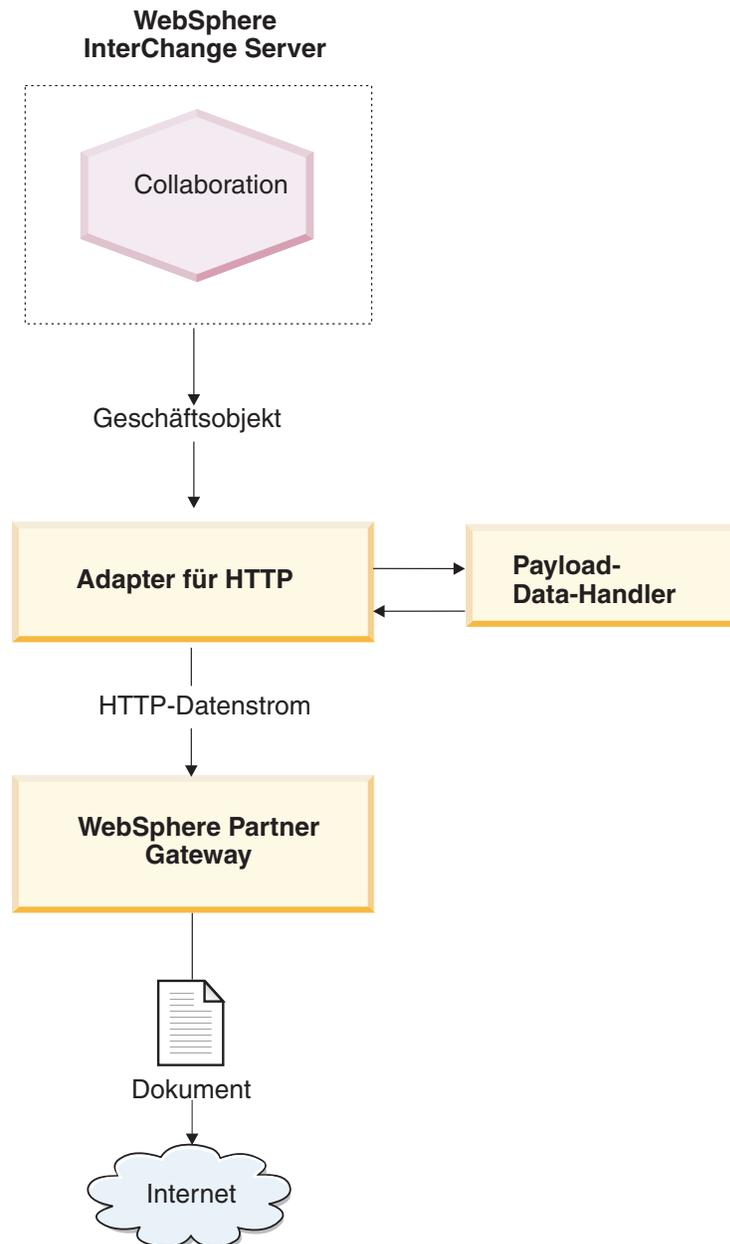


Abbildung 32. Nachrichtenfluss aus einer Collaboration an WebSphere Partner Gateway über das HTTP-Transportprotokoll

## Umgebung für den HTTP-Transport mit ICS einrichten

Da zum Senden und Empfangen von Dokumenten an bzw. von InterChange Server Adapter und Data-Handler erforderlich sind, müssen Sie die Einrichtungs- und Konfigurationsschritte am Adapter für HTTP ausführen. Informationen zur Konfiguration von WebSphere Partner Gateway zur Verwendung mit InterChange Server über HTTP finden Sie im Abschnitt „WebSphere Partner Gateway für die Verwendung mit InterChange Server konfigurieren“ auf Seite 125.

Der Adapter für HTTP ermöglicht WebSphere Partner Gateway den Austausch von Dokumenten mit InterChange Server in Form von HTTP-Nachrichten. Er unterstützt die folgenden Interaktionen mit InterChange Server:

- Bei der Anforderungsverarbeitung empfängt er das Anforderungsgeschäftsobjekt von InterChange Server, konvertiert es in einen HTTP-Datenstrom und sendet diesen Datenstrom an die angegebene URL-Adresse, an der er von WebSphere Partner Gateway empfangen werden kann.
- Für Ereignisbenachrichtigung ist er an der angegebenen URL-Adresse empfangsbereit, an die Dokumente von WebSphere Partner Gateway gesendet werden. Wenn er ein Dokument empfängt, konvertiert er es in ein Ereignisgeschäftsobjekt (mit Hilfe eines Data-Handlers) und sendet dieses Objekt an InterChange Server.

**Wichtig:** WebSphere Partner Gateway enthält den WebSphere Business Integration Adapter für HTTP *nicht*. Sie müssen dieses Produkt getrennt erwerben und entsprechend den Anweisungen im zugehörigen Handbuch *Adapter for HTTP User Guide* installieren. Vergewissern Sie sich anhand der Adapterdokumentation, dass die Version des Adapters mit der von Ihnen verwendeten Version von InterChange Server kompatibel ist.

Wenn Sie den Adapter für HTTP zur Kommunikation mit InterChange Server konfiguriert haben, führen Sie die in den folgenden Abschnitten beschriebenen Schritte aus, um diesen Adapter für HTTP-Nachrichten von WebSphere Partner Gateway empfangsbereit zu machen:

### Payload-Data-Handler angeben

Wie Abb. 32 auf Seite 156 zeigt, verwendet der Adapter für HTTP einen Data-Handler, um die von InterChange Server empfangenen Geschäftsobjekte in die entsprechenden HTTP-Datenströme zu konvertieren.

**Anmerkung:** Der Data-Handler, der vom Adapter für HTTP aufgerufen wird, konvertiert die Nutzdaten des Dokuments. Wenn Ihr Dokument in einen XML-Transportumschlag eingefügt ist (d. h. wenn es Anhänge enthält oder die Umschlagsmarkierung auf 'Ja' gesetzt ist), konfigurieren Sie den Attachment-Data-Handler als Payload-Data-Handler. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumente mit Anhängen verarbeiten“ auf Seite 133.

Um anzugeben, welcher Data-Handler zur Konvertierung der Nutzdaten zu verwenden ist, müssen Sie die im Abschnitt „Geschäftsobjekte konvertieren“ auf Seite 130 aufgeführten Schritte ausführen. Darüber hinaus müssen Sie den Adapter für HTTP zur Verwendung dieses Payload-Data-Handlers konfigurieren. Sie haben die beiden folgenden Möglichkeiten, den Payload-Data-Handler festzulegen:

- Definieren Sie in Connector Configurator die Connectorkonfigurationseigenschaft `DataHandlerMetaObjectName`, um den Namen des Data-Handler-Metaobjekts der höchsten Ebene anzugeben, das vom Adapter für HTTP zur Identifizierung von Data-Handlern verwendet wird. Stellen Sie sicher, dass das Data-Handler-Metaobjekt der höchsten Ebene in der Liste der unterstützten Geschäftsobjekte für den Adapter enthalten ist.
- Verwenden Sie im Geschäftsobjekt der höchsten Ebene das Attribut `MimeType`, das den MIME-Typ zur Angabe des Payload-Data-Handlers enthält. Weitere Informationen zu diesem Geschäftsobjekt finden Sie im Abschnitt „Geschäftsobjekt der höchsten Ebene“ auf Seite 162.

### Paketname des Protokollhandlers konfigurieren

Der Adapter für HTTP verwendet die Connectorkonfigurationseigenschaft `JavaProtocolHandlerPackages` zur Angabe des Namens der Java Protocol Handler-Pakete. Zur Integration mit WebSphere Partner Gateway müssen Sie sicherstellen, dass die Eigenschaft `JavaProtocolHandlerPackage` auf ihren Standardwert gesetzt wird:

```
com.ibm.net.ssl.internal.www.protocol
```

### Listener-Funktion für HTTP-Protokoll konfigurieren

Der Adapter für HTTP unterstützt hierarchische Konfigurationseigenschaften zur Ermittlung der Informationen, die er zur Konfiguration seiner Protokoll-Listener-Funktionen benötigt. Die Konfigurationseigenschaft der höchsten Ebene heißt `ProtocolListenerFramework`. Innerhalb dieser Ausgangseigenschaft finden sich verschiedene Ebenen von Untereigenschaften. Zur Konfiguration der Protokollhandler zur Verwendung mit dem Adapter für HTTP müssen Sie sicherstellen, dass die Eigenschaften in der Eigenschaft `ProtocolListener` konfiguriert werden, wie dies in den folgenden Schritten beschrieben wird:

1. Konfigurieren Sie eine Protokoll-Listener-Funktion mit Untereigenschaften unter der folgenden Konfigurationseigenschaft:

```
ProtocolListenerFramework
  ProtocolListeners
    HttpListener1
```

Zur Konfiguration Ihrer Protokoll-Listener-Funktionen definieren Sie die in Tabelle 54 aufgeführten Untereigenschaften.

Tabelle 54. Konfigurieren der Protokoll-Listener-Funktion

Eigenschaft	Beschreibung	Wert
Protocol	Typ der Protokoll-Listener-Funktion: <ul style="list-style-type: none"> <li>• HTTP</li> <li>• HTTPS</li> </ul>	http oder https
Host	IP-Adresse, an der die Protokoll-Listener-Funktion empfangsbereit ist	IP-Adresse des lokalen Computers, auf dem WebSphere Partner Gateway ausgeführt wird
Port	Port, an dem die Protokoll-Listener-Funktion für Anforderungen empfangsbereit ist	8080

2. Konfigurieren Sie die von der Protokoll-Listener-Funktion unterstützten URL-Konfigurationen mit Untereigenschaften unter der folgenden Konfigurationseigenschaft:

```

ProtocolListenerFramework
  ProtocolListeners
    HttpListener1
      URLsConfiguration
        URL1

```

Setzen Sie die Eigenschaft `ContextPath` auf die URI für die HTTP-Anforderungen, die von der Protokoll-Listener-Funktion empfangen werden.

**Anmerkung:** Dieses Verzeichnis muss mit dem Verzeichnis übereinstimmen, das in WebSphere Partner Gateway als Empfänger-URI angegeben ist. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumentenversand über das HTTP-Transportprotokoll an ICS konfigurieren“ auf Seite 125.

3. Wenn Ihr Dokument Anhänge enthält, müssen Sie eine Transformation für die Protokoll-Listener-Funktion konfigurieren, indem Sie Untereigenschaften der folgenden Konfigurationseigenschaft definieren:

```

ProtocolListenerFramework
  ProtocolListeners
    HttpListener1
      URLsConfiguration
        URL1
          TransformationRules
            TransformationRule1

```

Zur Konfiguration der Anhangstransformation für Ihre Protokoll-Listener-Funktion definieren Sie die in Tabelle 55 aufgeführten Untereigenschaften. Sie benötigen eine Transformationsregel für jede Instanz des Attachment-Data-Handlers, die Sie verwenden. Weitere Informationen zum Attachment-Data-Handler finden Sie im Abschnitt „Dokumente mit Anhängen verarbeiten“ auf Seite 133.

Tabelle 55. Konfigurieren der Anhangstransformation für die Protokoll-Listener-Funktion

Eigenschaft	Beschreibung	Wert
<code>ContentType</code>	Der Inhaltstyp der mit einem Data-Handler zu transformierenden Daten	Der den Anhangsdaten zugeordnete Inhaltstyp
<code>MimeType</code>	MIME-Typ zur Angabe des aufzurufenden Data-Handlers	Der Instanz des Attachment-Data-Handlers zugeordneter MIME-Typ
<code>Charset</code>	Der beim Transformieren von Daten des angegebenen Inhaltstyps zu verwendende Zeichensatz	Zeichensatz für die Anhangsdaten

Weitere Informationen zu diesen Eigenschaften finden Sie im Handbuch *Adapter for HTTP User Guide*.

## Geschäftsobjektdefinitionen für ICS über HTTP erstellen

Der Adapter für HTTP sendet Ihr Dokument an InterChange Server (und empfängt es von dort) in Form eines Geschäftsobjekts für Nutzdaten. Der Adapter für HTTP ruft den Payload-Data-Handler auf, um dieses Geschäftsobjekt wie folgt zu verarbeiten, wenn er ein WebSphere Partner Gateway-Dokument empfängt bzw. sendet:

- Bei der Anforderungsverarbeitung konvertiert der Payload-Data-Handler das Anforderungsgeschäftsobjekt in den entsprechenden HTTP-Datenstrom.
- Bei der Ereignisbenachrichtigung konvertiert der Data-Handler den HTTP-Datenstrom in ein Ereignisgeschäftsobjekt.

Daher müssen Sie die in Tabelle 56 gezeigten Geschäftsobjektdefinitionen erstellen, um die Struktur des Geschäftsobjekts für Nutzdaten darzustellen, die der Adapter für HTTP erwartet.

Tabelle 56. Geschäftsobjektdefinitionen für den Adapter für HTTP

Bedingung	Geschäftsobjektdefinition	Weitere Informationen
Wenn Sie für Ihre Nachricht den Pakettyp 'None' oder 'Backend Integration' verwenden <i>und</i> Ihre Dokumente <i>keine</i> Anhänge haben	Geschäftsobjekt für Nutzdaten: <ul style="list-style-type: none"> <li>• Geschäftsobjekt der höchsten Ebene</li> <li>• Anforderungsgeschäftsobjekt</li> <li>• Antwortgeschäftsobjekt (optional)</li> <li>• Fehlergeschäftsobjekt (optional)</li> </ul>	„Struktur des Geschäftsobjekts für Nutzdaten für ICS über HTTP erstellen“ auf Seite 161
Wenn Sie 'Backend Integration' für Ihre Nachricht verwenden	Fügen Sie dem Geschäftsobjekt für Nutzdaten die Geschäftsobjekte hinzu, die die Nachrichtenheaderinformationen enthalten: <ul style="list-style-type: none"> <li>• Dynamisches Metaobjekt</li> <li>• Geschäftsobjekt für HTTP-Eigenschaften</li> </ul>	„HTTP-Headerinformationen der Transportebene für ICS erstellen“ auf Seite 165.
Wenn das Dokument Anhänge enthält	Sie müssen außerdem zusätzliche Geschäftsobjekte zur Darstellung der Anhänge erstellen.	„Anhangsbezogene Geschäftsobjektdefinitionen erstellen“ auf Seite 145

## Struktur des Geschäftsobjekts für Nutzdaten für ICS über HTTP erstellen

Der Adapter für HTTP erwartet eine Geschäftsobjektstruktur für Nutzdaten, die aus folgenden Geschäftsobjekten besteht:

- Ein Geschäftsobjekt der höchsten Ebene
- Ein Anforderungsgeschäftsobjekt
- Ein Fehlergeschäftsobjekt (optional)
- Ein Antwortgeschäftsobjekt (optional)

Abb. 33 zeigt ein Beispiel einer Geschäftsobjektstruktur für eine Geschäftsobjektdefinition für Nutzdaten zur Verwendung mit InterChange Server über das HTTP-Transportprotokoll.

**Anmerkung:** Eine detaillierte Beschreibung dieser Geschäftsobjektstruktur finden Sie im Handbuch *Adapter for HTTP User Guide*.

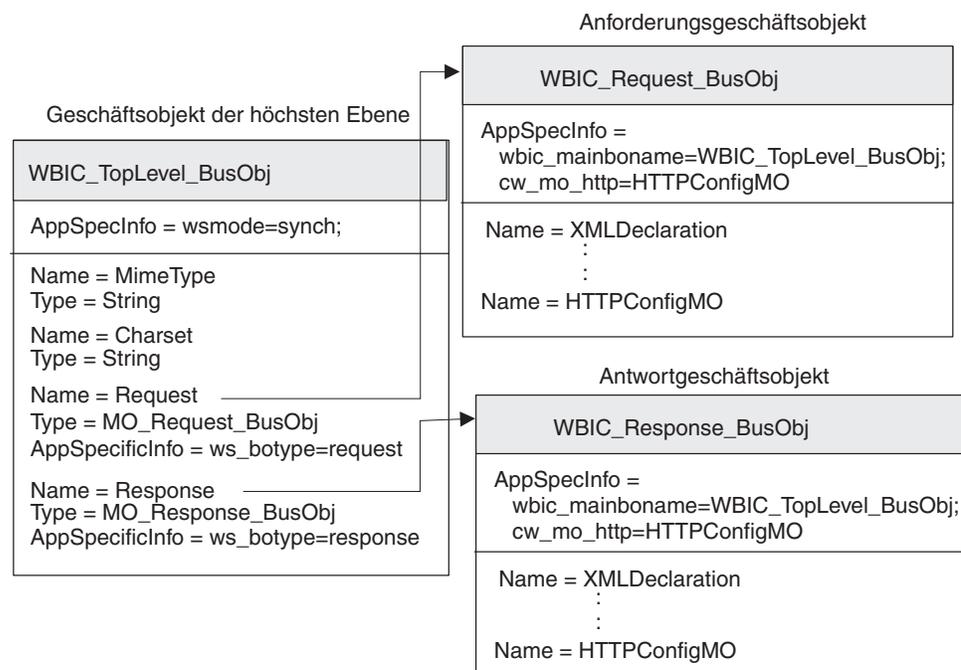


Abbildung 33. Geschäftsobjektstruktur für das HTTP-Geschäftsobjekt für Nutzdaten für ICS

**Geschäftsobjekt der höchsten Ebene:** Das Geschäftsobjekt der höchsten Ebene ist ein Wrapper für die Anforderungs- und Antwortgeschäftsobjekte. Sie müssen eine Geschäftsobjektdefinition für dieses Geschäftsobjekt erstellen. Tabelle 57 gibt eine Übersicht zu den Attributen der Geschäftsobjektdefinition der höchsten Ebene.

*Tabelle 57. Attribute des Geschäftsobjekts der höchsten Ebene*

Attribut	Attributtyp	Beschreibung
MimeType	Zeichenfolge	Definiert den Inhaltstyp und das Format der Daten, die an die URL-Adresse übergeben werden.
Charset	Zeichenfolge	Dient zur Bestimmung des aufzurufenden Data-Handlers.
Request	Geschäftsobjekt	Untergeordnetes Geschäftsobjekt, das die Anforderungsnachricht darstellt. Der Zweck dieses Geschäftsobjekts hängt davon ab, ob es an einer Anforderungsverarbeitung oder einer Ereignisbenachrichtigung beteiligt ist. Weitere Informationen zur Struktur dieses Geschäftsobjekts finden Sie im Abschnitt „Anforderungsgeschäftsobjekt“ auf Seite 163.
Response	Geschäftsobjekt	Untergeordnetes Geschäftsobjekt, das die Antwortnachricht darstellt (wenn Sie eine Antwort erwarten). Der Zweck dieses Geschäftsobjekts hängt davon ab, ob es an einer Anforderungsverarbeitung oder einer Ereignisbenachrichtigung beteiligt ist. Weitere Informationen zur Struktur dieses Geschäftsobjekts finden Sie im Abschnitt „Antwortgeschäftsobjekt“ auf Seite 164.

**Anmerkung:** Wenn Sie den Adapter für HTTP mit WebSphere Partner Gateway verwenden, müssen Sie *keine* Fehlergeschäftsobjekte in Ihr Geschäftsobjekt der höchsten Ebene einfügen.

In Tabelle 58 sind die anwendungsspezifischen Informationen zusammengefasst, die in der Geschäftsobjektdefinition der höchsten Ebene enthalten sein können.

*Tabelle 58. Anwendungsspezifische Informationen für die Geschäftsobjektdefinition der höchsten Ebene*

Anwendungsspezifische Information	Tag	Beschreibung
Geschäftsobjektebene	ws_mode	Definiert, ob die Interaktion synchron oder asynchron erfolgt.
Attributebene	ws_botype	Definiert, welches Attribut das Anforderungs- bzw. Antwortgeschäftsobjekt enthält.

Eine vollständige Beschreibung der Struktur des Geschäftsobjekts der höchsten Ebene und der zugehörigen anwendungsspezifischen Informationen finden Sie im Handbuch *Adapter for HTTP User Guide*.

**Anforderungsgeschäftsobjekt:** Das Anforderungsgeschäftsobjekt enthält die Daten, die an die URL-Adresse zu übergeben sind. Es stellt die HTTP-Anforderungsnachricht dar. Der Zweck dieses Anforderungsgeschäftsobjekts hängt wie folgt davon ab, an welchem InterChange Server-Vorgang es beteiligt ist:

- Bei der Ereignisbenachrichtigung (Senden eines Dokuments an InterChange Server) enthält das Anforderungsgeschäftsobjekt die Anforderungsnachricht von WebSphere Partner Gateway, die das Ereignis darstellt, das an InterChange Server zu senden ist.
- Bei der Anforderungsverarbeitung (Empfangen eines Dokuments von InterChange Server) enthält das Anforderungsgeschäftsobjekt die Anforderung, die InterChange Server an WebSphere Partner Gateway sendet.

**Anmerkung:** Das Geschäftsobjekt der höchsten Ebene gibt die zugehörigen untergeordneten Geschäftsobjekte als Anforderungs- und Antwortgeschäftsobjekte an. Diese Struktur wird jedoch bei der Anforderungsverarbeitung *und* bei der Ereignisbenachrichtigung verwendet.

Eine grundsätzliche Beschreibung der Struktur des Anforderungsobjekts finden Sie im Handbuch *Adapter for HTTP User Guide*. Zur Verwendung mit WebSphere Partner Gateway müssen Sie zwei Anpassungen an der Struktur der Anforderungsgeschäftsobjektdefinition vornehmen:

- Wenn das von WebSphere Partner Gateway an InterChange Server gesendete Dokument mit dem Back-End-Integrationspaket arbeitet, müssen Sie der Anforderungsgeschäftsobjektdefinition ein besonderes Attribut zur Angabe des Metaobjekts für die HTTP-Protokollkonfiguration hinzufügen.  
Dieses Attribut liefert Konfigurationsdaten für die Header der Transportebene der Nachricht. Weitere Informationen hierzu finden Sie im Abschnitt „HTTP-Headerinformationen der Transportebene für ICS erstellen“ auf Seite 165.
- Fügen Sie den anwendungsspezifischen Informationen auf Geschäftsobjektebene der Anforderungsgeschäftsobjektdefinition die in Tabelle 59 aufgeführten Tags hinzu.

*Tabelle 59. Tags in den anwendungsspezifischen Informationen des Anforderungsgeschäftsobjekts*

Tag in den anwendungsspezifischen Informationen	Beschreibung	Erforderlich
ws_tloname	Gibt den Namen des Geschäftsobjekts der höchsten Ebene an.	Nur erforderlich, wenn die Geschäftsobjektdefinition an einer Ereignisbenachrichtigung beteiligt ist.
cw_mo_http	Gibt das Metaobjekt für die HTTP-Protokollkonfiguration an, das die Felder für den HTTP-Header auf Transportebene enthält. Weitere Informationen hierzu finden Sie im Abschnitt „HTTP-Headerinformationen der Transportebene für ICS erstellen“ auf Seite 165.	Nur erforderlich, wenn Sie 'Backend Integration' verwenden.

**Anmerkung:** Wenn Sie den Attachment-Data-Handler zur Verarbeitung von Dokumenten verwenden, die in einen XML-Transportumschlag eingefügt sind, müssen Sie Ihr Anforderungsgeschäftsobjekt zur Aufnahme der Anhänge ändern, wie dies im Abschnitt „Anhangsbezogene Geschäftsobjektdefinitionen erstellen“ auf Seite 145 beschrieben ist.

**Antwortgeschäftsobjekt:** Das Antwortgeschäftsobjekt enthält die Daten, die von der URL-Adresse zu empfangen sind. Es enthält Attribute für die verschiedenen XML-Tags in der Antwortnachricht. Der Zweck dieses Antwortgeschäftsobjekts hängt wie folgt davon ab, an welchem InterChange Server-Vorgang es beteiligt ist:

- Bei der Ereignisbenachrichtigung enthält das Antwortgeschäftsobjekt die Antwortnachricht, die aus der Collaboration in InterChange Server gesendet wird.
- Bei der Anforderungsverarbeitung enthält das Antwortgeschäftsobjekt die Informationen aus WebSphere Partner Gateway als Antwort auf die Anforderung, die von InterChange Server gesendet wurde.

Unabhängig davon, ob die Antwort Teil einer Ereignisbenachrichtigung oder einer Anforderungsverarbeitung ist, wird ein Antwortgeschäftsobjekt *nur* gesendet, wenn der Austausch zwischen WebSphere Partner Gateway und InterChange Server *synchron* erfolgt und eine Geschäftsantwort als Reaktion auf die Anforderung erwartet wird.

Eine grundsätzliche Beschreibung der Struktur des Fehlergeschäftsobjekts (Fault Business Object) finden Sie im Handbuch *Adapter for HTTP User Guide*. Zur Verwendung mit WebSphere Partner Gateway müssen Sie zwei Anpassungen an der Struktur der Anforderungsgeschäftsobjektdefinition vornehmen:

- Wenn das von WebSphere Partner Gateway an InterChange Server gesendete Dokument mit dem Back-End-Integrationspaket arbeitet, müssen Sie der Antwortgeschäftsobjektdefinition ein besonderes Attribut zur Angabe des Metaobjekts für die HTTP-Protokollkonfiguration hinzufügen.  
Dieses Attribut liefert Konfigurationsdaten für die Header der Transportebene der Nachricht. Weitere Informationen hierzu finden Sie im Abschnitt „HTTP-Headerinformationen der Transportebene für ICS erstellen“ auf Seite 165.
- Fügen Sie den anwendungsspezifischen Informationen auf Geschäftsobjektebene der Antwortgeschäftsobjektdefinition die in Tabelle 59 auf Seite 163 aufgeführten Tags hinzu.
- Fügen Sie im Geschäftsobjekt der höchsten Ebene das Attribut `ws_botype` den anwendungsspezifischen Informationen auf Attributebene für das Attribut hinzu, das dem Antwortgeschäftsobjekt entspricht.

Dieser Tag hat folgende Syntax:

```
ws_botype=response
```

Wenn der Austausch zwischen WebSphere Partner Gateway und InterChange Server *asynchron* erfolgt, erwartet WebSphere Partner Gateway *keine* Antwort, so dass Sie kein Antwortgeschäftsobjekt erstellen müssen.



**Geschäftsobjekt für die benutzerdefinierten Eigenschaften erstellen:** Der Adapter für HTTP unterstützt ein *Geschäftsobjekt für benutzerdefinierte Eigenschaften*, in dem angepasste Eigenschaften im Metaobjekt für die HTTP-Protokollkonfiguration angegeben werden können. WebSphere Partner Gateway verwendet dieses Geschäftsobjekt zur Aufnahme von HTTP-Eigenschaften, die für das Back-End-Integrationspaket erforderlich sind. Es kann außerdem das Attribut Content-Type, das den Content-Type-Header angibt, der in der Anforderungsnachricht einzufügen ist, und das Attribut content-length enthalten, das die Länge der Nachricht in Byte angibt. Die einzelnen gültigen Transportheaderfelder werden in Tabelle 8 auf Seite 26 beschrieben.

Zur Erstellung einer Geschäftsobjektdefinition für benutzerdefinierte Eigenschaften für die HTTP-Headerfelder führen Sie die folgenden Schritte aus:

1. Erstellen Sie innerhalb der Geschäftsobjektdefinition ein Attribut für jedes einzelne Transportheaderfeld.  
Alle Attribute müssen den Attributtyp 'Zeichenfolge' haben. Sie können das Attribut mit dem exakten Namen der HTTP-Eigenschaft (wie in der Spalte Headerfeld von Tabelle 8 auf Seite 26 aufgeführt) benennen.
2. Fügen Sie für jedes Attribut im Geschäftsobjekt für HTTP-Eigenschaften anwendungsspezifische Informationen hinzu, um den Zweck des zugeordneten Attributs anzugeben.

Diese anwendungsspezifischen Informationen auf Attributebene haben folgendes Format:

```
ws_prop_name=HTTPeigenschaft
```

Dabei steht *HTTPeigenschaft* für einen der Werte in der Spalte Headerfeld von Tabelle 8 auf Seite 26.

In Abb. 34 auf Seite 165 enthält die Geschäftsobjektdefinition `HttpProps_BusObj` Attribute für die verschiedenen Transportheaderfelder. Diese Attribute haben alle anwendungsspezifische Informationen auf Attributebene zur Angabe des Namens des zugehörigen Protokollheaders. Zum Beispiel sind für das Attribut `x-aux-sender-id` die anwendungsspezifischen Informationen wie folgt definiert:

```
ws_prop_name=x-aux-sender-id
```

**Metaobjekt für die HTTP-Protokollkonfiguration erstellen:** Bei der Ereignisbenachrichtigung kann das Anforderungs-, Antwort- oder Fehlergeschäftsobjekt ein dynamische Metaobjekt für die *HTTP-Protokoll-Konfiguration* enthalten, das Konfigurationsinformationen (z. B. Headerinformationen) enthält.

Eine grundsätzliche Beschreibung der Struktur des Geschäftsobjekts für die HTTP-Protokollkonfiguration finden Sie im Handbuch *Adapter for HTTP User Guide*. Zur Verwendung mit WebSphere Partner Gateway müssen Sie die folgenden Anpassungen an der Struktur der Geschäftsobjektdefinition für die HTTP-Protokollkonfiguration vornehmen:

1. Erstellen Sie innerhalb der Geschäftsobjektdefinition ein Attribut für jedes erforderliche Feld.

Alle Attribute müssen den Attributtyp 'Zeichenfolge' haben.

**Anmerkung:** Eine vollständige Liste der Attribute im Metaobjekt für die HTTP-Protokollkonfiguration finden Sie im Handbuch *Adapter for HTTP User Guide*.

2. Fügen Sie dieser Geschäftsobjektdefinition das Attribut `UserDefinedProperties` hinzu.

Der Attributtyp dieses Attributs ist die Geschäftsobjektdefinition für das Geschäftsobjekt für die benutzerdefinierten Eigenschaften (siehe „Geschäftsobjekt für die benutzerdefinierten Eigenschaften erstellen“ auf Seite 166).

Zum Beispiel enthält die Geschäftsobjektdefinition `HttpConfigM0_BusObj` in Abb. 34 auf Seite 165 das Attribut `UserDefinedProperties`, dessen Attributtyp `HttpProps_BusObj` ist.

**Anforderungsgeschäftsobjektdefinition ändern:** Die Anforderungsgeschäftsobjektdefinition stellt die Informationen dar, die von WebSphere Partner Gateway angefordert werden. Informationen zur Erstellung des Anforderungsgeschäftsobjekts finden Sie im Abschnitt „Anforderungsgeschäftsobjekt“ auf Seite 163. Zum Einfügen des dynamischen Metaobjekts in die Struktur Ihres Geschäftsobjekts für Nutzdaten müssen Sie die folgenden Änderungen an Ihrer Anforderungsgeschäftsobjektdefinition vornehmen:

1. Fügen Sie Ihrer Anforderungsgeschäftsobjektdefinition ein Attribut hinzu, in dem das Metaobjekt für die HTTP-Protokollkonfiguration enthalten ist.

Der Attributtyp dieses Attributs ist die Geschäftsobjektdefinition für das Metaobjekt für die HTTP-Protokollkonfiguration (siehe „Metaobjekt für die HTTP-Protokollkonfiguration erstellen“ auf Seite 166).

2. Fügen Sie den anwendungsspezifischen Informationen auf Geschäftsobjekt-ebene Ihrer Anforderungsgeschäftsobjektdefinition den Tag `cw_mo_http` hinzu, um das Attribut anzugeben, in dem das Metaobjekt für die HTTP-Protokollkonfiguration enthalten ist.

Der Tag `cw_mo_http` hat folgendes Format:

```
cw_mo_http=HttpKonfigMetaObjAttr
```

Dabei ist `HttpKonfigMetaObjAttr` der Name des Attributs in dem Anforderungsgeschäftsobjekt, in dem das Metaobjekt für die HTTP-Protokollkonfiguration enthalten ist.

In Abb. 34 auf Seite 165 wurde der Anforderungsgeschäftsobjektdefinition `hub_HttpRequest_BusObj` zum Beispiel ein Attribut mit dem Namen `HttpConfigM0` hinzugefügt. Dieses Attribut enthält das dynamische Metaobjekt, das ein untergeordnetes Geschäftsobjekt des Typs `HttpConfigM0_BusObj` ist. Darüber hinaus wurden die anwendungsspezifischen Informationen des Anforderungsgeschäftsobjekts modifiziert, um den folgenden Tag `cw_mo_http` zur Angabe dieses dynamischen Metaobjekts einzufügen:

```
cw_mo_http=HttpConfigM0
```

## ICS-Artefakte für HTTP erstellen

Zur Konfiguration von InterChange Server zur Kommunikation mit WebSphere Partner Gateway über das HTTP-Transportprotokoll müssen Sie die in Tabelle 60 aufgeführten InterChange Server-Artefakte erstellen.

Tabelle 60. Artefakte für die Kommunikation mit ICS über das HTTP-Transportprotokoll

ICS-Artefakt	Zweck	Weitere Informationen
Geschäftsobjektdefinitionen	Stellen das Dokument dar.	„Geschäftsobjektdefinitionen für ICS über HTTP erstellen“ auf Seite 160
Connectorobjekt	Stellt den Adapter für HTTP während der Ausführung dar.	„HTTP-Connectorobjekt erstellen“
Collaboration-Schablone und Collaboration-Objekt	Stellen den Geschäftsprozess dar, der von InterChange Server zur Verarbeitung des Dokuments verwendet wird.	„Collaborations zur Kommunikation mit dem Adapter für HTTP binden“

## HTTP-Connectorobjekt erstellen

Zum Aufruf einer Instanz des Adapters für HTTP während der Ausführung müssen Sie im Tool 'System Manager' die folgenden Schritte ausführen:

### 1. Erstellen Sie die Connectorobjekte:

- Erstellen Sie ein Connectorobjekt, um eine Instanz des Adapters für HTTP darzustellen.

**Anmerkung:** Stellen Sie auf der Registerkarte **Unterstützte Geschäftsobjekte** von Connector Configurator sicher, dass Sie alle Geschäftsobjektdefinitionen angeben, die Sie zur Verwendung mit dem Adapter für HTTP erstellt haben. Eine Beschreibung dieser Geschäftsobjektdefinitionen finden Sie im Abschnitt „Geschäftsobjektdefinitionen für ICS über HTTP erstellen“ auf Seite 160.

- Wenn dies für Ihre Collaboration erforderlich ist, erstellen Sie ein Connectorobjekt für den Port-Connector.

### 2. Konfigurieren Sie die Connectorobjekte.

Informationen zur Konfiguration Ihres Connectorobjekts für den Adapter für HTTP zur Verwendung mit WebSphere Partner Gateway finden Sie im Abschnitt „Umgebung für den HTTP-Transport mit ICS einrichten“ auf Seite 157.

## Collaborations zur Kommunikation mit dem Adapter für HTTP binden

Wie im Abschnitt „Collaborations erstellen“ auf Seite 132 beschrieben, muss während der Ausführung ein Collaboration-Objekt vorhanden sein, damit InterChange Server ermitteln kann, wo Geschäftsobjekte zu empfangen sind und wohin Geschäftsobjekte zu senden sind. Bei der Erstellung des Collaboration-Objekts für die Collaboration, die Informationen mit Hilfe des Adapters für HTTP an WebSphere Partner Gateway sendet und von WebSphere Partner Gateway empfängt, binden Sie die Collaboration-Ports wie folgt:

- Für die Anforderungsverarbeitung stellen Sie den Empfangsport ('To'), der Anforderungen an WebSphere Partner Gateway sendet, auf das Connectorobjekt ein, das Sie für den Adapter für HTTP erstellt haben. Das heißt, der Adapter für HTTP ist der *Zieladapter*.
- Für die Ereignisbenachrichtigung stellen Sie den Absenderport ('From'), der Ereignisse von WebSphere Partner Gateway empfängt, auf das Connectorobjekt ein, das Sie für den Adapter für HTTP erstellt haben. Das heißt, der Adapter für HTTP ist der *Quellenadapter*.

## SOAP-Dokumente über HTTP/S senden

SOAP-Dokumente unterscheiden sich von anderen Typen von Dokumenten, die über HTTP/S ausgetauscht werden. Sie verwenden den Standardadapter für Web-Services, der den SOAP-Data-Handler aufruft, um SOAP-Nachrichten in Geschäftsobjekte und Geschäftsobjekte in SOAP-Nachrichten zu transformieren. Im folgenden Abschnitt wird beschrieben, wie SOAP-Dokumente über das HTTP-Transportprotokoll zwischen WebSphere Partner Gateway und WebSphere InterChange Server gesendet und empfangen werden.

**Anmerkung:** Informationen zum Senden und Empfangen anderer Dokumente als SOAP-Dokumente zwischen WebSphere Partner Gateway und WebSphere InterChange Server über das HTTP-Transportprotokoll finden Sie im Abschnitt „HTTP-Transportprotokoll mit ICS verwenden“ auf Seite 153.

In der Dokumentation zum Adapter für Web-Services finden Sie Informationen zur Geschäftsobjektstruktur und zum WSDL Object Discovery Agent (ODA), einem bei der Entwicklung verwendeten Tool, mit dem Sie SOAP-Geschäftsobjekte generieren können, die Informationen zu den Empfänger-Web-Services enthalten.

Wie im Handbuch *WebSphere Partner Gateway Hubkonfiguration* beschrieben, müssen Sie einen Empfänger konfiguriert haben, der Web-Service-Aufrufe von einem Back-End-System (dem Web-Services-Empfänger) empfängt. Außerdem muss ein Empfänger konfiguriert worden sein, der Web-Service-Aufrufe von einem externen Partner (externer Web-Services-Empfänger) empfängt.

### Erforderliche Komponenten für den Dokumentenversand und -empfang

Um ein SOAP-Dokument über das HTTP-Transportprotokoll von WebSphere Partner Gateway an InterChange Server zu senden, können Sie die in Tabelle 61 aufgeführte Komponente verwenden.

Tabelle 61. Erforderliche Komponenten für den SOAP-Dokumentenversand an InterChange Server über HTTP

Komponente	Beschreibung	Anmerkungen und Einschränkungen
WebSphere Business Integration Adapter für Web-Services	Dieser Adapter gibt InterChange Server die Möglichkeit, Geschäftsobjekte mit Anwendungen auszutauschen, die Daten in Form von HTTP-Nachrichten senden oder empfangen.	<ol style="list-style-type: none"><li>1. Dieser Adapter kann <i>ausschließlich</i> mit SOAP-Dokumenten verwendet werden.</li><li>2. Stellen Sie sicher, dass Sie den Adapter für Web-Services 3.4.0 (oder höher) verwenden. Vergewissern Sie sich anhand der Informationen im Handbuch <i>Adapter for Web Services User Guide</i>, dass die Version des Adapters mit der von Ihnen verwendeten Version von WebSphere InterChange Server kompatibel ist.</li></ol>

**Anmerkung:** Wenn ein SOAP-Dokument Anhänge enthält, müssen diese Anhänge nicht mit dem Attachment-Data-Handler verarbeitet werden.

## Web-Service-Aufruf durch externe Partner

Die folgenden Schritte werden ausgeführt, wenn ein externer Partner eine Anforderung für eine Collaboration sendet, die als ein vom internen Partner bereitgestellter Web-Service zugänglich ist:

1. Der externe Partner sendet eine SOAP-Anforderungsnachricht an das Ziel, das in dem für die Collaboration generierten WSDL-Dokument angegeben ist. Beachten Sie, dass der im WSDL-Dokument angegebene Endpunkt der Web-Services-Empfänger (URL-Adresse) von WebSphere Partner Gateway und nicht der tatsächliche Endpunkt ist.
2. WebSphere Partner Gateway empfängt die Nachricht und leitet sie an den Adapter für Web-Services weiter.
3. Der Adapter für Web-Services sendet die SOAP-Nachricht an den SOAP-Data-Handler, um die SOAP-Nachricht in ein Geschäftsobjekt zu konvertieren. Der Adapter ruft die als Web-Service bereitgestellte Collaboration auf.
4. Wenn es sich um eine Anforderungs-/Antwortoperation handelt, gibt die Collaboration ein SOAP-Antwortgeschäftsobjekt (bzw. ein entsprechendes Fehlergeschäftsobjekt) zurück.
5. Wenn die Collaboration ein SOAP-Antwortgeschäftsobjekt (bzw. ein entsprechendes Fehlergeschäftsobjekt) zurückgegeben hat, ruft der Adapter für Web-Services den SOAP-Data-Handler auf, um das SOAP-Antwortgeschäftsobjekt (bzw. das entsprechende Fehlergeschäftsobjekt) in eine SOAP-Antwortnachricht zu konvertieren. Der Adapter gibt die Antwort an WebSphere Partner Gateway zurück. Wenn die Collaboration kein SOAP-Antwortgeschäftsobjekt (bzw. kein entsprechendes Fehlergeschäftsobjekt) zurückgegeben hat, gibt der Adapter für Web-Services den entsprechenden HTTP-Antwortstatuscode zurück.
6. WebSphere Partner Gateway leitet die Antwort an den Web-Service weiter.

## Web-Service-Aufruf durch den internen Partner

Die öffentliche WSDL, die von WebSphere Partner Gateway bereitgestellt wird, kann zur Erstellung von Geschäftsobjekten mit dem WSDL-ODA verwendet werden. Es ist wichtig zu beachten, dass bei der Angabe des Web-Service durch einen externen Partner zur Verwendung durch den internen Partner die öffentliche URL-Adresse, die vom internen Partner zum Aufrufen des Web-Service verwendet wird, die folgende Abfragezeichenfolge enthalten muss:

*?to=<Geschäfts-ID des externen Partners, der als Web-Service-Provider fungiert>*

Zum Beispiel teilt die folgende Adresse WebSphere Partner Gateway mit, dass der Provider des Web-Service der Partner mit der Geschäfts-ID 123456789 ist:

`http://<hub-ip-adresse>/bcgreceiver/Receiver?to=123456789`

Der WSDL-ODA fügt die Abfragezeichenfolge dem Standardwert des URL-Attributs des Web-Service-Geschäftsobjekts der höchsten Ebene nicht hinzu.

Die folgenden Schritte werden ausgeführt, wenn eine Collaboration eine Anforderung (an den Adapter für Web-Services) sendet, um einen Web-Service eines externen Partners aufzurufen:

1. Die Collaboration sendet eine Serviceaufrufanforderung an den Adapter, der wiederum den SOAP-Data-Handler aufruft, um das Geschäftsobjekt in eine SOAP-Anforderungsnachricht zu konvertieren.
2. Der Adapter ruft den Web-Service auf, indem er die SOAP-Nachricht an den externen Web-Services-Empfänger (URL-Adresse) in WebSphere Partner Gateway sendet.
3. WebSphere Partner Gateway agiert als Proxy und sendet die SOAP-Nachricht an den Endpunkt, der dem Empfänger-Web-Service (externer Partner) entspricht. Dadurch wird der Web-Service aufgerufen.
4. Der aufgerufene Web-Service empfängt die SOAP-Anforderungsnachricht und führt die angeforderte Verarbeitung aus.
5. Der aufgerufene Web-Service sendet eine SOAP-Antwortnachricht (bzw. eine Fehlermeldung). Im Fall einer Operation, die nur in eine Richtung ausgeführt wird, wird der entsprechende HTTP-Statuscode zurückgegeben.
6. Bei einem Web-Service mit Anforderungs-/Antwortverarbeitung leitet WebSphere Partner Gateway die SOAP-Antwortnachricht (bzw. die Fehlermeldung) an den Adapter weiter, der den Data-Handler aufruft, um sie in ein Antwort- bzw. Fehlergeschäftobjekt zu konvertieren. Der Connector gibt das SOAP-Antwortgeschäftobjekt (bzw. das Fehlergeschäftobjekt) an die Collaboration zurück.



---

## Kapitel 10. Integration von InterChange Server über JMS

In diesem Kapitel wird die Integration von WebSphere Partner Gateway mit WebSphere InterChange Server über das JMS-Transportprotokoll beschrieben. Es enthält Informationen zur Konfiguration von InterChange Server und der Adapter, die für die Kommunikation über JMS erforderlich sind.

**Anmerkung:** Informationen zur Konfiguration von WebSphere Partner Gateway zur Kommunikation mit InterChange Server über JMS finden Sie im Abschnitt „WebSphere Partner Gateway für die Verwendung mit InterChange Server konfigurieren“ auf Seite 125. Allgemeine Informationen zur Konfiguration von InterChange Server finden Sie im Abschnitt „InterChange Server konfigurieren“ auf Seite 128.

Dieses Kapitel enthält die folgenden Informationen zum Senden und Empfangen von Dokumenten zwischen WebSphere Partner Gateway und WebSphere InterChange Server unter Verwendung des JMS-Transportprotokolls:

- „Erforderliche Komponenten für den Austausch von Dokumenten über den JMS-Transport“
- „Umgebung für den JMS-Transport einrichten“ auf Seite 179
- „Geschäftsobjektdefinitionen für JMS erstellen“ auf Seite 182

---

### Erforderliche Komponenten für den Austausch von Dokumenten über den JMS-Transport

Die Kommunikation von WebSphere Partner Gateway mit InterChange Server über das JMS-Transportprotokoll setzt voraus, dass die Komponenten zur Arbeit mit JMS konfiguriert werden. In Tabelle 62 sind diese Konfigurationsschritte zusammengefasst.

*Tabelle 62. WebSphere Partner Gateway und InterChange Server für das JMS-Transportprotokoll konfigurieren*

Komponente	Version	Weitere Informationen
WebSphere Partner Gateway	6.1	„Dokumentenversand über das JMS-Transportprotokoll an ICS konfigurieren“ auf Seite 126  „Dokumentenempfang von ICS über das JMS-Transportprotokoll konfigurieren“ auf Seite 127
WebSphere InterChange Server	4.3	„ICS-Artefakte für JMS erstellen“ auf Seite 186

Zum Senden oder Empfangen eines Dokuments zwischen WebSphere Partner Gateway und InterChange Server über das JMS-Transportprotokoll verwenden Sie außerdem die Komponenten, die in Tabelle 63 aufgeführt sind.

Tabelle 63. Erforderliche Komponenten zur Übertragung von Dokumenten von und an InterChange Server über JMS

Komponente	Beschreibung	Anmerkungen und Einschränkungen
WebSphere Business Integration Adapter für JMS (Adapter für JMS)	Dieser Adapter gibt InterChange Server die Möglichkeit, Geschäftsobjekte mit Anwendungen auszutauschen, die Daten in Form von JMS-Nachrichten senden oder empfangen. Der Adapter für JMS und WebSphere Partner Gateway kommunizieren über JMS-Warteschlangen.	Vergewissern Sie sich, dass der Adapter für JMS Version 2.7.0 (oder höher) verwendet wird. Dieser Adapter bietet Unterstützung für angepasste Headereigenschaften. Prüfen Sie anhand der Adapterdokumentation, ob die Version des Adapters mit der von Ihnen verwendeten Version von InterChange Server kompatibel ist.
Payload-Data-Handler	Dieser Data-Handler konvertiert die Nutzdaten (payload) zwischen dem Dokumentformat und der entsprechenden Geschäftsobjektdarstellung.	Weitere Informationen hierzu finden Sie im Abschnitt „Payload-Data-Handler angeben“ auf Seite 180.
Attachment-Data-Handler	Dieser Data-Handler konvertiert Dokumente mit Anhängen in Geschäftsobjekte.	Dieser Data-Handler ist <i>nur</i> für Dokumente mit Anhängen erforderlich. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumente mit Anhängen verarbeiten“ auf Seite 133.

In den folgenden Abschnitten wird beschrieben, wie die Komponenten in Tabelle 63 zusammenarbeiten, um Dokumente zwischen WebSphere Partner Gateway und InterChange Server über das JMS-Transportprotokoll zu senden und zu empfangen.

## Dokumentenversand über das JMS-Transportprotokoll

Beim Senden eines Dokuments von WebSphere Partner Gateway an InterChange Server über das JMS-Transportprotokoll verwenden Sie den Adapter für JMS, um die von WebSphere Partner Gateway in eine JMS-Warteschlange gestellte Nachricht abzurufen. Der Adapter leitet die Nachricht dann an InterChange Server weiter. In Abb. 35 ist zusammengefasst, wie WebSphere Partner Gateway über das JMS-Transportprotokoll Dokumente an InterChange Server sendet.

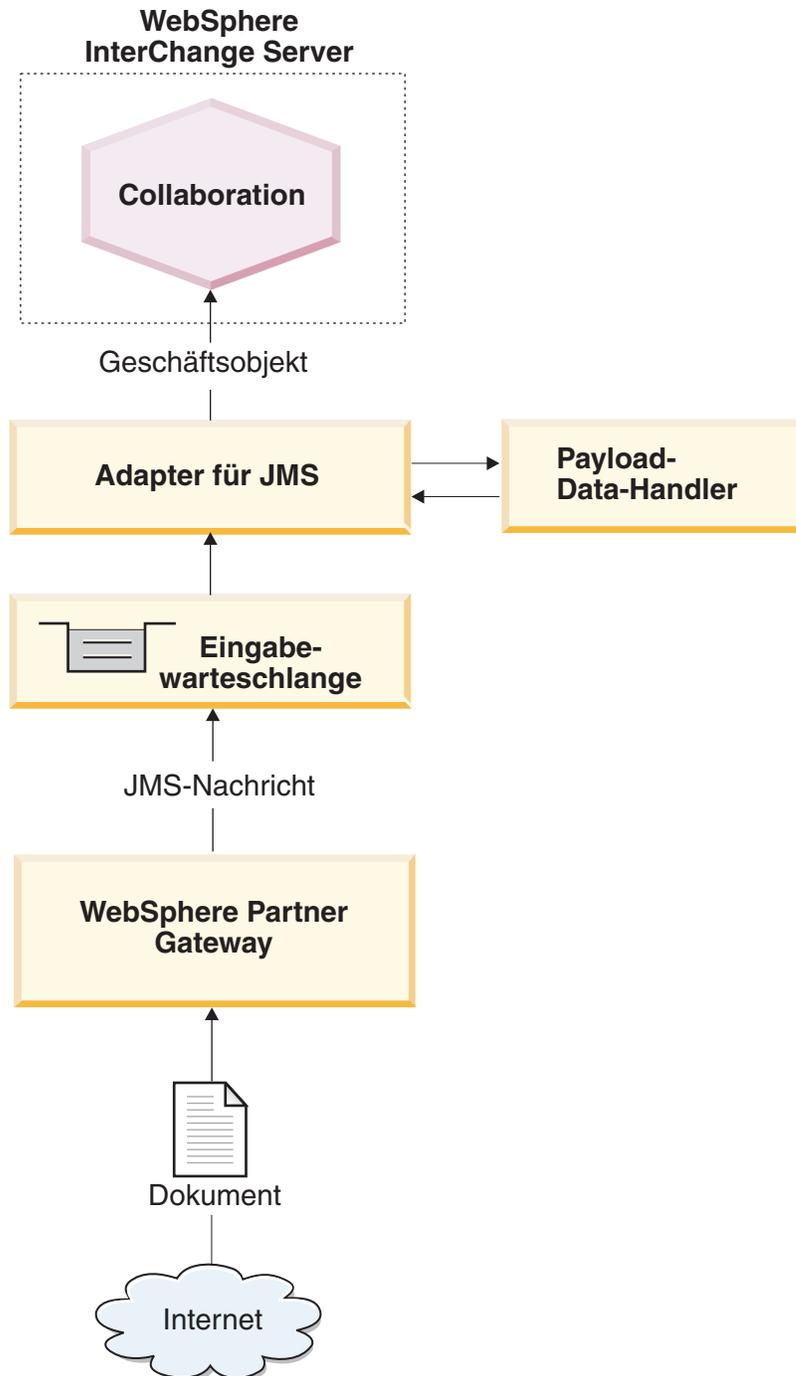


Abbildung 35. Nachrichtenfluss von WebSphere Partner Gateway an eine Collaboration über das JMS-Transportprotokoll

Die folgenden Schritte beschreiben, wie WebSphere Partner Gateway an einer Ereignisbenachrichtigung durch Senden eines Dokuments an eine Collaboration innerhalb von InterChange Server über das JMS-Transportprotokoll teilnimmt:

1. WebSphere Partner Gateway übergibt eine Nachricht an die JMS-Ausgangswarteschlange.

Wenn der Pakettyp des Dokuments 'Backend Integration' ist, wurden von WebSphere Partner Gateway angepasste Eigenschaften in dieser Nachricht

bereitgestellt. Der JMS-Nachrichtenheader 'JMSType' wird mit dem Inhaltstyp (content-type) der Nutzdaten (payload) definiert.

**Anmerkung:** Innerhalb von WebSphere Partner Gateway müssen Sie ein Ziel konfigurieren, das die JMS-Warteschlange angibt, an die WebSphere Partner Gateway die Nachricht sendet und die vom Adapter für JMS abgefragt wird.

2. Wenn der Adapter für JMS eine Nachricht in einer seiner Eingabewarteschlangen erkennt, ruft er die Nachricht ab.

Die JMS-Warteschlange, die von WebSphere Partner Gateway als Ausgangswarteschlange verwendet wird, ist dieselbe Warteschlange, die vom Adapter für JMS als Eingabewarteschlange verwendet wird. Informationen zur Konfiguration dieser Warteschlange finden Sie im Abschnitt „JMS-Warteschlangen konfigurieren“ auf Seite 179. Detaillierte Informationen zur Verarbeitung des Adapters für JMS finden Sie im Handbuch *Adapter for JMS User Guide*.

3. Der Adapter für JMS versetzt die Nachricht in die eigene Verarbeitungswarteschlange.
4. Der Adapter für JMS extrahiert den Hauptteil der JMS-Nachricht und ruft einen Data-Handler mit diesem Hauptteil der Nachricht auf. Der Data-Handler konvertiert den Hauptteil der JMS-Nachricht in ein Geschäftsobjekt.

**Anmerkung:** Wenn Ihre Nachrichten Anhänge haben, können Sie den Attachment-Data-Handler installieren und anschließend den Adapter für JMS so konfigurieren, dass er den Attachment-Data-Handler aufruft, um den Hauptteil der JMS-Nachricht in ein Geschäftsobjekt zu konvertieren. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumente mit Anhängen verarbeiten“ auf Seite 133.

Wenn als Pakettyp 'Backend Integration' verwendet wird und das Dokument Anhänge enthält, ist der konfigurierte Data-Handler für die Behandlung der Nutzdaten und der Anhänge zuständig.

5. Der Data-Handler gibt das Geschäftsobjekt an den Adapter für JMS zurück.

**Anmerkung:** Wenn der Attachment-Data-Handler verwendet wurde, enthält dieses Geschäftsobjekt die Nutzdaten und die Anhänge.

6. Wenn der Adapter für JMS ein untergeordnetes dynamisches Metaobjekt (angegeben durch `cw_mo_conn` in den anwendungsspezifischen Informationen auf Geschäftsobjektebene) antrifft, füllt der Adapter die im Geschäftsobjekt vorhandenen benutzerdefinierten JMS-Header mit den in der JMS-Nachricht vorhandenen Headern.
7. Der Adapter für JMS übermittelt das Geschäftsobjekt im Rahmen einer Subskriptionszustellung an InterChange Server.

## Dokumentenempfang über das JMS-Transportprotokoll

Beim Empfangen eines Dokuments von InterChange Server durch WebSphere Partner Gateway über das JMS-Transportprotokoll verwenden Sie den Adapter für JMS, der die Nachricht, die er von InterChange Server empfängt, in eine JMS-Warteschlange stellt, wo sie von WebSphere Partner Gateway abgerufen werden kann. In Abb. 36 ist zusammengefasst, wie WebSphere Partner Gateway über das JMS-Transportprotokoll Dokumente von InterChange Server empfängt.

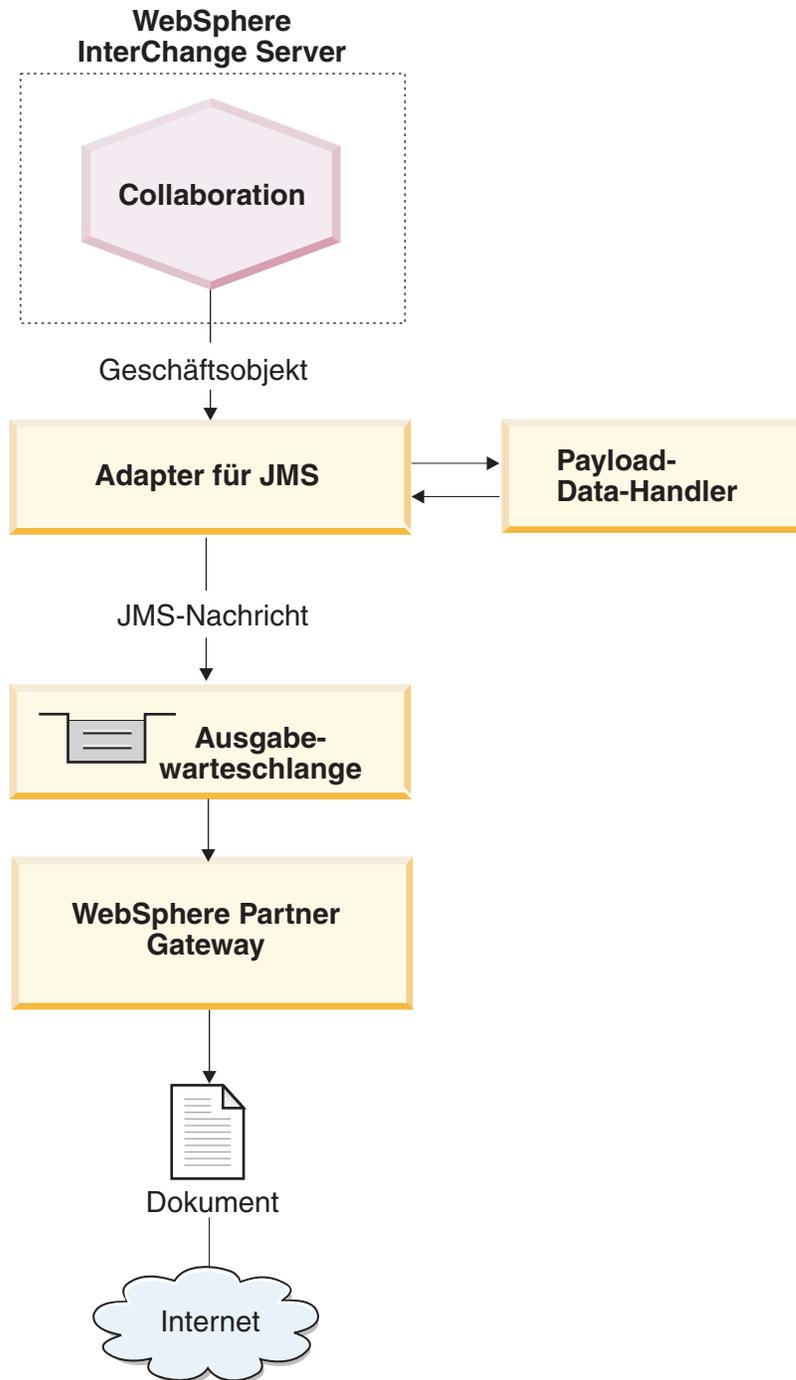


Abbildung 36. Nachrichtenfluss aus einer Collaboration an WebSphere Partner Gateway über das JMS-Transportprotokoll

Die folgenden Schritte beschreiben, wie WebSphere Partner Gateway an einer Anforderungsverarbeitung teilnimmt, indem es ein Dokument aus einer Collaboration innerhalb von InterChange Server über das JMS-Transportprotokoll empfängt.

1. Die Collaboration innerhalb von InterChange Server führt einen Serviceaufruf an den Adapter für JMS aus, und sendet diesem das Anforderungsgeschäftsobjekt.

Das Anforderungsgeschäftsobjekt enthält anwendungsspezifische Informationen, die auf ein dynamisches Metaobjekt verweisen, das wiederum die JMS-Headerinformationen der Transportebene enthält, die von WebSphere Partner Gateway erwartet werden.

2. Der Adapter für JMS verwendet einen Data-Handler, um das von der Collaboration gesendete Geschäftsobjekt in eine JMS-Nachricht zu konvertieren. Der Adapter liest die Eigenschaften `DataHandlerMimeType` und `DataHandlerConfigMO`, um den zu verwendenden Data-Handler zu ermitteln. Weitere Informationen hierzu finden Sie im Abschnitt „Payload-Data-Handler angeben“ auf Seite 180.

**Anmerkung:** Wenn Ihre Dokumente Anhänge haben, installieren Sie den Attachment-Data-Handler und konfigurieren anschließend den Adapter für JMS so, dass er den Attachment-Data-Handler aufruft, um das Anforderungsgeschäftsobjekt in ein Dokument mit Anhängen zu konvertieren. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumente mit Anhängen verarbeiten“ auf Seite 133.

3. Der Data-Handler konvertiert das Geschäftsobjekt in eine Zeichenfolge und gibt diese an den Adapter für JMS zurück.
4. Der Adapter für JMS ermittelt aus dem Anforderungsgeschäftsobjekt den Namen des dynamischen Metaobjekts für die angepassten JMS-Eigenschaften. Der Adapter durchsucht die anwendungsspezifischen Informationen des Anforderungsgeschäftsobjekts nach dem Tag `cw_mo_conn`, der das Attribut angibt, in dem das dynamische Metaobjekt enthalten ist. Wenn Sie mit dem Back-End-Integrationspaket für Ihr Dokument arbeiten, können Sie in diesem dynamischen Metaobjekt Headerinformationen der Transportebene angeben.
5. Der Adapter für JMS durchsucht das dynamische Metaobjekt nach dem Attribut `JMSProperties`. Wenn dieses Attribut Daten enthält, definiert der Adapter die Headerfelder der Transportebene im Anforderungsdokument. Innerhalb des Attributs `JMSProperties` können Sie außerdem den Standard-JMS-Header des Inhaltstyps (Content-Type) angeben. Weitere Informationen hierzu finden Sie im Abschnitt „JMS-Headerinformationen erstellen“ auf Seite 183.
6. Der Adapter für JMS erstellt eine JMS-Nachricht unter Verwendung der vom Data-Handler zurückgegebenen Zeichenfolge. Darüber hinaus legt er alle angepassten Eigenschaften fest, wie sie im dynamischen Metaobjekt definiert sind.
7. Der Adapter für JMS sendet die resultierende Anforderungsnachricht an die Ausgabewarteschlange. Die Warteschlange kann im statischen oder im dynamischen Metaobjekt angegeben werden. Informationen zur Angabe von Warteschlangen finden Sie im Abschnitt „JMS-Warteschlangen angeben“ auf Seite 181. WebSphere Partner Gateway ist über diese JMS-Warteschlange empfangsbereit, die als zugehörige Eingangswarteschlange in der Empfängerdefinition konfiguriert ist.
8. WebSphere Partner Gateway empfängt die Nachricht aus der zugehörigen JMS-Eingangswarteschlange, die als Empfänger konfiguriert ist.

**Anmerkung:** WebSphere Partner unterstützt über JMS nur *asynchrone* Interaktionen mit InterChange Server. Daher ist es unter Umständen nicht sinnvoll, auf die Antwort zu warten. Die Antwort vom externen Partner oder von WebSphere Partner Gateway kann über eine andere Warteschlange erfolgen. Sie können den Adapter für JMS so konfigurieren, dass er diese Warteschlange abfragt. Die Antwort, die über die Warteschlange kommt, kann dann im Rahmen der Ereigniszustellung an InterChange Server geleitet werden.

## Umgebung für den JMS-Transport einrichten

Da das Senden und Empfangen von Dokumenten mit InterChange Server die Verwendung von JMS-Warteschlangen und des Adapters für JMS erfordert, müssen Sie die in Tabelle 64 beschriebenen Einrichtungs- und Konfigurationsschritte ausführen. Informationen zur Konfiguration von WebSphere Partner Gateway zur Verwendung mit InterChange Server über JMS finden Sie im Abschnitt „WebSphere Partner Gateway für die Verwendung mit InterChange Server konfigurieren“ auf Seite 125.

Tabelle 64. Einrichten der Umgebung für die Verwendung des JMS-Transportprotokolls

Konfigurationsschritt	Weitere Informationen
1. Konfigurieren Sie Ihre JMS-Warteschlangen.	„JMS-Warteschlangen konfigurieren“
2. Konfigurieren Sie den WebSphere Business Integration Adapter für JMS.	„Adapter für JMS konfigurieren“ auf Seite 180

**Anmerkung:** Wenn Ihre Dokumente Anhänge enthalten, müssen Sie außerdem den Attachment-Data-Handler installieren und konfigurieren. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumente mit Anhängen verarbeiten“ auf Seite 133.

## JMS-Warteschlangen konfigurieren

Zur Verwendung des JMS-Transportprotokolls mit InterChange Server müssen Sie das von WebSphere MQ zur Verfügung gestellte JMS-System einrichten. Die unterstützten Versionen von InterChange Server verwenden WebSphere MQ Version 6.0 mit dem neuesten Fixpack als JMS-Provider. Zum Konfigurieren des Mechanismus für das JMS-Transportprotokoll können die Anweisungen im Handbuch *WebSphere Partner Gateway Hubkonfiguration* verwendet werden.

**Wichtig:** Die Schritte im Handbuch *WebSphere Partner Gateway Hubkonfiguration* müssen auf dem Computer ausgeführt werden, auf dem sich WebSphere Partner Gateway befindet. Dieses Handbuch setzt voraus, dass der JMS-Transportmechanismus, der für den Adapter für JMS und InterChange Server erforderlich ist, bereits im Rahmen der Installation von InterChange Server eingerichtet wurde.

Wenn Sie Ihre JMS-Warteschlangen für die Verwendung zwischen WebSphere Partner Gateway und InterChange Server erstellen, beachten Sie folgende Punkte:

- Ein Teil des Installationsprozesses von InterChange Server erfordert die Erstellung eines WebSphere MQ-Warteschlangenmanagers. Diesen Warteschlangenmanager können Sie mit WebSphere Partner Gateway verwenden.
- Wenn Sie die Aliasnamen für Ihre JMS-Warteschlangen erstellen, ist es sinnvoll, sie so zu benennen, dass sie einen Hinweis auf die Flussrichtung zwischen WebSphere Partner Gateway und InterChange Server geben. Wenn Sie zum Beispiel die in der Spalte *Originalwarteschlangenname* von Tabelle 65 aufgeführten Warteschlangen erstellen, könnten Sie diese Warteschlangen so umbenennen, dass sie die Richtung in Bezug auf InterChange Server wie in der Spalte *Richtungsbezogener Warteschlangenname* von Tabelle 65 gezeigt angeben.

Tabelle 65. Benennen von JMS-Warteschlangen mit Richtungsangabe für InterChange Server

Originalwarteschlangenname	Richtungsbezogener Warteschlangenname
inQ	ICS2HUB

Tabelle 65. Benennen von JMS-Warteschlangen mit Richtungsangabe für InterChange Server (Forts.)

Originalwarteschlangenname	Richtungsbezogener Warteschlangenname
outQ	HUB2ICS

## Adapter für JMS konfigurieren

Der Adapter für JMS ermöglicht WebSphere Partner Gateway den Austausch von Dokumenten mit InterChange Server in Form von JMS-Nachrichten. Er unterstützt die folgenden Interaktionen mit InterChange Server:

- Für die Anforderungsverarbeitung empfängt er das Anforderungsgeschäftsobjekt von InterChange Server, konvertiert es in eine JMS-Nachricht (mit Hilfe eines Data-Handlers) und stellt die JMS-Nachricht in eine JMS-Warteschlange (siehe Abb. 36 auf Seite 177), in der sie von WebSphere Partner Gateway aufgenommen werden kann.
- Für die Ereignisbenachrichtigung fragt er eine JMS-Warteschlange nach JMS-Nachrichten von WebSphere Partner Gateway ab. Wenn er eine JMS-Nachricht findet, konvertiert er sie in ein Ereignisgeschäftsobjekt (mit Hilfe eines Data-Handlers) und sendet sie an InterChange Server.

**Wichtig:** WebSphere Partner Gateway enthält den WebSphere Business Integration Adapter für JMS *nicht*. Sie müssen dieses Produkt getrennt erwerben und entsprechend den Anweisungen im zugehörigen Handbuch *Adapter for JMS User Guide* installieren. Es ist wichtig, die in diesem Handbuch beschriebenen Schritte zu lesen, um den Adapter für JMS korrekt zu installieren und zu konfigurieren.

Wenn Sie den Adapter für JMS zur Kommunikation mit InterChange Server konfiguriert haben, führen Sie die im folgenden Abschnitt beschriebenen Schritte aus, um diesen Adapter zum Empfang von JMS-Nachrichten aus WebSphere Partner Gateway zu konfigurieren:

- „Payload-Data-Handler angeben“
- „JMS-Warteschlangen angeben“ auf Seite 181

### Payload-Data-Handler angeben

Wie Abb. 36 auf Seite 177 zeigt, verwendet der Adapter für JMS einen Data-Handler, um die von InterChange Server empfangenen Geschäftsobjekte in die entsprechenden JMS-Nachrichten zu konvertieren.

**Anmerkung:** Der Data-Handler, der vom Adapter für JMS aufgerufen wird, konvertiert die Nutzdaten des Dokuments. Wenn Ihr Dokument in einen XML-Transportumschlag eingefügt ist (d. h. wenn es Anhänge enthält oder die Umschlagsmarkierung auf 'Ja' gesetzt ist), konfigurieren Sie den Attachment-Data-Handler als Payload-Data-Handler. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumente mit Anhängen verarbeiten“ auf Seite 133.

Um anzugeben, welcher Data-Handler zur Konvertierung der Nutzdaten zu verwenden ist, müssen Sie die im Abschnitt „Geschäftsobjekte konvertieren“ auf Seite 130 aufgeführten Schritte ausführen. Darüber hinaus müssen Sie den Adapter für JMS zur Verwendung dieses Payload-Data-Handlers konfigurieren. Führen Sie in Connector Configurator die folgenden Schritte aus:

1. Definieren Sie die folgenden Connectorkonfigurationseigenschaften, um den Data-Handler anzugeben:

- Definieren Sie die Eigenschaften `DataHandlerConfigMO` und `DataHandlerMimeType` mit dem Namen des Data-Handler-Metaobjekts der höchsten Ebene bzw. dem unterstützten MIME-Typ.
- Definieren Sie die Eigenschaft `DataHandlerClassName` mit dem Namen der zu instanziiierenden `DataHandler`-Klasse.

**Anmerkung:** Sie definieren *entweder* die Eigenschaften `DataHandlerConfigMO` und `DataHandlerMimeType` *oder* die Eigenschaft `DataHandlerClassName`.

2. Fügen Sie das `DataHandler`-Metaobjekt der höchsten Ebene in die Liste der unterstützten Geschäftsobjekte ein.

Sie können den zu verwendenden Data-Handler auch im statischen oder dynamischen Metaobjekt angeben. In diesen Metaobjekten stehen die gleichen Eigenschaften (`DataHandlerMimeType`, `DataHandlerConfigMO` und `DataHandlerClassName`) als Attribute zur Verfügung. Eine vollständige Beschreibung finden Sie im Handbuch *Adapter for JMS User Guide*.

### JMS-Warteschlangen angeben

Wenn der Adapter für JMS ein Dokument von InterChange Server empfängt, stellt er die Nachricht in seine Ausgangswarteschlange, die wiederum eine der Warteschlangen ist, die von der Empfangskomponente (Receiver) von WebSphere Partner Gateway abgefragt werden. Wenn WebSphere Partner Gateway ein Dokument an InterChange Server sendet, wird das Dokument analog in die Ausgangswarteschlange gestellt, die vom Adapter für JMS abgefragt wird.

In Tabelle 66 ist die Konfiguration der JMS-Warteschlangen zusammengefasst, die vom Adapter für JMS zum Empfangen und Senden von Dokumenten verwendet werden.

**Anmerkung:** Eine vollständige Beschreibung der Konfiguration von JMS-Warteschlangen finden Sie im Handbuch *Adapter for JMS User Guide*.

Tabelle 66. JMS-Warteschlangen

JMS-Warteschlange	Konfiguration
Eingabewarteschlange	<p>Setzen Sie die Connectorkonfigurationseigenschaft <code>InputDestination</code> auf den Namen der JMS-Warteschlange, die vom Adapter für JMS auf eingehende Nachrichten abgefragt wird.</p> <p>Stellen Sie sicher, dass der Name dieser Warteschlange der gleiche Name ist, den WebSphere Partner Gateway für die JMS-Ausgangswarteschlange verwendet. Wenn diese Warteschlange nicht in <code>InputDestination</code> angegeben wird, fragt der Adapter für JMS die Warteschlange <i>nicht</i> ab.</p> <p><b>Anmerkung:</b> Die Eigenschaft <code>InputDestination</code> enthält eine durch Kommata getrennte Liste von Eingabewarteschlangen. Wenn der Adapter für JMS mehrere Warteschlangen abfragt, stellen Sie sicher, dass diese Liste den Namen der JMS-Warteschlange enthält, die von WebSphere Partner Gateway als JMS-Ausgangswarteschlange verwendet wird.</p>
Ausgabewarteschlange	<p>Die Collaboration kann während der Ausführung das Attribut <code>OutputQueue</code> im dynamischen Metaobjekt dynamisch auf den Namen der JMS-Warteschlange setzen, an die der Adapter für JMS seine ausgehende Nachricht sendet.</p>

Sie müssen sicherstellen, dass die statischen oder dynamischen Metaobjekte so konfiguriert sind, dass sie in die Warteschlange schreiben können, über die der WebSphere Partner Gateway-Empfänger empfangsbereit ist.

## Geschäftsobjektdefinitionen für JMS erstellen

Der Adapter für JMS sendet Ihr Dokument an InterChange Server (und empfängt es von dort) in Form eines Geschäftsobjekts für Nutzdaten. Der Adapter für JMS ruft den Payload-Data-Handler auf, um dieses Geschäftsobjekt wie folgt zu verarbeiten, wenn er ein WebSphere Partner Gateway-Dokument empfängt bzw. sendet:

- Bei der Anforderungsverarbeitung konvertiert der Payload-Data-Handler das Anforderungsgeschäftsobjekt in die entsprechende JMS-Nachricht.
- Bei der Ereignisbenachrichtigung konvertiert der Data-Handler die JMS-Nachricht in ein Ereignisgeschäftsobjekt.

Daher müssen Sie die in Tabelle 67 gezeigten Geschäftsobjektdefinitionen erstellen, um die Struktur des Geschäftsobjekts für Nutzdaten darzustellen, die der Adapter für JMS erwartet.

Tabelle 67. Geschäftsobjektdefinitionen für den Adapter für JMS

Bedingung	Geschäftsobjektdefinition	Weitere Informationen
Wenn Sie für Ihre Nachricht den Pakettyp 'None' oder 'Backend Integration' verwenden <i>und</i> Ihre Dokumente <i>keine</i> Anhänge haben	Geschäftsobjekt für Nutzdaten.	„Struktur des Geschäftsobjekts für Nutzdaten für JMS erstellen“.
Wenn Sie für Ihr Dokument den Pakettyp 'Backend Integration' verwenden	Geschäftsobjekt zur Aufnahme der Nachrichtenheaderdaten: <ul style="list-style-type: none"> <li>• Dynamisches Metaobjekt</li> <li>• Geschäftsobjekt für JMS-Eigenschaften</li> </ul>	„JMS-Headerinformationen erstellen“ auf Seite 183.
Wenn das Dokument Anhänge enthält	Sie müssen außerdem zusätzliche Geschäftsobjekte zur Darstellung der Anhänge erstellen.	„Anhangsbezogene Geschäftsobjektdefinitionen erstellen“ auf Seite 145

## Struktur des Geschäftsobjekts für Nutzdaten für JMS erstellen

Die Struktur des Geschäftsobjekts für Nutzdaten für das JMS-Transportprotokoll hängt wie folgt vom Pakettyp ab:

- Wenn Ihr Dokument mit dem Pakettyp 'None' arbeitet, gibt es keine besonderen Bestimmungen zur Erstellung des Geschäftsobjekts für Nutzdaten für ein Dokument, das über das JMS-Transportprotokoll gesendet wird.

Wie im Abschnitt „Geschäftsobjekt für das Dokument“ auf Seite 129 erläutert, müssen Sie ein Attribut für jede zu übertragende Einzelinformation der Nutzdaten erstellen.

- Wenn Ihr Dokument mit 'Backend Integration' arbeitet, müssen Sie die folgenden Schritte ausführen:

1. Fügen Sie dem Geschäftsobjekt für Nutzdaten ein spezielles Attribut zur Angabe des dynamischen Metaobjekts hinzu. Dieses Attribut liefert Konfigurationsdaten für die Header der Transportebene der Nachricht.

- Fügen Sie den anwendungsspezifischen Informationen auf Geschäftsobjekt-ebene den Tag `cw_mo_conn` hinzu, um das Attribut anzugeben, in dem das dynamische Metaobjekt enthalten ist.

Weitere Informationen zu diesen Schritten finden Sie im Abschnitt „JMS-Headerinformationen erstellen“.

**Anmerkung:** Bei der Anforderungsverarbeitung kann das JMS-Transportprotokoll *nur* asynchrone Interaktionen unterstützen. Sie können ein Anforderungsgeschäftsobjekt senden, jedoch *keine* Antwort empfangen. Daher müssen Sie eine Geschäftsobjektdefinition für die Anforderung, jedoch nicht für eine Antwort erstellen.

## JMS-Headerinformationen erstellen

Wenn Sie Dokumente senden oder empfangen, die das Back-End-Integrationspaket über das JMS-Transportprotokoll verwenden, muss Ihr Anforderungsgeschäftsobjekt angepasste Headerinformationen der Transportebene enthalten. Der Adapter für JMS erwartet, dass sich diese angepassten Headerinformationen in seinem *dynamischen Metaobjekt* befinden.

Abb. 37 zeigt die Geschäftsobjektstruktur, die der Adapter für JMS für ein Anforderungsgeschäftsobjekt verwendet, das ein WebSphere Partner Gateway-Dokument mit 'Backend Integration' darstellt.

**Anmerkung:** Das Handbuch *Adapter for JMS User Guide* enthält Informationen zu dieser Anforderungsgeschäftsobjektstruktur. Lesen Sie die Informationen in diesem Handbuch, wenn Sie Ihre Geschäftsobjektdefinitionen erstellen.

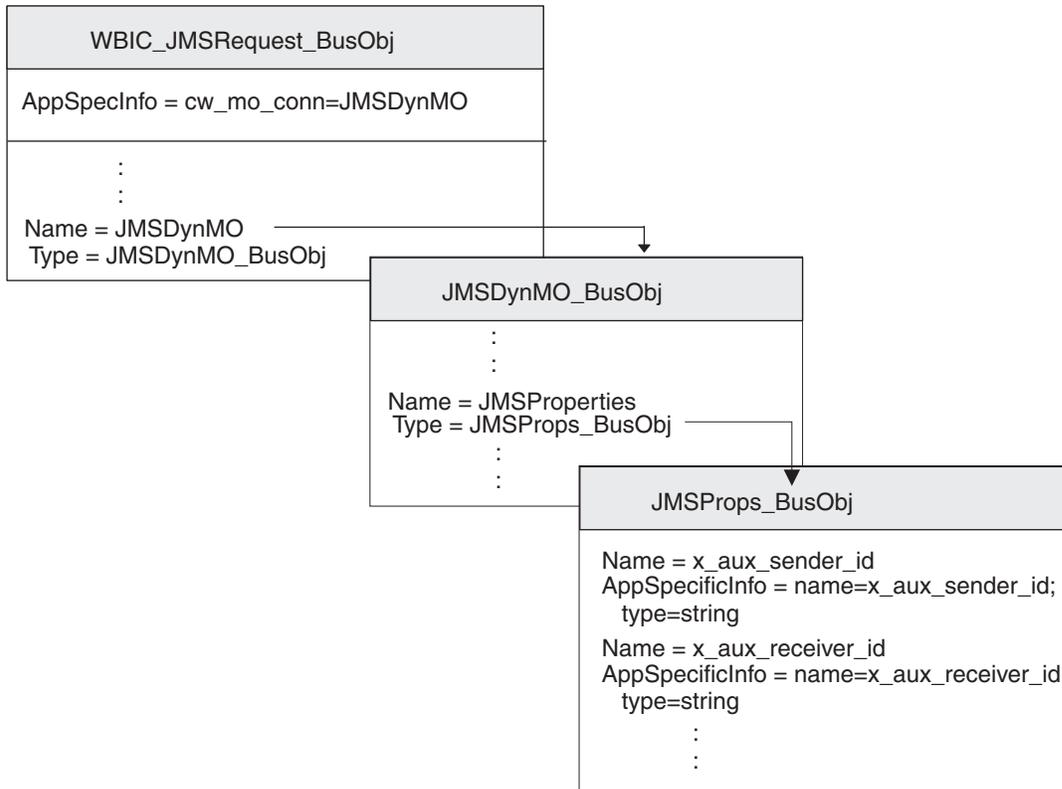


Abbildung 37. Beziehung zwischen dem Anforderungsgeschäftsobjekt und dem dynamischen JMS-Metaobjekt

Stellen Sie sicher, dass Ihre Geschäftsobjektstruktur ein dynamisches untergeordnetes Metaobjekt enthält, indem Sie die folgenden Schritte ausführen:

1. Erstellen Sie eine Geschäftsobjektdefinition, die die JMS-Eigenschaften enthält, die für das Back-End-Integrationspaket erforderlich sind.
2. Erstellen Sie eine Geschäftsobjektdefinition für das dynamische Metaobjekt.
3. Ändern Sie die Geschäftsobjektdefinition für Ihr Anforderungsgeschäftsobjekt, so dass sie ein Attribut für das dynamische Metaobjekt enthält.

Jeder dieser Schritte wird in den folgenden Abschnitten beschrieben.

### Geschäftsobjekt für JMS-Eigenschaften erstellen

Ein *Geschäftsobjekt für JMS-Eigenschaften* enthält JMS-Eigenschaften, die für die vom Back-End-Integrationspaket benötigten Header der Transportebene erforderlich sind. Es kann außerdem das Attribut `content-type`, das den Content-Type-Header angibt, der in der Anforderungsnachricht einzufügen ist, und das Attribut `content-length` enthalten, das die Länge der Nachricht in Byte angibt. Die einzelnen gültigen Transportheaderfelder werden in Tabelle 8 auf Seite 26 beschrieben.

Zur Erstellung einer Geschäftsobjektdefinition für JMS-Eigenschaften führen Sie die folgenden Schritte aus:

1. Erstellen Sie innerhalb der Geschäftsobjektdefinition ein Attribut für jedes einzelne Headerfeld der Transportebene.  
Alle Attribute müssen den Attributtyp 'Zeichenfolge' haben. Bei JMS-Nachrichten werden in den Namen der Transportheaderfelder an Stelle von Silbentrennungsstrichen Unterstreichungszeichen verwendet (siehe hierzu Tabelle 68).
2. Fügen Sie für jedes Attribut in der Geschäftsobjektdefinition für JMS-Eigenschaften anwendungsspezifische Informationen hinzu, um den Zweck des zugeordneten Attributs anzugeben.  
Diese anwendungsspezifischen Informationen auf Attributebene müssen folgendes Format haben:  
`name=JMSeigenschaft;type=string`  
Dabei ist *JMSeigenschaft* einer der Werte in der Spalte JMS-Eigenschaftsname in Tabelle 68.
3. Für jedes der Attribute in der Geschäftsobjektdefinition für JMS-Eigenschaften können Sie einen Standardwert hinzufügen, um den allgemeinen (bzw. einzig gültigen) Wert für das entsprechende Feld der Transportebene anzugeben.

Tabelle 68. Attribute für die Geschäftsobjektdefinition für JMS-Eigenschaften

Transportheaderfeld	JMS-Eigenschaftsname
x-aux-sender-id	x_aux_sender_id
x-aux-receiver-id	x_aux_receiver_id
x-aux-protocol	x_aux_protocol
x-aux-protocol-version	x_aux_protocol_version
x-aux-process-type	x_aux_process_type
x-aux-process-version	x_aux_process_version
x-aux-create-datetime	x_aux_create_datetime
x-aux-msg-id	x_aux_msg_id
x-aux-production	x_aux_production
x-aux-system-msg-id	x_aux_system_msg_id
x-aux-payload-root-tag	x_aux_payload_root_tag

Tabelle 68. Attribute für die Geschäftsobjektdefinition für JMS-Eigenschaften (Forts.)

Transportheaderfeld	JMS-Eigenschaftsname
x-aux-process-instance-id	x_aux_process_instance_id
x-aux-event-status-code	x_aux_event_status_code
x-aux-third-party-bus-id	x_aux_third_party_bus_id
x-aux-transport-retry-count	x_aux_transport_retry_count
content-type	content_type
content-length	content_length

**Anmerkung:** Tabelle 68 enthält *keine* vollständige Liste der Header, die für die Back-End-Integration erforderlich sind. Eine vollständige Liste sowie eine Beschreibung der Header finden Sie im Abschnitt „Inhalt des Headers der Transportebene“ auf Seite 26. Stellen Sie sicher, dass Sie in den Feldnamen des Transportheaders Silbentrennungsstriche durch Unterstreichungszeichen ersetzen.

In Abb. 37 auf Seite 183 enthält die Geschäftsobjektdefinition JMSProps\_BusObj Attribute für die verschiedenen Headerfelder der Transportebene. Diese Attribute haben alle anwendungsspezifische Informationen auf Attributebene zur Angabe des Namens des zugehörigen Protokollheaders. Zum Beispiel sind für das Attribut `x_aux_sender_id` die anwendungsspezifischen Informationen wie folgt definiert:

```
name=x_aux_sender_id;type=string
```

### Dynamisches JMS-Metaobjekt erstellen

Dieses dynamische Metaobjekt enthält ein untergeordnetes Geschäftsobjekt mit Konfigurationsdaten für den Adapter für JMS. Zur Erstellung einer Geschäftsobjektdefinition für ein dynamisches Metaobjekt führen Sie die folgenden Schritte aus:

1. Erstellen Sie ein Attribut mit dem Namen `JMSProperties`, dessen Attributtyp die Geschäftsobjektdefinition für das Geschäftsobjekt für JMS-Eigenschaften ist (siehe „Geschäftsobjekt für JMS-Eigenschaften erstellen“ auf Seite 184).
2. Fügen Sie nach Bedarf weitere Konfigurationseigenschaften hinzu. Eine Liste der gültigen Attribute für das dynamische Metaobjekt finden Sie im Handbuch *Adapter for JMS User Guide*. Lesen Sie in diesem Handbuch die Informationen zur Erstellung von Attributen, um das dynamische Metaobjekt zu konfigurieren.

Damit der Adapter für JMS mit WebSphere Partner Gateway arbeiten kann, *muss* die Geschäftsobjektdefinition für das dynamische Metaobjekt das Attribut `JMSProperties` enthalten, dessen Attributtyp die Geschäftsobjektdefinition des Geschäftsobjekts für JMS-Eigenschaften ist (siehe „Geschäftsobjekt für JMS-Eigenschaften erstellen“ auf Seite 184). Zum Beispiel enthält die Geschäftsobjektdefinition `JMSDynMO_BusObj` in Abb. 37 auf Seite 183 Attribute für die verschiedenen Konfigurationseigenschaften (nicht gezeigt) sowie das Attribut `JMSProperties`.

### Anforderungsgeschäftsobjektdefinition ändern

Zum Einfügen der Geschäftsobjektstruktur in Ihr Anforderungsgeschäftsobjekt müssen Sie die folgenden Änderungen an Ihrer Anforderungsgeschäftsobjektdefinition vornehmen:

1. Fügen Sie Ihrer Anforderungsgeschäftsobjektdefinition ein Attribut hinzu, in dem das untergeordnete dynamische Metaobjekt enthalten ist.  
Der Attributtyp für dieses Attribut ist die Geschäftsobjektdefinition für das dynamische Metaobjekt (siehe „Dynamisches JMS-Metaobjekt erstellen“).

2. Fügen Sie den anwendungsspezifischen Informationen auf Geschäftsobjekt-ebene Ihrer Anforderungsgeschäftsobjektdefinition den Tag `cw_mo_conn` hinzu, um das Attribut anzugeben, in dem das dynamische Metaobjekt enthalten ist.

Der Tag `cw_mo_conn` hat folgendes Format:

```
cw_mo_conn=dynamischesMetaObjAttr
```

Dabei ist `dynamischesMetaObjAttr` der Name des Attributs in dem Anforderungsgeschäftsobjekt, in dem das dynamische Metaobjekt enthalten ist.

In Abb. 37 auf Seite 183 wurde der Anforderungsgeschäftsobjektdefinition `HUB_JMSRequest_BusObj` zum Beispiel ein Attribut mit dem Namen `JMSDynM0` hinzugefügt. Dieses Attribut enthält das dynamische Metaobjekt, das ein untergeordnetes Geschäftsobjekt des Typs `JMSDynM0_BusObj` ist. Darüber hinaus wurden die anwendungsspezifischen Informationen des Anforderungsgeschäftsobjekts modifiziert, um den folgenden Tag `cw_mo_conn` zur Angabe dieses dynamischen Metaobjekts einzufügen:

```
cw_mo_conn=JMSDynM0
```

## ICS-Artefakte für JMS erstellen

Zur Konfiguration von InterChange Server zur Kommunikation mit WebSphere Partner Gateway über das JMS-Transportprotokoll müssen Sie die in Tabelle 69 aufgeführten InterChange Server-Artefakte erstellen.

Tabelle 69. ICS-Artefakte zur Kommunikation über das JMS-Transportprotokoll

ICS-Artefakt	Zweck	Weitere Informationen
Geschäftsobjektdefinitionen	Stellen das Dokument dar.	„Geschäftsobjektdefinitionen für JMS erstellen“ auf Seite 182
Connectorobjekt	Stellt den Adapter für JMS während der Ausführung dar.	„JMS-Connectorobjekt erstellen“
Collaboration-Schablone und Collaboration-Objekt	Stellen den Geschäftsprozess dar, der von InterChange Server zur Verarbeitung des Dokuments verwendet wird.	„Collaborations zur Kommunikation mit dem Adapter für JMS binden“ auf Seite 187

## JMS-Connectorobjekt erstellen

Zum Aufruf einer Instanz des Adapters für JMS während der Ausführung führen Sie im Tool 'System Manager' die folgenden Schritte aus:

1. Erstellen Sie die Connectorobjekte:
  - Erstellen Sie ein Connectorobjekt, um eine Instanz des Adapters für JMS darzustellen.

**Anmerkung:** Stellen Sie auf der Registerkarte **Unterstützte Geschäftsobjekte** von Connector Configurator sicher, dass Sie alle Geschäftsobjektdefinitionen angeben, die Sie zur Verwendung mit dem Adapter für JMS erstellt haben. Eine Beschreibung dieser Geschäftsobjektdefinitionen finden Sie im Abschnitt „Geschäftsobjektdefinitionen für JMS erstellen“ auf Seite 182.

- Wenn dies für Ihre Collaboration erforderlich ist, erstellen Sie ein Connectorobjekt für den Port-Connector.
2. Konfigurieren Sie die Connectorobjekte.

Informationen zur Konfiguration Ihres Adapters für JMS zur Verwendung mit WebSphere Partner Gateway finden Sie im Abschnitt „Adapter für JMS konfigurieren“ auf Seite 180.

## **Collaborations zur Kommunikation mit dem Adapter für JMS binden**

Wie im Abschnitt „Collaborations erstellen“ auf Seite 132 beschrieben, muss ein Collaboration-Objekt während der Ausführung vorhanden sein, damit InterChange Server ermitteln kann, wo Geschäftsobjekte zu empfangen sind und wohin Geschäftsobjekte zu senden sind. Bei der Erstellung des Collaboration-Objekts für die Collaboration, die Informationen mit Hilfe des Adapters für JMS an WebSphere Partner Gateway sendet und von WebSphere Partner Gateway empfängt, binden Sie die Collaboration-Ports wie folgt:

- Für die Anforderungsverarbeitung stellen Sie den Empfangsport (“To”), der Anforderungen an WebSphere Partner Gateway sendet, auf das Connectorobjekt ein, das Sie für den Adapter für JMS erstellt haben. Das heißt, der Adapter für JMS ist der *Zieladapter*.
- Für die Ereignisbenachrichtigung stellen Sie den Absenderport (“from”), der Ereignisse von WebSphere Partner Gateway empfängt, auf das Connectorobjekt ein, das Sie für den Adapter für JMS erstellt haben. Das heißt, der Adapter für JMS ist der *Quellenadapter*.



---

## **Teil 4. Integration mit anderen Back-End-Systemen**

In den folgenden Kapiteln wird die Vorgehensweise für die Integration von WebSphere Partner Gateway mit Back-End-Systemen, wie beispielsweise WebSphere Message Broker und WebSphere Data Interchange, beschrieben.



---

## Kapitel 11. Integration mit WebSphere Message Broker

In diesem Kapitel wird die Integration von WebSphere Partner Gateway mit WebSphere Message Broker beschrieben.

### Anmerkungen:

1. Eine Beschreibung des allgemeinen Verfahrens zur Integration von WebSphere Partner Gateway mit einem Back-End-System finden Sie in Kapitel 2, „Back-End-Integration planen“, auf Seite 9.
2. Für die Informationen dieses Kapitels wird vorausgesetzt, dass Sie mit WebSphere Message Broker und den zugehörigen Komponenten wie Projekten und Nachrichtenflüssen vertraut sind.

Die Integration von WebSphere Partner Gateway mit einem Back-End-System wird häufig von zwei verschiedenen Personen ausgeführt oder in zwei verschiedene Aufgabenbereiche (Rollen) untergliedert. Jede Rolle ist hierbei für die Konfiguration einer bestimmten Komponente zuständig, für die der entsprechende Mitarbeiter Experte ist. Aus diesem Grund wird in diesem Kapitel die Integration mit WebSphere Message Broker in die Konfiguration von WebSphere Partner Gateway und die Konfiguration von Message Broker unterteilt. In Tabelle 70 sind diese Konfigurationsrollen sowie die Abschnitte innerhalb des Kapitels aufgeführt, in denen Sie die zugehörigen Konfigurationsinformationen finden.

*Tabelle 70. Rollen bei der WebSphere Message Broker-Integration*

Konfigurationsrolle	Weitere Informationen
Konfiguration von WebSphere Partner Gateway	<ol style="list-style-type: none"><li>1. „Integration mit WebSphere Message Broker planen“ auf Seite 192</li><li>2. „WebSphere Partner Gateway für die Verwendung mit WebSphere Message Broker konfigurieren“ auf Seite 193</li></ol>
Konfiguration von WebSphere Message Broker	<ol style="list-style-type: none"><li>1. „Integration mit WebSphere Message Broker planen“ auf Seite 192</li><li>2. „WebSphere Message Broker konfigurieren“ auf Seite 196</li></ol>

**Anmerkung:** Diese Konfigurationsrollen können getrennt voneinander ausgeführt werden. Allerdings werden für sie auch allgemeine Informationen benötigt, um den Datenaustausch zwischen den beiden Komponenten zu ermöglichen.

Dieses Kapitel enthält die folgenden Informationen:

- „Integration mit WebSphere Message Broker planen“ auf Seite 192
- „WebSphere Partner Gateway für die Verwendung mit WebSphere Message Broker konfigurieren“ auf Seite 193
- „WebSphere Message Broker konfigurieren“ auf Seite 196
- „HTTP-Transportprotokoll mit WebSphere Message Broker verwenden“ auf Seite 197
- „SOAP-Dokumente senden“ auf Seite 201
- „JMS-Transportprotokoll mit WebSphere Message Broker verwenden“ auf Seite 203

## Integration mit WebSphere Message Broker planen

Zur Planung der Integration mit WebSphere Message Broker befolgen Sie die in Kapitel 2, „Back-End-Integration planen“, auf Seite 9 aufgeführten Schritte. In Tabelle 71 sind die Schritte zur Integration von WebSphere Partner Gateway mit WebSphere Message Broker zusammengefasst.

*Tabelle 71. Planung der WebSphere Message Broker-Integration*

Integrationschritt	Weitere Informationen
1. Vergewissern Sie sich, dass Sie eine unterstützte Version von WebSphere Message Broker installiert und für WebSphere Partner Gateway verfügbar haben.	Kapitel 11: „Von WebSphere Partner Gateway unterstützte WebSphere Message Broker-Versionen“
2. Bestimmen Sie das Geschäftsprotokoll des WebSphere Message Broker-Dokuments.	Kapitel 2: „Zu verwendendes Geschäftsprotokoll“ auf Seite 9
3. Bestimmen Sie den Pakettyp für das Dokument: 'None' oder 'Backend Integration'.	Kapitel 2: „Zu verwendender Pakettyp“ auf Seite 25
4. Bestimmen Sie den zwischen WebSphere Partner Gateway und WebSphere Message Broker zu verwendenden Nachrichtentransport.	Kapitel 11: „Von WebSphere Message Broker unterstützte Nachrichtentransporte“
5. Konfigurieren Sie WebSphere Partner Gateway.	Kapitel 11: „WebSphere Partner Gateway für die Verwendung mit WebSphere Message Broker konfigurieren“ auf Seite 193

### Von WebSphere Partner Gateway unterstützte WebSphere Message Broker-Versionen

WebSphere Partner Gateway unterstützt die Integration mit Version 6.0 (neuestes Fixpack) von Message Broker. Message Broker ist auf verschiedenen Plattformen verfügbar, zu denen Windows 2000 und verschiedene UNIX-basierte Plattformen zählen. Weitere Informationen finden Sie im Installationshandbuch für Message Broker in der Dokumentation zu WebSphere Message Broker.

### Von WebSphere Message Broker unterstützte Nachrichtentransporte

Die folgenden beiden Nachrichtentransportprotokolle werden von WebSphere Message Broker unterstützt:

- HTTP-Transportprotokoll (einschließlich Web-Services)
- JMS-Transportprotokoll

Die Unterstützung für diese Nachrichtentransportprotokolle setzt die Installation und Konfiguration von IBM WebSphere MQ voraus.

#### HTTP

Message Broker verwendet das HTTP-Transportprotokoll für Web-Services-Transaktionen. Zum Senden und Empfangen von Dokumenten zwischen WebSphere Partner Gateway und Message Broker über das HTTP-Protokoll benötigen Sie keine weitere Software. Allerdings ist zum Senden eines Dokuments aus Message Broker heraus an ein anderes Ziel WebSphere MQ erforderlich.

**Anmerkung:** WebSphere Partner Gateway unterstützt sowohl synchrone als auch asynchrone Interaktionen mit Message Broker über HTTP.

## JMS

Message Broker verwendet das JMS-Transportprotokoll für die meisten seiner Transaktionen. Zum Senden und Empfangen von Dokumenten zwischen WebSphere Partner Gateway und Message Broker über das JMS-Transportprotokoll müssen Sie JMS-Warteschlangen verwenden. Wenn sich diese beiden Komponenten auf verschiedenen Computern befinden, müssen Sie die JMS-Warteschlangen auf beiden Computern erstellen. Grundsätzlich basiert die Unterstützung für JMS auf der Verwendung eines Nachrichtenflusses innerhalb von Message Broker und den zugrunde liegenden JMS-Warteschlangen. Weitere Informationen zur Konfiguration für JMS finden Sie im Abschnitt „JMS-Transportprotokoll mit WebSphere Message Broker verwenden“ auf Seite 203.

**Anmerkung:** WebSphere Partner Gateway unterstützt nur asynchrone Interaktionen mit Message Broker über JMS.

## Unterstützung für WebSphere Message Broker-Integration

WebSphere Partner Gateway stellt Muster bereit, die Sie beim Integrationsprozess mit Message Broker unterstützen. Diese Muster befinden sich im folgenden Verzeichnis des Produktverzeichnisses von WebSphere Partner Gateway:

`Integration/WBI/WBIMB/samples`

---

## WebSphere Partner Gateway für die Verwendung mit WebSphere Message Broker konfigurieren

Eine allgemeine Übersicht zur Konfiguration von WebSphere Partner Gateway für die Kommunikation mit einem Back-End-System finden Sie im Abschnitt „WebSphere Partner Gateway konfigurieren“ auf Seite 47. In diesem Abschnitt werden die Schritte zusammengefasst, die zur Konfiguration von WebSphere Partner Gateway zur Kommunikation mit Message Broker erforderlich sind.

Die Konfiguration von WebSphere Partner Gateway erfolgt in folgenden Schritten:

- Unterstützung für ausgehende Dokumente konfigurieren  
Informationen zum Senden von Dokumenten von WebSphere Partner Gateway an Message Broker finden Sie im Abschnitt „Unterstützung für ausgehende Dokumente bereitstellen“.
- Unterstützung für eingehende Dokumente konfigurieren  
Informationen zum Senden von Dokumenten von Message Broker an WebSphere Partner Gateway finden Sie im Abschnitt „Unterstützung für eingehende Dokumente bereitstellen“ auf Seite 195.

## Unterstützung für ausgehende Dokumente bereitstellen

Damit WebSphere Partner Gateway Dokumente an beliebige Back-End-Systeme senden kann, müssen Sie die im Abschnitt „Zieleinheit für Partnerdokument definieren“ auf Seite 49 beschriebenen Schritte ausführen. Wenn Message Broker das Back-End-System ist, müssen Sie ein Ziel erstellen, dessen Transporttyp mit dem Transportprotokoll übereinstimmt, das für Nachrichten zwischen WebSphere Partner Gateway und Message Broker verwendet wird. Wenn der Hub ein Dokument an Message Broker sendet, muss er wissen, wohin das Dokument weiterzuleiten ist. Diese Position muss dem verwendeten Transportprotokoll entsprechen.

Das Transportprotokoll muss eines der von Message Broker unterstützten sein (siehe „Von WebSphere Message Broker unterstützte Nachrichtentransporte“ auf Seite 192).

In den folgenden Abschnitten wird die Erstellung von Zielen für die folgenden, von Message Broker unterstützten Transportprotokolle zusammengefasst:

- „Dokumentenversand über das HTTP-Transportprotokoll konfigurieren“
- „Dokumentenempfang über das JMS-Transportprotokoll konfigurieren“

### Dokumentenversand über das HTTP-Transportprotokoll konfigurieren

Wenn der Hub ein Dokument über das HTTP-Protokoll an Message Broker sendet, leitet er die Nachricht über das definierte Ziel weiter. Diese Ziel gibt die URL-Adresse an, über die das Dokument von Message Broker empfangen werden kann. Wenn Message Broker das HTTP-Protokoll verwendet, leitet dieses Produkt das Dokument an den HTTPInput-Knoten des Nachrichtenflusses, der der angegebenen URL-Adresse zugeordnet ist.

Damit der Hub Dokumente mit dem HTTP-Transportprotokoll über ein Ziel senden kann, müssen Sie auf der Seite **Zieldetails** der Community Console ein Gateway erstellen. Dieses Ziel muss so konfiguriert sein, dass das Transportprotokoll HTTP 1.1 verwendet wird und Schreiboperationen an die URL-Adresse gerichtet sind, für die der zugehörige HTTPInput-Knoten empfangsbereit ist. Wie Tabelle 72 zeigt, müssen Sie diese URL-Adresse im Feld für die Empfänger-URI der Zieldefinition angeben.

**Anmerkung:** Eine Übersicht zur Erstellung eines Ziels finden Sie im Abschnitt „Zieleinheit für Partnerdokument definieren“ auf Seite 49.

*Tabelle 72. HTTP-Werte für die Seite 'Zieldetails' zur Kommunikation mit WebSphere Message Broker*

Feld der Empfänger-details	Wert	Anmerkungen und Einschränkungen
Empfänger-URI	Die URL-Adresse muss mit der für den HTTPInput-Knoten konfigurierten URL-Adresse im Message Broker-Nachrichtenfluss übereinstimmen.	Ermitteln Sie diese URL-Adresse aus der Konfiguration des Nachrichtenflusses in der WebSphere Message Broker-Integration.

### Dokumentenempfang über das JMS-Transportprotokoll konfigurieren

Wenn der Hub ein Dokument über das JMS-Protokoll an Message Broker sendet, leitet er die Nachricht an die entsprechende JMS-Warteschlange weiter, von der sie an die JMS-Warteschlange übertragen werden kann, aus der sie von Message Broker abgerufen werden kann. Damit der Hub auf diese JMS-Position zugreifen kann, müssen Sie in WebSphere Partner Gateway ein Ziel erstellen, das das JMS-Transportprotokoll verwendet. Dieses Ziel muss zum Schreiben an die Warteschlange konfiguriert werden, deren Inhalt an die Warteschlange übertragen wird, aus der Message Broker Nachrichten empfängt.

**Anmerkung:** Eine Übersicht zur Erstellung eines Ziels finden Sie im Abschnitt „Zieleinheit für Partnerdokument definieren“ auf Seite 49.

Damit der Hub Dokumente über ein Ziel mit dem JMS-Transportprotokoll senden kann, erstellen Sie auf der Seite **Zieldetails** der Community Console ein Ziel. Wenn Sie als JMS-Provider WebSphere MQ Version 6.0 mit dem neuesten Fixpack verwenden, sollten Sie die detaillierten Anweisungen zu diesem Thema im Handbuch *WebSphere Partner Gateway Hubkonfiguration* lesen. Verwenden Sie darüber hinaus die in Tabelle 73 aufgeführten Informationen für das JMS-Protokoll auf der Seite **Zieldetails**.

*Tabelle 73. JMS-Werte für die Seite 'Zieldetails' zur Kommunikation mit WebSphere Message Broker*

Feld der Zieldetails	Wert	Anmerkungen und Einschränkungen
JMS-Warteschlangenname	Name der JMS-Warteschlange auf dem Computer, auf dem sich WebSphere Partner Gateway befindet.	In dieser Warteschlange empfangene Dokumente werden an die JMS-Warteschlange auf dem Computer übertragen, auf dem sich Message Broker befindet.

## Unterstützung für eingehende Dokumente bereitstellen

Damit WebSphere Partner Gateway Nachrichten von einem beliebigen Back-End-System empfangen kann, müssen Sie die im Abschnitt „Abrufposition für Back-End-Dokumente definieren“ auf Seite 53 beschriebenen Schritte ausführen. Wenn Message Broker Ihr Back-End-System ist, müssen Sie die folgenden Schritte ausführen:

1. Definieren Sie im Partnerprofil für den internen Partner den Zieltyp, und geben Sie die zugehörige IP-Adresse an, für die der Empfänger empfangsbereit ist.
2. Erstellen Sie einen Empfänger, dessen Transporttyp mit dem Transportprotokoll übereinstimmt, das für Dokumente zwischen WebSphere Partner Gateway und Message Broker verwendet wird.

Damit der Hub ein Dokument von Message Broker empfangen kann, muss er die Position kennen, an der die Nachrichten abzurufen sind. Diese Position muss dem zu verwendenden Transportprotokoll entsprechen.

In den folgenden Abschnitten wird die Erstellung von Empfängern für Transportprotokolle zusammengefasst, die von Message Broker unterstützt werden.

### Dokumenteneingang über das HTTP-Transportprotokoll konfigurieren

Wenn der Hub ein Dokument über das HTTP-Transportprotokoll empfängt, ruft der Empfänger das Dokument aus dem definierten Empfänger ab. Dieser Empfänger gibt die URL-Adresse an, für die der Empfänger für Dokumente von Message Broker empfangsbereit ist. Wenn Message Broker mit dem HTTP-Transportprotokoll arbeitet, sendet der HTTPRequest-Knoten das Dokument an die entsprechende URL-Adresse, an der es vom Hub empfangen werden kann.

Damit der Hub Dokumente über einen Empfänger mit dem HTTP-Transportprotokoll empfangen kann, müssen Sie auf der Seite **Empfängerliste** der Community Console ein Ziel erstellen. Dieser Empfänger muss das Transportprotokoll HTTP 1.1 verwenden. Der Hub bestimmt diese URL-Adresse als Kombination aus folgenden Informationen:

- Die IP-Adresse des Host-Computers, die aus dem Profil für den internen Partner ermittelt wird
- Die Empfänger-URL-Adresse, die aus dem URL-Feld der Empfängerdefinition ermittelt wird

**Anmerkung:** Eine Übersicht zur Erstellung eines Empfängers finden Sie im Abschnitt „Abrufposition für Back-End-Dokumente definieren“ auf Seite 53.

Damit Message Broker Dokumente an diesen Empfänger senden kann, muss der HTTPRequest-Knoten des Nachrichtenflusses zum Senden von Dokumenten an diese URL-Adresse konfiguriert werden. Daher müssen Sie sicherstellen, dass diese Empfänger-URL-Adresse für die Message Broker-Konfiguration verfügbar ist.

### **Dokumenteneingang über das JMS-Transportprotokoll konfigurieren**

Wenn der Hub ein Dokument von Message Broker über das JMS-Protokoll empfängt, ruft er das Dokument aus der entsprechenden JMS-Eingabewarteschlange ab, in die es von der JMS-Ausgabewarteschlange übertragen wurde, an die sie von Message Broker gesendet wurde. Damit der Hub auf diese JMS-Position zugreifen kann, müssen Sie in WebSphere Partner Gateway einen Empfänger erstellen, der das JMS-Transportprotokoll verwendet. Über den Empfänger ist der Hub für alle Dokumente in seiner Eingabewarteschlange empfangsbereit und ruft sie ab.

**Anmerkung:** Eine Übersicht zur Erstellung eines Empfängers finden Sie im Abschnitt „Abrufposition für Back-End-Dokumente definieren“ auf Seite 53.

Damit der Hub Dokumente über einen Empfänger mit dem JMS-Transportprotokoll empfangen kann, müssen Sie auf der Seite **Empfängerliste** der Community Console einen Empfänger erstellen. Wenn Sie als JMS-Provider WebSphere MQ Version 6.0 mit dem neuesten Fixpack verwenden, sollten Sie die detaillierten Anweisungen zu diesem Thema im Handbuch *WebSphere Partner Gateway Hub-konfiguration* lesen. Verwenden Sie darüber hinaus die in Tabelle 74 aufgeführten Informationen für das JMS-Protokoll auf der Seite **Empfängerdetails**.

*Tabelle 74. JMS-Werte für die Seite 'Empfängerdetails' zur Kommunikation mit WebSphere Message Broker*

<b>Feld der Empfängerdetails</b>	<b>Wert</b>	<b>Anmerkungen und Einschränkungen</b>
JMS-Warteschlangename	Name der JMS-Eingabewarteschlange, die Dokumente aus der Ausgabewarteschlange von Message Broker empfängt	Dokumente in dieser Eingabewarteschlange wurden aus der JMS-Ausgabewarteschlange auf dem Computer übertragen, auf dem sich Message Broker befindet.

---

## **WebSphere Message Broker konfigurieren**

Für die Interaktionen zwischen WebSphere Partner Gateway und Message Broker müssen Sie ein Nachrichtenflussprojekt (Message Flow) innerhalb der Broker-Perspektive für die Anwendungsentwicklung (Broker Application Development Perspective) des Toolkits von Message Broker erstellen. Dieses Projekt muss die folgenden Artefakte enthalten:

- Nachrichtenflüsse
- PIP-Dateien (nur RosettaNet) oder Nachrichtendefinitionsdateien

**Anmerkung:** Weitere Informationen zur Erstellung von Nachrichtenflussprojekten finden Sie in der Dokumentation zu WebSphere Message Broker.

## Nachrichtenfluss erstellen

Die eigentliche Geschäftsprozesslogik, die Sie zur Verarbeitung von Informationen benötigen, wird durch den *Nachrichtenfluss* innerhalb von Message Broker ausgeführt. Daher muss der entsprechende Nachrichtenfluss für Message Broker vorhanden sein, um Ihre WebSphere Partner Gateway-Dokumente korrekt zu verarbeiten. Stellen Sie sicher, dass ein Nachrichtenfluss vorhanden ist, der die benötigte Geschäftslogik zur Verfügung stellt:

- Wenn ein solcher Nachrichtenfluss momentan *nicht* vorhanden ist, müssen Sie einen erstellen oder importieren.
- Wenn ein Nachrichtenfluss vorhanden ist, müssen Sie mit seiner Verwendungsweise vertraut sein.

Zur Verarbeitung eingehender und ausgehender Dokumente durch Message Broker verwendet der entsprechende Nachrichtenfluss besondere Transportknoten. Der Typ des zu verwendenden Transportknotens hängt, wie gezeigt, vom jeweiligen Transportprotokoll ab.

Tabelle 75. Erstellen von Nachrichtenflüssen für verschiedene Transportprotokolle

Transportprotokoll	Weitere Informationen
HTTP HTTP (SOAP-Dokumente)	„Nachrichtenfluss für den HTTP-Transport erstellen“ auf Seite 199
JMS	„Nachrichtenfluss für den JMS-Transport erstellen“ auf Seite 208

## Projekt implementieren

Wenn Ihr Nachrichtenflussprojekt die richtigen Artefakte enthält, müssen Sie es in Message Broker implementieren. Die Implementierung eines Nachrichtenflussprojekts erfolgt über die Broker-Administratorperspektive (Broker Administrator Perspective) des Toolkits von Message Broker.

---

## HTTP-Transportprotokoll mit WebSphere Message Broker verwenden

In diesem Abschnitt wird das Senden und Empfangen von Dokumenten zwischen WebSphere Partner Gateway und WebSphere Message Broker unter Verwendung des HTTP-Transportprotokolls beschrieben.

**Anmerkung:** Alle Verweise auf das HTTP-Transportprotokoll sind ebenso auf HTTPS anwendbar.

## Erforderliche Komponenten für den Austausch von Dokumenten über den HTTP-Transport

Zum Senden und Empfangen von Dokumenten zwischen WebSphere Partner Gateway und Message Broker über das HTTP-Protokoll benötigen Sie keine weitere Software. Benötigt werden lediglich WebSphere Partner Gateway und Message Broker. Für die Kommunikation von WebSphere Partner Gateway mit Version 5.0 von Message Broker über das HTTP-Transportprotokoll müssen diese beiden Komponenten konfiguriert werden. In Tabelle 76 sind diese Konfigurationsschritte zusammenfasst.

Tabelle 76. WebSphere Partner Gateway und WebSphere Message Broker konfigurieren

Komponente	Version	Weitere Informationen
WebSphere Partner Gateway	6.1	„Dokumentenversand über das HTTP-Transportprotokoll konfigurieren“ auf Seite 194  „Dokumenteneingang über das HTTP-Transportprotokoll konfigurieren“ auf Seite 195
WebSphere Message Broker	6.0 mit neuestem Fixpack	„WebSphere Message Broker konfigurieren“ auf Seite 196

Darüber hinaus müssen Sie zum Senden oder Empfangen eines Dokuments zwischen WebSphere Partner Gateway und Message Broker über das HTTP-Transportprotokoll Version 6.0 mit neuestem Fixpack von IBM WebSphere MQ als JMS-Provider einsetzen.

### Dokumente über den HTTP-Transport senden

Beim Senden eines Dokuments von WebSphere Partner Gateway an Message Broker über das HTTP-Transportprotokoll verwenden Sie spezielle HTTP-Transportknoten innerhalb des Message Broker-Nachrichtenflusses, um das von WebSphere Partner Gateway als HTTP-Datenstrom gesendete Dokument abzurufen. Die Knoten des Nachrichtenflusses führen die erforderliche Verarbeitung aus und leiten das Dokument anschließend an ein Ziel (eine JMS-Ausgabewarteschlange) weiter.

Die folgenden Schritte beschreiben, wie WebSphere Partner Gateway über das HTTP-Transportprotokoll ein Dokument an einen Nachrichtenfluss in Message Broker sendet:

1. WebSphere Partner Gateway sendet eine HTTP-Nachricht an Message Broker. Wenn der Pakettyp des Dokuments 'Backend Integration' ist, wurden von WebSphere Partner Gateway angepasste Eigenschaften in dieser Nachricht bereitgestellt.

**Anmerkung:** Innerhalb von WebSphere Partner Gateway müssen Sie ein Ziel konfigurieren, das die URL-Adresse angibt, an die WebSphere Partner Gateway die Nachricht sendet und die von Message Broker abgefragt wird. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumentenversand über das HTTP-Transportprotokoll konfigurieren“ auf Seite 194.

2. Der HTTPInput-Knoten des Nachrichtenflusses nimmt das Dokument auf und sendet es an den nächsten Knoten des Nachrichtenflusses. Dieser Knoten ist in der Regel ein Berechnungsknoten (Compute-Knoten).
3. Die Knoten des Nachrichtenflusses führen die Geschäftslogik aus. Wenn die Geschäftslogik abgeschlossen ist, sendet der Nachrichtenfluss das resultierende Dokument an den HTTPReply-Knoten.
4. Der HTTPReply-Knoten sendet die Ausgabenachricht standardmäßig an den Client (WebSphere Partner Gateway) zurück.

Alternativ kann der Nachrichtenfluss die Nachricht in einen MQOutput-Knoten stellen. Der MQOutput-Knoten empfängt das Dokument und sendet es an die vorgesehene JMS-Warteschlange bzw. an eine andere Anwendung.

## Dokumente über den HTTP-Transport empfangen

Zum Empfangen eines Dokuments von Message Broker durch WebSphere Partner Gateway über das HTTP-Transportprotokoll verwenden Sie spezielle HTTP-Transportknoten innerhalb des Message Broker-Nachrichtenflusses, um das Dokument, das von WebSphere Partner Gateway empfangen werden soll, als HTTP-Datenstrom zu senden. Die Knoten des Nachrichtenflusses führen die erforderliche Verarbeitung aus und erledigen die Anforderung und die Antwort (bei synchroner Interaktion) mit WebSphere Partner Gateway.

Die folgenden Schritte beschreiben, wie WebSphere Partner Gateway über das HTTP-Transportprotokoll ein Dokument von einem Nachrichtenfluss in Message Broker empfängt:

1. Der Nachrichtenfluss innerhalb von Message Broker empfängt ein Dokument in seinem MQInput-Knoten (einer JMS-Eingabewarteschlange).
2. Der MQInput-Knoten des Nachrichtenflusses empfängt das Dokument und sendet es an den HTTPRequest-Knoten.
3. Der HTTPRequest-Knoten führt die Anforderungs- und Antwortinteraktionen mit dem Client (WebSphere Partner Gateway) über die angegebene URL-Adresse aus.
4. WebSphere Partner Gateway empfängt die Nachricht von der zugehörigen URL-Adresse, die für den Empfänger konfiguriert ist.

Weitere Informationen zum Empfänger finden Sie im Abschnitt „Dokumenteingang über das HTTP-Transportprotokoll konfigurieren“ auf Seite 195.

## Nachrichtenfluss für den HTTP-Transport erstellen

Ein Message Broker-Nachrichtenfluss verwendet zur Verarbeitung von Dokumenten über das HTTP-Transportprotokoll die folgenden Transportknoten:

- HTTPInput
- HTTPReply
- HTTPRequest

Die Verwendungsreihenfolge dieser Transportknoten hängt wie folgt von der Kommunikationsrichtung ab:

- Wenn WebSphere Partner Gateway ein Dokument an Message Broker *sendet*, umfasst der Nachrichtenfluss die in Tabelle 77 aufgeführten Typen von Knoten (in der gezeigten Reihenfolge), um die Geschäftslogik zu beschreiben.
- Wenn WebSphere Partner Gateway ein Dokument von Message Broker *empfängt*, umfasst der Nachrichtenfluss die in Tabelle 78 aufgeführten Typen von Knoten (in der gezeigten Reihenfolge), um die Geschäftslogik zu beschreiben.

Tabelle 77. Knoten zum Senden von Dokumenten an WebSphere Message Broker über HTTP

Knotentyp	Zweck	Anmerkungen
HTTPInput	Empfängt das WebSphere Partner Gateway-Anforderungsdokument in einem Nachrichtenfluss.	<p>Setzen Sie das Feld für den URL-Selektor (URL Selector) in den Basiseigenschaften (Basic Properties) dieses Transportknotens auf die URL-Adresse, an die WebSphere Partner Gateway Dokumente sendet (die URL-Adresse, die im WebSphere Partner Gateway-Empfänger konfiguriert ist). Die URL-Adresse muss folgendes Format haben:</p> <p><code>http://hostname:port/pfad</code></p> <p>Dabei ist <i>hostname</i> der Name des Computers, auf dem sich Message Broker befindet, <i>port</i> die HTTP-Portnummer, an der Message Broker empfangsbereit ist, und <i>pfad</i> die Angabe der Position auf diesem Computer.</p> <p>Weitere Informationen hierzu finden Sie im Abschnitt „Dokumentenversand über das HTTP-Transportprotokoll konfigurieren“ auf Seite 194.</p>
Compute	Führt die Operationen der Geschäftslogik aus, z. B. die Aktualisierung von Headerinformationen.	Verwenden Sie ESQL zur Ausführung der Geschäftslogik. Der Berechnungsknoten sendet die resultierende Nachricht an den HTTPReply-Knoten.
HTTPReply	Gibt eine Antwort an WebSphere Partner Gateway zurück.	Standardmäßig sendet dieser Knoten die Ausgabemessage an den Client. Sie können ihn jedoch auch zum Senden an einen MQOutput-Knoten konfigurieren.
MQOutput	Empfängt das Dokument vom HTTPReply-Knoten und sendet es an WebSphere Partner Gateway.	Dieser Transportknoten sendet das resultierende Dokument an eine JMS-Ausgabewarteschlange, die sie an das nächste Ziel weiterleitet.

Tabelle 78. Knoten zum Empfangen von Dokumenten von WebSphere Message Broker über HTTP

Knotentyp	Zweck	Anmerkungen
MQInput	Empfängt das Dokument von WebSphere Partner Gateway.	Dieser Transportknoten empfängt das eingehende Dokument aus einer JMS-Eingabewarteschlange.
HTTPRequest	Führt Anforderungs-/Antwortinteraktionen mit WebSphere Partner Gateway aus.	<p>In diesem Transportknoten muss das Feld der Web-Services-URL in den Basiseigenschaften (Basic Properties) auf die URL-Adresse gesetzt werden, an der WebSphere Partner Gateway für Dokumente empfangsbereit ist (die URL-Adresse, die im WebSphere Partner Gateway-Empfänger konfiguriert ist). Die URL-Adresse muss folgendes Format haben:</p> <p><code>http://hostname:port/bcgreceiver/pfad</code></p> <p>Dabei ist <i>hostname</i> der Name des Computers, auf dem sich WebSphere Partner Gateway befindet, <i>port</i> die HTTP-Portnummer, an der die Empfangskomponente (Receiver) von WebSphere Partner Gateway empfangsbereit ist, und <i>pfad</i> die Angabe der Position auf diesem Computer.</p> <p>Weitere Informationen hierzu finden Sie im Abschnitt „Dokumenteneingang über das HTTP-Transportprotokoll konfigurieren“ auf Seite 195.</p>

Weitere detaillierte Informationen zur Erstellung und Konfiguration von Nachrichtenflussknoten finden Sie in der Dokumentation zu WebSphere Message Broker.

## SOAP-Dokumente senden

SOAP-Dokumente unterscheiden sich von allen anderen Dokumenttypen, die über HTTP/S ausgetauscht werden. Im folgenden Abschnitt wird beschrieben, wie SOAP-Dokumente zwischen WebSphere Partner Gateway und WebSphere Message Broker über das HTTP-Transportprotokoll gesendet und empfangen werden.

Die Konfiguration von WebSphere Partner Gateway und Message Broker für die Übertragung von SOAP-Dokumenten ist der Konfiguration zur Übertragung von Nicht-SOAP-Dokumenten über das HTTP-Protokoll sehr ähnlich. Tabelle 79 enthält eine Übersicht, wo Sie Informationen zur Konfiguration dieser beiden Integrationskomponenten finden.

*Tabelle 79. WebSphere Partner Gateway und WebSphere Message Broker zur Übertragung von SOAP-Dokumenten konfigurieren*

Integrationskomponente	Konfigurationsschritt	Weitere Informationen
WebSphere Partner Gateway	Zum Senden von Dokumenten über HTTP konfigurieren Sie den Empfänger und das Ziel für SOAP-Dokumente in der gleichen Weise wie für Nicht-SOAP-Dokumente.	„Dokumentenversand über das HTTP-Transportprotokoll konfigurieren“ auf Seite 194  „Dokumenteneingang über das HTTP-Transportprotokoll konfigurieren“ auf Seite 195
WebSphere Message Broker	Die Nachrichtenflüsse zur Verarbeitung von SOAP-Dokumenten sind denen für Nicht-SOAP-Dokumente, die über HTTP gesendet werden, sehr ähnlich. Zur Verarbeitung von SOAP-Dokumenten ist nur ein zusätzlicher Transportknoten erforderlich.	Informationen zum Senden eines SOAP-Dokuments an Message Broker finden Sie in Tabelle 80 auf Seite 202.  Informationen zum Empfangen eines SOAP-Dokuments von Message Broker finden Sie in „Nachrichtenfluss für den HTTP-Transport erstellen“ auf Seite 199.

Damit Message Broker ein SOAP-Dokument, das von WebSphere Partner Gateway gesendet wird, korrekt verarbeiten kann, muss der Nachrichtenfluss einen HTTPRequest-Knoten zur Ausführung der Kommunikation mit dem Web-Services-Client enthalten. In Tabelle 80 sind die Knoten aufgeführt, die in einem Message Broker-Nachrichtenfluss zur Verarbeitung eines von WebSphere Partner Gateway gesendeten SOAP-Dokuments erforderlich sind.

Table 80. Knoten zum Senden von SOAP-Dokumenten an WebSphere Message Broker

Knotentyp	Zweck	Anmerkungen
HTTPInput	Empfängt das WebSphere Partner Gateway-Anforderungsdokument in einem Nachrichtenfluss.	<p>Setzen Sie das Feld für den URL-Selektor (URL Selector) in den Basiseigenschaften (Basic Properties) dieses Transportknotens auf die URL-Adresse, an die WebSphere Partner Gateway Dokumente sendet (die URL-Adresse, die im Ziel von WebSphere Partner Gateway konfiguriert ist). Die URL-Adresse muss folgendes Format haben:  <code>http://hostname:port/pfad</code></p> <p>Dabei ist <i>hostname</i> der Name des Computers, auf dem sich WebSphere Partner Gateway befindet, <i>port</i> die HTTP-Portnummer, an der die Empfangskomponente (Receiver) von WebSphere Partner Gateway empfangsbereit ist, und <i>pfad</i> die Angabe der Position auf diesem Computer.</p> <p>Weitere Informationen hierzu finden Sie im Abschnitt „Dokumentenversand über das HTTP-Transportprotokoll konfigurieren“ auf Seite 194.</p>
Compute	Führt die Operationen der Geschäftslogik aus, z. B. die Aktualisierung von Headerinformationen.	Verwenden Sie ESQL zur Ausführung der Geschäftslogik. Der Berechnungsknoten sendet die resultierende Nachricht an den HTTPReply-Knoten.
HTTPRequest	Sendet die SOAP-Anforderung an den externen Web-Service-Provider (WebServices) und empfängt eine Antwort von diesem Web-Service.	Keine
HTTPReply	Gibt eine Antwort an WebSphere Partner Gateway zurück.	Standardmäßig sendet dieser Knoten die Ausgabemessage an den Client.

---

## JMS-Transportprotokoll mit WebSphere Message Broker verwenden

In diesem Abschnitt wird beschrieben, wie Komponenten zum Senden und Empfangen von Dokumenten zwischen WebSphere Partner Gateway und WebSphere Message Broker unter Verwendung des JMS-Transportprotokolls konfiguriert werden. Es enthält die folgenden Informationen zum Senden und Empfangen von Dokumenten:

- „Erforderliche Komponenten für den Austausch von Dokumenten über den JMS-Transport“
- „Dokumentenversand über das JMS-Transportprotokoll“ auf Seite 174
- „Dokumentenempfang über das JMS-Transportprotokoll“ auf Seite 176

### Erforderliche Komponenten für den Austausch von Dokumenten über den JMS-Transport

Zum Senden oder Empfangen eines Dokuments zwischen WebSphere Partner Gateway und Message Broker Version 6.0 über das JMS-Transportprotokoll ist WebSphere MQ als JMS-Provider erforderlich. In den folgenden Abschnitten wird beschrieben, wie WebSphere Partner Gateway, Message Broker und WebSphere MQ zusammenarbeiten, um Dokumente über das JMS-Transportprotokoll auszutauschen.

#### Dokumentenversand über das JMS-Transportprotokoll

Beim Senden eines Dokuments von WebSphere Partner Gateway an Message Broker über das JMS-Transportprotokoll verwenden Sie die JMS-Warteschlangen. WebSphere Partner Gateway sendet ein Dokument an die vorgesehene JMS-Ausgabewarteschlange, von der aus es an die JMS-Eingabewarteschlange übertragen wird, über die Message Broker empfangsbereit ist. Beim Empfangen eines Dokuments ruft Message Broker es aus seiner Eingabewarteschlange ab. Der Nachrichtenfluss von Message Broker enthält spezielle (JMS-)Transportknoten für WebSphere MQ, die den Zugriff auf die JMS-Warteschlangen ausführen. In Abb. 38 ist zusammengefasst, wie WebSphere Partner Gateway über das JMS-Transportprotokoll Dokumente an Message Broker sendet.

## WebSphere Business Integration Message Broker

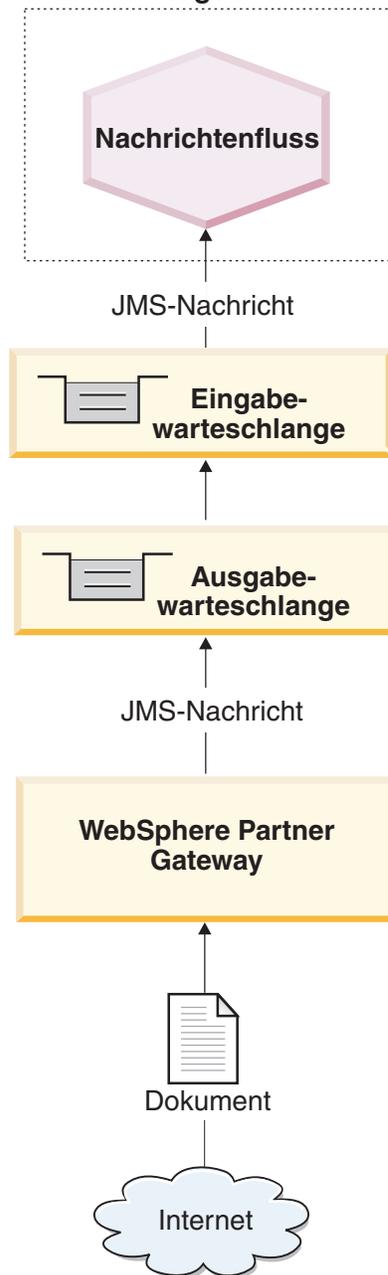


Abbildung 38. Nachrichtenfluss über das JMS-Transportprotokoll von WebSphere Partner Gateway an einen Nachrichtenfluss

Die folgenden Schritte beschreiben, wie WebSphere Partner Gateway über das JMS-Transportprotokoll ein Dokument an einen Nachrichtenfluss in Message Broker sendet:

1. WebSphere Partner Gateway übergibt eine Nachricht an die JMS-Ausgabewarteschlange.

Wenn der Pakettyp des Dokuments 'Backend Integration' ist, wurden von WebSphere Partner Gateway angepasste Eigenschaften in dieser Nachricht bereitgestellt. Der JMS-Nachrichtenheader 'JMSType' wird mit dem Inhaltstyp (content-type) der Nutzdaten definiert.

**Anmerkung:** Innerhalb von WebSphere Partner Gateway müssen Sie ein Ziel konfigurieren, das die JMS-Ausgabewarteschlange angibt, an die WebSphere Partner Gateway die Nachricht sendet und die von Message Broker abgefragt wird. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumentenempfang über das JMS-Transportprotokoll konfigurieren“ auf Seite 194.

2. WebSphere MQ überträgt das Dokument aus der Ausgabewarteschlange auf dem Computer, auf dem sich WebSphere Partner Gateway befindet, an die Eingabewarteschlange, die von Message Broker abgefragt wird.
3. Wenn Message Broker eine Nachricht in seiner Eingabewarteschlange erkennt, ruft Message Broker die Nachricht ab und sendet sie an den entsprechenden Nachrichtenfluss.  
Informationen zur Konfiguration dieser Warteschlange finden Sie im Abschnitt „Umgebung für den JMS-Transport einrichten“ auf Seite 207.
4. Der MQInput-Knoten sendet das Dokument an den nächsten Knoten des Nachrichtenflusses. Dieser Knoten ist in der Regel ein Berechnungsknoten (Compute-Knoten).
5. Die Knoten des Nachrichtenflusses führen die Geschäftslogik aus.  
Wenn die Geschäftslogik abgeschlossen ist, sendet der Nachrichtenfluss das resultierende Dokument an den MQOutput-Knoten.
6. Der MQOutput-Knoten sendet das Dokument an die entsprechende Warteschlange.

### **Dokumentenempfang über das JMS-Transportprotokoll**

Zum Empfangen eines Dokuments von Message Broker durch WebSphere Partner Gateway über das JMS-Transportprotokoll verwenden Sie JMS-Warteschlangen. Message Broker sendet ein Dokument an die vorgesehene JMS-Ausgabewarteschlange, von der aus es an die JMS-Eingabewarteschlange übertragen wird, über die WebSphere Partner Gateway empfangsbereit ist. Wenn WebSphere Partner Gateway ein Dokument empfängt, ruft es es aus seiner Eingabewarteschlange ab. Der Nachrichtenfluss von Message Broker enthält spezielle (JMS-)Transportknoten für WebSphere MQ, die den Zugriff auf die JMS-Warteschlangen ausführen. Abb. 39 veranschaulicht, wie Dokumente von Message Broker an WebSphere Partner Gateway gesendet werden.

## WebSphere Business Integration Message Broker

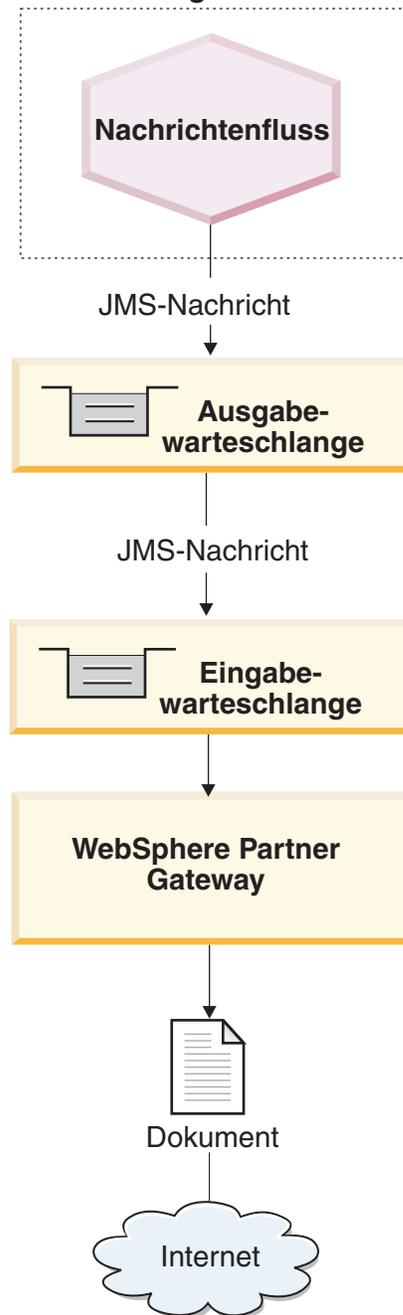


Abbildung 39. Nachrichtenfluss über das JMS-Transportprotokoll aus einem Nachrichtenfluss an WebSphere Partner Gateway

Die folgenden Schritte beschreiben, wie WebSphere Partner Gateway über das JMS-Transportprotokoll ein Dokument von einem Nachrichtenfluss in Message Broker empfängt:

1. Der Nachrichtenfluss innerhalb von Message Broker empfängt ein Dokument in seinem MQInput-Knoten.  
Der Nachrichtenfluss empfängt die eingehende Nachricht aus einer JMS-Eingabewarteschlange.

2. Die MQInput-Warteschlange des Nachrichtenflusses empfängt das Dokument und sendet es an den nächsten Knoten des Nachrichtenflusses. Dieser Knoten ist in der Regel ein Berechnungsknoten (Compute-Knoten).
3. Die Knoten des Nachrichtenflusses führen die Geschäftslogik aus.  
Wenn die Geschäftslogik abgeschlossen ist, sendet der Nachrichtenfluss das resultierende Dokument an den MQOutput-Knoten.
4. Der MQOutput-Knoten sendet das Dokument an die entsprechende JMS-Ausgabewarteschlange.
5. WebSphere MQ überträgt das Dokument aus der Warteschlange auf dem Computer, auf dem sich Message Broker befindet, an die Warteschlange, die von WebSphere Partner Gateway abgefragt wird.
6. WebSphere Partner Gateway empfängt die Nachricht aus der zugehörigen JMS-Eingabewarteschlange, die im Empfänger konfiguriert ist.  
Weitere Informationen zum Empfänger finden Sie im Abschnitt „Dokumenteingang über das JMS-Transportprotokoll konfigurieren“ auf Seite 196. Informationen zur Einrichtung dieser Warteschlange finden Sie im Abschnitt „Umgebung für den JMS-Transport einrichten“.

## Umgebung für den JMS-Transport einrichten

Für das Senden und Empfangen von Dokumenten mit Message Broker sind (ferne und lokale) JMS-Warteschlangen erforderlich. Informationen zur Konfiguration von WebSphere Partner Gateway zur Verwendung mit Message Broker über JMS finden Sie im Abschnitt „WebSphere Partner Gateway für die Verwendung mit WebSphere Message Broker konfigurieren“ auf Seite 193. Zur Verwendung des JMS-Transportprotokolls mit Message Broker können Sie das von WebSphere MQ zur Verfügung gestellte JMS-System einrichten. Version 6.0 von Message Broker arbeitet mit Version 6.0 von WebSphere MQ als JMS-Provider. Zum Konfigurieren des Mechanismus für das JMS-Transportprotokoll können die Anweisungen im Handbuch *WebSphere Partner Gateway Hubkonfiguration* verwendet werden.

**Wichtig:** Die Schritte im Handbuch *WebSphere Partner Gateway Hubkonfiguration* müssen auf dem Computer ausgeführt werden, auf dem sich WebSphere Partner Gateway befindet. In diesem Handbuch wird davon ausgegangen, dass der JMS-Transportmechanismus, der für Message Broker erforderlich ist, im Rahmen der Installation von Message Broker bereits konfiguriert wurde.

Wenn Sie Ihre JMS-Warteschlangen für die Verwendung zwischen WebSphere Partner Gateway und Message Broker erstellen, beachten Sie folgende Punkte:

- Bei der Installation von Message Broker sollten folgende Warteschlangenmanager erstellt werden:
  - Ein WebSphere MQ-Warteschlangenmanager, der der Brokerdomäne zugeordnet ist  
Mit dem folgenden Befehl können Sie diesen Warteschlangenmanager sowie eine Reihe benannter Warteschlangen erstellen:  
`mqsicreatebroker`
  - Ein WebSphere MQ-Warteschlangenmanager für Message Broker  
Da Message Broker eine Reihe vorbestimmter Warteschlangennamen verwendet, ist pro Broker ein separater WebSphere MQ-Warteschlangenmanager erforderlich. Message Broker kann diesen Warteschlangenmanager gemeinsam benutzen, der entweder zusammen mit dem Konfigurationsmanager (Configuration Manager) oder dem optionalen Benutzernamensserver (User Name Server) oder beiden betrieben wird.

Weitere Informationen finden Sie im Handbuch *WebSphere Message Broker Installation and Configuration Guide*.

- Wenn Sie die Aliasnamen für Ihre JMS-Warteschlangen erstellen, ist es sinnvoll, sie so zu benennen, dass sie einen Hinweis auf die Flussrichtung zwischen WebSphere Partner Gateway und Message Broker geben.

Wenn Sie zum Beispiel die in der Spalte Originalwarteschlangenname von Tabelle 65 aufgeführten Warteschlangen erstellen, könnten Sie diese Warteschlangen so umbenennen, dass sie die Flussrichtung wie in der Spalte Richtungsbezogener Warteschlangenname von Tabelle 81 gezeigt angeben.

Tabelle 81. Benennen von JMS-Warteschlangen mit Richtungsangabe für WebSphere Message Broker

Originalwarteschlangenname	Richtungsbezogener Warteschlangenname
inQ	MB2HUB
outQ	HUB2MB

## Nachrichtenfluss für den JMS-Transport erstellen

Ein Message Broker-Nachrichtenfluss verwendet zur Verarbeitung von Dokumenten über das JMS-Transportprotokoll die folgenden Transportknoten:

- MQInput
- MQOutput

Die Verwendungsreihenfolge dieser Transportknoten hängt wie folgt von der Kommunikationsrichtung ab:

- Wenn WebSphere Partner Gateway ein Dokument an Message Broker *sendet*, umfasst der Nachrichtenfluss die in Tabelle 82 aufgeführten Typen von Knoten (in der gezeigten Reihenfolge), um die Geschäftslogik zu beschreiben.
- Wenn WebSphere Partner Gateway ein Dokument von Message Broker *empfängt*, umfasst der Nachrichtenfluss die in Tabelle 83 aufgeführten Typen von Knoten (in der gezeigten Reihenfolge), um die Geschäftslogik zu beschreiben.

Tabelle 82. Knoten zum Senden von Dokumenten an WebSphere Message Broker über JMS

Knotentyp	Zweck	Anmerkungen und Einschränkungen
MQInput	Empfängt das Dokument von WebSphere Partner Gateway.	Der Wert im Feld für den Warteschlangenname (Queue Name) in den Basis-eigenschaften (Basic Properties) dieses Transportknotens ist die Eingabewarteschlange des Nachrichtenflusses. WebSphere MQ muss so eingerichtet werden, dass diese JMS-Warteschlange Dokumente aus der Ausgabewarteschlange von WebSphere Partner Gateway empfängt. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumentenempfang über das JMS-Transportprotokoll konfigurieren“ auf Seite 194.
Compute	Führt die Operationen der Geschäftslogik aus, z. B. das Entfernen von Headerinformationen.	Keine
MQOutput	Empfängt das Dokument vom Berechnungsknoten (Compute-Knoten) und sendet es als Ausgabe des Nachrichtenflusses.	Dieser Transportknoten sendet das resultierende Dokument an eine JMS-Ausgabewarteschlange, die sie an das nächste Ziel weiterleitet.

Tabelle 83. Knoten zum Empfangen von Dokumenten von WebSphere Message Broker

Knotentyp	Zweck	Anmerkungen und Einschränkungen
MQInput	Empfängt das Dokument im Nachrichtenfluss.	Dieser Transportknoten empfängt das eingehende Dokument aus einer JMS-Eingabewarteschlange.
Compute	Führt die Operationen der Geschäftslogik aus, z. B. die Aktualisierung von Headerinformationen.	Keine
MQOutput	Empfängt das Dokument vom Berechnungsknoten und sendet es an WebSphere Partner Gateway.	Der Wert im Feld für den Warteschlangennamen (Queue Name) in den Basiseigenschaften (Basic Properties) dieses Transportknotens ist die Ausgabewarteschlange des Nachrichtenflusses. WebSphere MQ muss so eingerichtet werden, dass diese JMS-Warteschlange Dokumente an die Eingabewarteschlange von WebSphere Partner Gateway sendet. Weitere Informationen hierzu finden Sie im Abschnitt „Dokumenteneingang über das JMS-Transportprotokoll konfigurieren“ auf Seite 196.

Weitere detaillierte Informationen zur Erstellung und Konfiguration von Nachrichtenflussknoten finden Sie in der Dokumentation zu WebSphere Message Broker.



---

## Kapitel 12. Integration mit WebSphere Data Interchange

In diesem Kapitel wird die Integration von WebSphere Partner Gateway mit WebSphere Data Interchange beschrieben.

**Anmerkung:** Eine Beschreibung des allgemeinen Verfahrens zur Integration von WebSphere Partner Gateway mit einem Back-End-System finden Sie in Kapitel 2, „Back-End-Integration planen“, auf Seite 9.

---

### Zielgruppe dieses Kapitels

WebSphere Partner Gateway verfügt jetzt über Funktionalität zum Verarbeiten von EDI (zum Beispiel Entfernen von Umschlägen und Konvertierung) ähnlich der Funktionalität in WebSphere Data Interchange. Falls Sie die Möglichkeit haben, diese EDI-Funktionalität an Stelle von WebSphere Data Interchange zu verwenden, benötigen Sie die Informationen in diesem Kapitel nicht. Informationen zum Konfigurieren des Hubs für die Verarbeitung von EDI-Dokumenten finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

Die Informationen in diesem Kapitel sind daher für folgende Personen bestimmt:

- Personen, die eine frühere Version von WebSphere Partner Gateway (bisher unter der Bezeichnung WebSphere Business Integration Connect) einsetzen und auch weiterhin eine Integration mit WebSphere Data Interchange wünschen.
- Personen, die die erweiterte Funktionalität von WebSphere Data Interchange benötigen, die die EDI-Unterstützung von WebSphere Partner Gateway nicht enthält.

Wenn Sie mit WDI arbeiten wollen, müssen Sie die im vorliegenden Kapitel aufgeführten Arbeitsschritte ausführen. Beachten Sie, dass sich in WebSphere Partner Gateway Version 6.0 die Schritte zur EDI-Konfiguration geändert haben. Sie wählen jetzt einen speziellen Dokumentenfluss für die Version von EDI aus, die Sie verwenden. Wenn Sie zum Beispiel EDI-X12-Dokumente austauschen, wählen Sie für den Dokumentenfluss 'ISA' (und nicht 'Alle') aus.

---

### Für dieses Kapitel zu verwendende Ressourcen

Dieses Kapitel enthält eine Erläuterung des Dokumentaustauschprozesses und listet die Schritte zur Einrichtung einer Beispielumgebung für einen solchen Austausch auf. Das in diesem Kapitel verwendete Szenario ist dem Szenario im Lernprogramm *Integrating WebSphere Data Interchange V3.2 with WebSphere Business Integration Connect V4.2* ähnlich, das auf der folgenden Website zur Verfügung gestellt wird:

[www.ibm.com/developerworks/websphere/](http://www.ibm.com/developerworks/websphere/)

Das Lernprogramm stellt zusätzliche Scripts (im Abschnitt über die Konfiguration von WebSphere MQ) sowie Muster für Transformationszuordnungen zur Verfügung. Durch Ausführen der im Lernprogramm behandelten Schritte können Sie die in diesem Kapitel beschriebene Umgebung einrichten.

**Anmerkung:** Im Lernprogramm wird die Integration mit WebSphere Business Integration Connect Version 4.2 beschrieben. Mit kleineren Änderungen gelten diese Schritte aber auch für WebSphere Partner Gateway.

Es wird vorausgesetzt, dass Sie mit der Verwendung von WebSphere Data Interchange vertraut sind. Während Sie dieses Kapitel lesen, ist es sinnvoll, immer wieder auf die Dokumentation zu WebSphere Data Interchange zurückzugreifen, um zusätzliche Informationen zu erhalten.

---

## Einführung

WebSphere Data Interchange ermöglicht die Integration des elektronischen Datenaustauschs (Electronic Data Interchange - EDI) in die WebSphere-Funktionen für Geschäftsprozesse und Nachrichtenübertragung sowie die internetbasierte Business-to-Business-Funktionalität (B2B). Dokumente und Nachrichten werden zwischen WebSphere Partner Gateway und WebSphere Data Interchange über das JMS-Transportprotokoll ausgetauscht. Sie müssen den Pakettyp 'None' angeben, wenn Sie ein Dokument an WebSphere Data Interchange senden.

**Anmerkung:** WebSphere Data Interchange stellt andere Typen von Integrationsoptionen zur Verfügung, wie zum Beispiel eine dateibasierte Integration. Detaillierte Informationen zur Aktivierung des Dokumentaustauschs durch dateibasierte Integration finden Sie in der Dokumentation zu WebSphere Data Interchange.

## Dokumentenversand an WebSphere Data Interchange definieren

Beim Senden eines EDI-Dokuments von WebSphere Partner Gateway an WebSphere Data Interchange führt das System die folgenden Operationen aus:

1. Ein externer Partner sendet ein EDI-Dokument an WebSphere Partner Gateway. Das Dokument wird in einem speziellen Paket über ein Transportprotokoll gesendet. (Im vorliegenden Beispiel wird ein AS2-Paket via HTTP übertragen.) WebSphere Partner Gateway entfernt das AS2-Paket vom EDI-Dokument.
2. WebSphere Partner Gateway stellt das EDI-Dokument in eine Warteschlange.

**Anmerkung:** WebSphere Partner Gateway bestimmt das im Dokument verwendete Protokoll durch Untersuchen der ersten drei Zeichen des EDI-Dokuments. Anschließend werden die Absender- und Empfängerdaten aus dem Protokolltyp ermittelt. Einzelheiten hierzu finden Sie im Abschnitt „Übersicht über das EDI-Routing“ auf Seite 255.

3. WebSphere Data Interchange liest das EDI-Dokument aus der Warteschlange. Es führt die Aufgaben der Paketentfernung, Prüfung und Übersetzung des EDI-Dokuments aus.

**Anmerkung:** WebSphere Data Interchange muss mit den erforderlichen Zuordnungen, Handelspartnerprofilen und anderen Informationen konfiguriert werden. Detaillierte Informationen hierzu enthält die Dokumentation zu WebSphere Data Interchange.

4. WebSphere Data Interchange gibt das Dokument an ein Back-End-System weiter. Wenn WebSphere InterChange Server das Back-End-System ist, sendet WebSphere Data Interchange das Dokument an WebSphere Business Integration Adapter für MQ, um ein Geschäftsobjekt zu erstellen und eine Collaboration innerhalb von InterChange Server aufzurufen.

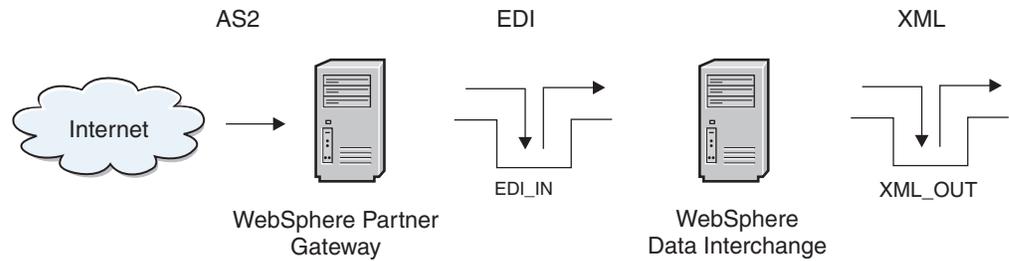


Abbildung 40. EDI-Dokument von WebSphere Partner Gateway

In Abb. 40 sendet ein externer Partner ein EDI-Dokument in einem AS-Paket an WebSphere Partner Gateway, wo es wiederum an die Warteschlange EDI\_IN auf der WebSphere Data Interchange-Seite gesendet wird. Beachten Sie, dass die ferne Warteschlange, die Übertragungswarteschlange, die Empfängerwarteschlange (in diesem Beispiel EDI\_IN) und die Absender- und Empfängerkanäle konfiguriert sein müssen, so dass die an WebSphere Partner Gateway gesendete Nachricht an die Warteschlange EDI\_IN übertragen wird. Der WebSphere Data Interchange-Server nimmt das EDI-Dokument auf, sucht nach den Benutzerprofilen, Zuordnungen usw., konvertiert das Dokument in XML und stellt es in die Warteschlange XML\_OUT.

## Dokumentenempfang von WebSphere Data Interchange definieren

Das Empfangen eines EDI-Dokuments von WebSphere Data Interchange durch WebSphere Partner Gateway erfolgt in folgenden Schritten:

1. WebSphere Data Interchange stellt das EDI-Dokument in eine Warteschlange.
2. WebSphere Partner Gateway liest die Nachricht aus der Warteschlange.

**Anmerkung:** WebSphere Partner Gateway bestimmt, wie das Dokument weiterzuleiten ist. Weitere Informationen hierzu enthält der Abschnitt „Übersicht über das EDI-Routing“ auf Seite 255.

3. WebSphere Partner Gateway leitet das Dokument an den entsprechenden externen Partner weiter.

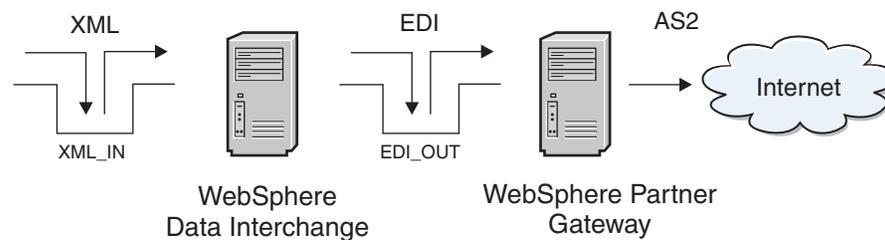


Abbildung 41. Senden eines EDI-Dokuments an WebSphere Partner Gateway

In Abb. 41 wird ein XML-Dokument in die Warteschlange XML\_IN zur Übersetzung durch WebSphere Data Interchange gestellt. Es wird angenommen, dass die Benutzerprofile, Zuordnungen usw. bereits ausgeführt wurden. Bei Empfang eines gültigen XML-Dokuments konvertiert WebSphere Data Interchange dieses in das EDI-Format und stellt die Ausgabe in die Warteschlange EDI\_OUT (eine ferne Warteschlange).

Es wird außerdem angenommen, dass die Übertragungswarteschlange, Absender- und Empfängerkanäle und die Empfängerwarteschlange auf der WebSphere Partner Gateway-Seite eingerichtet sind. Bei Empfang des Dokuments leitet WebSphere Partner Gateway es an den externen Partner weiter.

## Beispielszenario für dieses Kapitel

In diesem Kapitel werden die Schritte zur Einrichtung des Austauschs von EDI-Dokumenten zwischen zwei Handelspartnern beschrieben. Die EDI-Dokumente werden über Internet gesendet, wobei AS2 (über HTTP) als Kommunikationsprotokoll eingesetzt wird.

In diesem Beispiel sind die Handelspartner Partner 1 und Partner 2. Abb. 42 veranschaulicht die Konfigurationen der beiden Partner.

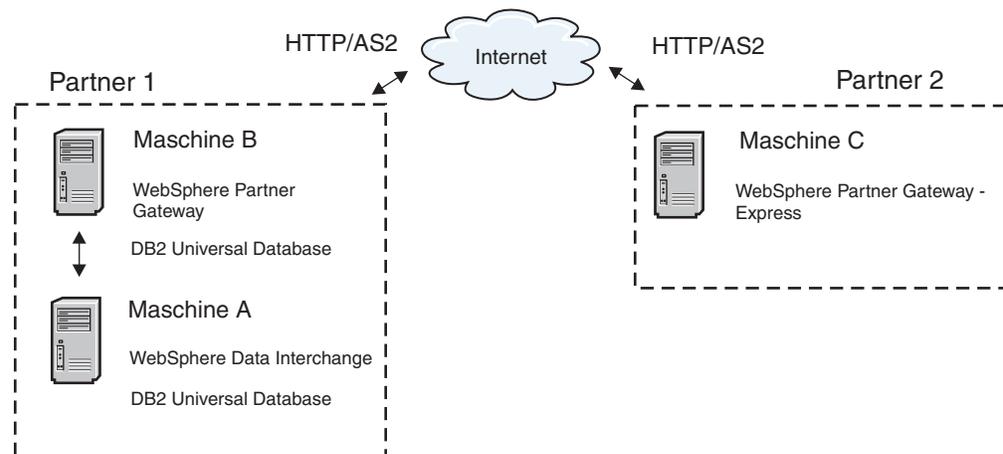


Abbildung 42. Konfiguration zweier Partner im Beispielszenario

Auf den drei Computern ist die folgende Software installiert:

- Computer A enthält WebSphere Data Interchange Server 3.2 und WebSphere Data Interchange Client 3.2 sowie die vorausgesetzten Softwareprodukte.
- Computer B enthält WebSphere Partner Gateway Enterprise Edition sowie die vorausgesetzten Softwareprodukte.
- Computer C enthält WebSphere Partner Gateway - Express.

Eine vollständige Liste der Softwarevoraussetzungen finden Sie im Handbuch *Installation* für WebSphere Partner Gateway und in der Dokumentation zu WebSphere Data Interchange.

In diesem Beispiel betreibt Partner 1 zwei Computer. Auf Computer A sind sowohl WebSphere MQ als auch WebSphere Data Interchange Server installiert. Auf Computer B sind WebSphere MQ sowie WebSphere Partner Gateway Enterprise Edition installiert. Computer B unterstützt die Kommunikation zwischen den beiden Handelspartnern.

WebSphere Data Interchange unterstützt die Integration mit WebSphere MQ, die eine Interoperabilität mit einer breiten Auswahl an Unternehmensanwendungen und Geschäftsprozesssteuerkomponenten ermöglicht. WebSphere Partner Gateway arbeitet mit WebSphere MQ als JMS-Provider. Die Integration zwischen WebSphere Data Interchange und WebSphere Partner Gateway als solche erfolgt über MQ-Nachrichten, die an JMS-API-Clients adressiert sind.

WebSphere Partner Gateway dient zur Kommunikation von EDI-Transaktionen über das Internet unter Verwendung des AS2-Protokolls.

Beachten Sie, dass in diesem Beispiel Partner 2 das Produkt WebSphere Partner Gateway - Express zum Empfangen von Transaktionen über AS2 verwendet und eine eigene WebSphere Data Interchange-Umgebung zur Ausführung von Umsetzungen und Bestätigungen hat.

In diesem Kapitel werden die Details zur Konfiguration der in diesem Beispielszenario verwendeten Computer erläutert. Der Fluss der Nachrichten erfolgt bidirektional, so dass Artefakte zum Senden und zum Empfangen behandelt werden.

---

## Integration mit WebSphere Data Interchange planen

### Von WebSphere Partner Gateway unterstützte WebSphere Data Interchange-Versionen

Die unterstützte Version von WebSphere Data Interchange ist Version 3.2 mit CSD 22. Für die Schnittstelle mit WebSphere Data Interchange wird WebSphere MQ verwendet. Die unterstützte Version von WebSphere MQ ist Version 6.0 mit dem neuesten Fixpack.

---

## Umgebung für den Nachrichtenaustausch konfigurieren

Zur Herstellung der Kommunikation zwischen WebSphere Data Interchange und WebSphere Partner Gateway führen Sie die folgenden Einrichtungs- und Konfigurationsschritte aus:

- „WebSphere MQ-Kommunikation konfigurieren“
- „WebSphere Data Interchange konfigurieren“ auf Seite 217
- „JMS-Umgebung einrichten“ auf Seite 222
- „WebSphere Partner Gateway Enterprise Edition konfigurieren“ auf Seite 223

### WebSphere MQ-Kommunikation konfigurieren

Der erste Schritt bei der Einrichtung der Umgebung besteht in der Konfiguration der übergreifenden WebSphere MQ-Kommunikation. Mit übergreifender Kommunikation wird das Senden von Nachrichten von einem Warteschlangenmanager zu einem anderen bezeichnet. Der erste Schritt ist das Definieren eines Warteschlangenmanagers (und zugehöriger Objekte) für das WebSphere Data Interchange-System und das WebSphere Partner Gateway-System. Wenn Sie beabsichtigen, Nachrichten in beide Richtungen zu senden, richten Sie auf beiden Systemen einen Quellenwarteschlangenmanager und einen Empfängerwarteschlangenmanager ein. Im Quellenwarteschlangenmanager definieren Sie einen Absenderkanal, eine Definition für ferne Warteschlange und eine Übertragungwarteschlange. Im Empfängerwarteschlangenmanager definieren Sie einen Empfängerkanal und eine Empfängerwarteschlange.

**Anmerkung:** Weitere Einzelheiten zur Definition von Warteschlangenmanagern finden Sie in der Dokumentation zu WebSphere MQ.

Dieser Abschnitt zeigt die Werte, die Sie zur Einrichtung der für das Beispielszenario erforderlichen Warteschlangenmanager und der zugeordneten Objekte verwenden würden. In diesem Szenario ist WebSphere MQ Version 5.3 auf beiden Computern (A und B) installiert. Der erste Schritt besteht anschließend in der

Erstellung eines Warteschlangenmanagers auf Computer A und Computer B zur Verwendung durch WebSphere Data Interchange bzw. WebSphere Partner Gateway Enterprise Edition.

**Anmerkung:** Ihr Warteschlangenmanager für WebSphere Data Interchange muss so konfiguriert werden, dass er den WebSphere Data Interchange Server mit Hilfe der Anwendung WDI Adapter auslöst.

- Auf Computer A würden Sie den für die Verwendung mit WebSphere Data Interchange definierten Warteschlangenmanager verwenden. In den verbleibenden Abschnitten dieses Kapitels wird dieser Warteschlangenmanager als WDI32\_QM bezeichnet.
- Auf Computer B würden Sie den Warteschlangenmanager verwenden, der bei der Installation und Konfiguration von WebSphere Partner Gateway Enterprise Edition zu Anfang erstellt wurde. In den verbleibenden Abschnitten dieses Kapitels wird dieser Warteschlangenmanager als HUB\_QM bezeichnet.

Zum Senden von Nachrichten von einem Warteschlangenmanager zu einem anderen unter Verwendung von WebSphere MQ definieren Sie die folgenden Objekte:

- Im Quellenwarteschlangenmanager:
  - Absenderkanal
  - Definition für ferne Warteschlange
  - Übertragungswarteschlange
- Im Empfängerwarteschlangenmanager:
  - Empfängerkanal
  - Empfängerwarteschlange

Im Beispielszenario fungieren sowohl Computer A als auch Computer B als Absender und Empfänger. Daher müssten Sie eine Reihe von Objekten auf beiden Computern definieren.

In Tabelle 84 sind die Objekte aufgelistet, die Sie erstellen würden, um Computer A und Computer B als Absender und Empfänger einzurichten.

*Tabelle 84. Zu erstellende WebSphere MQ-Objekte*

WebSphere MQ-Objekt	Computer A	Computer B
Warteschlangenmanager	WDI32_QM	HUB_QM
Absenderkanal	TO.HUB60	TO.WDI32
Empfängerkanal	TO.WDI32	TO.HUB60
Ferne Warteschlange	EDI_OUT_A	EDI_OUT_B
Übertragungswarteschlange	XMITQ_A	XMITQ_B
Lokale Warteschlange	EDI_IN_A	EDI_IN_B
Lokale Warteschlange	XML_IN_A	XML_IN_B
Lokale Warteschlange	XML_OUT_A	XML_OUT_B

Abb. 43 zeigt den Nachrichtenfluss zwischen Computer A und Computer B sowie die Rolle der in Tabelle 84 aufgelisteten WebSphere MQ-Objekte.

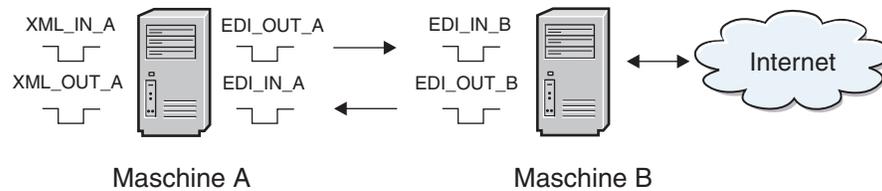


Abbildung 43. Nachrichtenfluss zwischen Computer A und Computer B

Abhängig von der verwendeten WebSphere MQ-Plattform könnten Sie verschiedene Methoden zum Definieren dieser Objekte verwenden. Zum Beispiel könnten Sie unter Windows die Objekte mit WebSphere MQ Explorer definieren.

## WebSphere Data Interchange konfigurieren

Damit WebSphere Data Interchange Nachrichten aus der WebSphere MQ-Warteschlange empfangen und EDI-Nachrichten in eine Warteschlange schreiben kann, müssen Sie Profile im WebSphere Data Interchange Client konfigurieren. Bei Verwendung von WebSphere Data Interchange Client würden Sie die folgenden Profile erstellen, die in den nachfolgenden Abschnitten beschrieben werden:

- MQ Series-Warteschlangenprofil
- Netzprofil
- Mailboxprofil
- Serviceprofil

Im Beispielszenario empfängt WebSphere Data Interchange XML-Nachrichten aus der WebSphere MQ-Warteschlange XML\_IN\_A und schreibt das Ergebnis der Übersetzung in die WebSphere MQ-Warteschlange EDI\_OUT\_A. Dieser Vorgang wird als XML-in-EDI-Übersetzung bezeichnet. WebSphere Data Interchange empfängt EDI-Dokumente aus WebSphere Partner Gateway Enterprise Edition in der WebSphere MQ-Warteschlange EDI\_IN\_A und schreibt das Ergebnis der Übersetzung in XML\_OUT\_A.

### MQSeries<sup>(R)</sup>-Warteschlangenprofil

Ein MQSeries-Warteschlangenprofil (MQSeries Queue Profile) enthält Informationen über eine WebSphere MQ-Nachrichtenwarteschlange. Tabelle 85 zeigt die Eigenschaften zur Konfiguration der einzelnen Profile.

Tabelle 85. Eigenschaften in einem MQSeries-Warteschlangenprofil

MQ-Eigenschaft	Beschreibung
Queue Profile ID (Warteschlangenprofil-ID)	Die eindeutige Kennung zur Benennung des Profils (logischer Name).
Full Queue Name (Vollständiger Warteschlangenname)	Der tatsächliche Name der WebSphere MQ-Warteschlange.
Queue Manager Name (Warteschlangenmanagername)	Der tatsächliche Name des WebSphere MQ-Warteschlangenmanagers.
Description (Beschreibung)	Eine beliebige Zeichenfolge zur Beschreibung des Zwecks des Profils.
Maximum Length (Maximale Länge)	Die größte zulässige Nachricht für die Warteschlange, entsprechend der Definition in WebSphere MQ.

Tabelle 85. Eigenschaften in einem MQSeries-Warteschlangenprofil (Forts.)

MQ-Eigenschaft	Beschreibung
Destructive Reads (Löschende Lesevorgänge)	Falls ausgewählt, veranlassen diese Operationen WebSphere Data Interchange die Nachrichten beim Lesen aus der WebSphere MQ-Warteschlange zu entfernen.
Syncpoint Control (Synchronisationspunktsteuerung)	Wenn ausgewählt, erfolgt das Lesen und Schreiben von Warteschlangennachrichten unter Steuerung eines Synchronisationspunkts. Wenn die Steuerung durch Synchronisationspunkt aktiviert ist, finden Änderungen an einer Nachrichtenwarteschlange erst statt, wenn WebSphere Data Interchange einen Synchronisationspunkt ausgibt.

Da Sie mit den WebSphere MQ-Warteschlangen arbeiten, benötigen Sie für *jede* Warteschlange ein MQSeries-Warteschlangenprofil in WebSphere Data Interchange. Insgesamt würden Sie vier MQSeries-Warteschlangenprofile erstellen, also eines für jede WebSphere MQ-Warteschlange, die im Nachrichtenfluss verwendet wird. Über den Konfigurationsbereich von WebSphere Data Interchange Client würden Sie folgende Aktionen ausführen:

1. Erstellen Sie ein MQSeries-Warteschlangenprofil für XML\_IN\_A und EDI\_OU\_A.

In Tabelle 86 sind die tatsächlichen Parameter aufgeführt, die in jedem von Ihnen erstellten MQSeries-Warteschlangenprofil angegeben sind. Die hier dargestellten Warteschlangen werden bei der XML-in-EDI-Übersetzung verwendet.

Tabelle 86. MQSeries-Warteschlangenprofil für XML\_IN\_A und EDI\_OU\_A

Eigenschaft der Warteschlange	Wert für XML_IN_A	Wert für EDI_OU_A
Queue Profile ID (Warteschlangenprofil-ID)	XML_IN_A	EDI_OU_A
Full Queue Name (Vollständiger Warteschlangenname)	XML_IN_A	EDI_OUT_A
Queue Manager Name (Warteschlangenmanagername)	WDI32_QM	WDI32_QM
Destructive Reads (Löschende Lesevorgänge)	Ausgewählt	Ausgewählt
Syncpoint Control (Synchronisationspunktsteuerung)	Ausgewählt	Ausgewählt

**Anmerkung:** Die Warteschlangenprofil-ID (Queue Profile ID) ist auf maximal acht Zeichen beschränkt. Daher muss die Profil-ID für die Warteschlange EDI\_OUT\_A den Namen EDI\_OU\_A erhalten. Alle Verweise auf die WebSphere MQ-Warteschlange EDI\_OUT\_A in WebSphere Data Interchange verwenden EDI\_OU\_A.

2. Erstellen Sie ein MQSeries-Warteschlangenprofil für EDI\_IN\_A und XML\_OU\_A. Tabelle 87 zeigt die Eigenschaften für jede einzelne Warteschlange, die bei der EDI-in-XML-Übersetzung verwendet wird.

Tabelle 87. MQSeries-Warteschlangenprofil für EDI\_IN\_A und XML\_OU\_A

Eigenschaft der Warteschlange	Wert für EDI_IN_A	Wert für XML_OU_A
Queue Profile ID (Warteschlangenprofil-ID)	EDI_IN_A	XML_OU_A
Full Queue Name (Vollständiger Warteschlangenname)	EDI_IN_A	XML_OUT_A
Queue Manager Name (Warteschlangenmanagername)	WDI32_QM	WDI32_QM
Destructive Reads (Löschende Lesevorgänge)	Ausgewählt	Ausgewählt
Syncpoint Control (Synchronisationspunktsteuerung)	Ausgewählt	Ausgewählt

## Netzprofil

Für WebSphere Data Interchange definieren Netzprofile die Merkmale von Netzen, die Sie zur Kommunikation mit Handelspartnern verwenden. Für dieses Szenario würden Sie ein Netzprofil (Network Profile) erstellen und konfigurieren, das mit den zuvor erstellten WebSphere MQ-Warteschlangen kommuniziert.

Tabelle 88 zeigt die Eigenschaften zur Konfiguration des Netzprofils.

Tabelle 88. Eigenschaften in einem Netzprofil

Netzeigenschaft	Beschreibung
Network ID (Netz-ID)	Eine eindeutige Kennung zur Benennung des Profils
Communication Routine (Kommunikationsroutine)	Der Name des Programms, das Netzbefehle erstellt und das Netzprogramm zur Verarbeitung der Befehle aufruft
Network Program (Netzprogramm)	Das Programm, das von der Kommunikationsroutine zur Verarbeitung von Anforderungen aufgerufen wird
Network Parameters (Netzparameter)	Für das Netzprogramm erforderliche Parameter

Für dieses Szenario erstellen und konfigurieren Sie wie folgt ein Netzprofil, das mit den zuvor erstellten WebSphere MQ-Warteschlangen kommuniziert (siehe „MQSeries<sup>(R)</sup>-Warteschlangenprofil“ auf Seite 217):

1. Erstellen Sie ein neues Netzprofil mit dem Namen HUB\_IN.

Dieses Netzprofil wird im XML-in-EDI-Szenario verwendet. In Tabelle 89 sind die tatsächlichen, für HUB\_IN angegebenen Parameter aufgeführt.

Tabelle 89. Netzprofil für HUB\_IN

Netzeigenschaft	Wert für das Profil HUB_IN
Network ID (Netz-ID)	HUB_IN
Communication Routine (Kommunikationsroutine)	VANIMQ
Network Program (Netzprogramm)	EDIMQSR
Network Parameters (Netzparameter)	SENDMQ=EDI_OU_A RECEIVMQ=XML_IN_A

- Erstellen Sie ein zweites Netzprofil mit dem Namen HUB\_OUT.

Dieses Netzprofil wird in der Übersetzung von EDI-Dokumenten verwendet, die aus WebSphere Partner Gateway Enterprise Edition empfangen werden. Ein zweites Netzprofil ist erforderlich, weil WebSphere Partner Gateway Enterprise Edition Nachrichten in die WebSphere MQ-Warteschlangen stellt, die RFH2-Header enthalten. In Tabelle 90 sind die Eigenschaften von HUB\_OUT aufgeführt.

Tabelle 90. Netzprofil für HUB\_OUT

Netzeigenschaft	Wert für das Profil HUB_OUT
Network ID (Netz-ID)	HUB_OUT
Communication Routine (Kommunikationsroutine)	VANIMQ
Network Program (Netzprogramm)	EDIRFH2
Network Parameters (Netzparameter)	SENDMQ=XML_OU_A RECEIVEMQ=EDI_IN_A

## Mailboxprofil

Mailboxprofile enthalten die Informationen, die von WebSphere Data Interchange zur Identifizierung von Einzelpersonen und Gruppen in Ihrer Organisation benötigt werden, die zu übersetzende Dokumente empfangen. Tabelle 91 zeigt die für jedes Mailboxprofil zu konfigurierenden Eigenschaften.

Tabelle 91. Eigenschaften in einem Mailboxprofil

Mailboxeigenschaft	Beschreibung
Mailbox ID (Mailbox-ID)	Eine eindeutige Kennung zur Benennung des Profils
Network ID (Netz-ID)	Die Netz-ID des zuvor erstellten Netzprofils

Sie erstellen Mailboxprofile wie folgt für jede einzelne WebSphere MQ-Warteschlange, um die Einzelpersonen und Gruppen in der Organisation anzugeben:

- Erstellen Sie ein Mailboxprofil für jede verwendete WebSphere MQ-Warteschlange.

In Tabelle 92 sind die tatsächlichen Parameter in den einzelnen Mailboxprofilen aufgeführt.

Tabelle 92. Mailboxprofile für XML\_IN\_A und EDI\_OU\_A

Mailboxeigenschaft	Wert für XML_IN_A	Wert für EDI_OU_A
Mailbox ID (Mailbox-ID)	XML_IN_A	EDI_OU_A
Network ID (Netz-ID)	HUB_IN	HUB_IN
Receive File (Empfangsdatei)	XML_IN_A	EDI_OU_A

- Erstellen Sie ein zweites Paar Mailboxen.

Tabelle 93 listet die Eigenschaften für jede Mailbox auf.

Tabelle 93. Mailboxprofile für EDI\_IN\_A und XML\_OU\_A

Mailboxeigenschaft	Wert für EDI_IN_A	Wert für XML_OU_A
Mailbox ID (Mailbox-ID)	EDI_IN_A	XML_OU_A

Tabelle 93. Mailboxprofile für EDI\_IN\_A und XML\_OU\_A (Forts.)

Mailboxeigenschaft	Wert für EDI_IN_A	Wert für XML_OU_A
Network ID (Netz-ID)	HUB_OUT	HUB_OUT
Receive File (Empfangsdatei)	EDI_IN_A	XML_OU_A

## Serviceprofil

Serviceprofile geben Ihnen die Möglichkeit, einen Dienstprogrammbefehl einzugeben und alle Dateien zu definieren, die bei der Ausführung dieses Befehls verwendet werden.

Für das Beispielszenario führen Sie die folgenden Schritte aus:

1. Erstellen Sie ein neues Serviceprofil (Service Profile) für XML\_IN\_A. Sie definieren die Eigenschaften auf der Registerkarte **General** ('Allgemein') wie folgt:

- Befehlsverkettung fortsetzen: **Bei Erfolg**
- PERFORM-Befehl:
 

```
PERFORM TRANSFORM WHERE INFILE(XML_IN_A) SYNTAX(X)
OUTTYPE(MQ)OUTFILE(EDI_OU_A)
```

In Tabelle 94 sind die Eigenschaften für die allgemeinen Dateien ('Common Files') aufgeführt.

Tabelle 94. Allgemeine Dateien für XML\_IN\_A

Eigenschaft 'Common File'	Wert
Tracking File (Überwachungsdatei)	..\trk\xml_in.trk
Exception File (Ausnahmendatei)	..\xex\xml_in.xex
Work File (Arbeitsdatei)	..\wrk\xml_in.wrk
Report File (Berichtsdatei)	..\rpt\xml_in.rpt
Query File (Abfragedatei)	..\qry\xml_in.qry

2. Geben Sie folgende Details auf der Registerkarte **Output Files** ('Ausgabedateien') ein:

- Name im Befehl: **EDI\_OU\_A**
- Systemdateiname: **..\edi\edi\_out.txt**

**Anmerkung:** Wegen der Zeichenlängenbeschränkungen wird EDI\_OU\_A an Stelle von EDI\_OUT\_A verwendet.

3. Erstellen Sie ein zweites Serviceprofil für EDI\_IN\_A. Sie definieren die Eigenschaften auf der Registerkarte **General** ('Allgemein') wie folgt:

- Befehlsverkettung fortsetzen: **Bei Erfolg**
- PERFORM-Befehl:
 

```
PERFORM TRANSFORM WHERE INFILE(EDI_IN_A) SYNTAX(E)
OUTTYPE(MQ) OUTFILE(XML_OU_A)
```

In Tabelle 95 sind die Eigenschaften für die allgemeinen Dateien ('Common Files') aufgeführt.

Tabelle 95. Allgemeine Dateien für EDI\_IN\_A

Eigenschaft 'Common File'	Wert
Tracking File (Überwachungsdatei)	..\trk\edi_in.trk
Exception File (Ausnahmendatei)	..\xex\edi_in.xex
Work File (Arbeitsdatei)	..\wrk\edi_in.wrk

Tabelle 95. Allgemeine Dateien für EDI\_IN\_A (Forts.)

Eigenschaft 'Common File'	Wert
Report File (Berichtsdatei)	..\rpt\edi_in.rpt
Query File (Abfragedatei)	..\qry\edi_in.qry

4. Geben Sie folgende Details auf der Registerkarte **Output Files** ('Ausgabedateien') ein:

- Name im Befehl: **XML\_OU\_A**
- Systemdateiname: **..\xml\xml\_out.txt**

**Anmerkung:** Wegen der Zeichenlängenbeschränkungen wird XML\_OU\_A an Stelle von XML\_OUT\_A verwendet. Diese Beschränkung wurde mit CSD10 für WebSphere Interchange Server beseitigt.

### Datentransformationszuordnungen importieren und kompilieren

Nach der im vorigen Abschnitt beschriebenen Erstellung der Profile können Sie alle Zuordnungen importieren, die Sie zur Umwandlung Ihrer Daten benötigen. Anschließend kompilieren Sie die Transformationszuordnungen und definieren für jede Zuordnung eine Regel. Diese Aufgaben werden über den WebSphere Data Interchange Client ausgeführt. Informationen dazu finden Sie in der Dokumentation zu WebSphere Data Interchange.

## JMS-Umgebung einrichten

Wie zuvor in diesem Kapitel erwähnt, kann WebSphere Partner Gateway Enterprise Edition die WebSphere MQ-Implementierung des Java Message Service (JMS) zur Integration mit WebSphere Data Interchange nutzen.

**Anmerkung:** Alternativ ist es möglich, LDAP oder WebSphere Application Server als JNDI-Provider zu verwenden.

In diesem zweiten Abschnitt werden die Schritte skizziert, die zur Erstellung einer JMS-Umgebung auf Computer B erforderlich sind:

- „JMSAdmin konfigurieren“
- „JMS-Objekte erstellen“ auf Seite 223

WebSphere MQ-Klassen für Java und WebSphere MQ-Klassen für JMS sind in WebSphere MQ für Windows Version 5.3 integriert.

### JMSAdmin konfigurieren

Verwenden Sie das Tool JMSAdmin, das in WebSphere MQ zur Verfügung steht, um die JMS-Objekte im JNDI-Format zu erstellen. Informationen zur Erstellung der Standardkonfigurationsdatei JMSAdmin.config finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

Zur Erstellung der JMS-Objekte für diesen Lerntext:

1. Zur Verwendung eines dateibasierten JNDI-Providers würden Sie sicherstellen, dass die Datei 'JMSAdmin.config' die nachfolgenden Zeilen enthielte:

```
INITIAL_CONTEXT_FACTORY=com.sun.jndi.fscontext.RefFSContextFactory
PROVIDER_URL=file:/opt/mqm/java/JNDI
```

2. Wenn das Verzeichnis JNDI nicht bereits vorhanden ist, erstellen Sie das Verzeichnis JNDI unterhalb des folgenden Verzeichnisses:

```
/opt/mqm/java/bin
```

Vor dem Aufrufen des Tools JMSAdmin würden Sie sicherstellen, dass die Variable CLASSPATH die folgenden Einträge enthielte:

```
/opt/mqm/java/lib/jms.jar  
/opt/mqm/java/lib/com.ibm.mq.jar  
/opt/mqm/java/lib/com.ibm.mqjms.jar  
/opt/mqm/java/lib/jta.jar  
/opt/mqm/java/lib/connector.jar  
/opt/mqm/java/lib/jndi.jar  
/opt/mqm/java/lib/providerutil.jar  
/opt/mqm/java/lib/fscontext.jar
```

**Anmerkung:** Die obigen Einträge, die sich auf Linux<sup>(TM)</sup> beziehen, setzen voraus, dass Sie mit einer dateibasierten JNDI-Komponente arbeiten.

### JMS-Objekte erstellen

Zur Erstellung der erforderlichen JMS-Objekte verwenden Sie das Tool JMSAdmin. Für das Beispielszenario würden Sie wie folgt vorgehen:

1. Definieren Sie einen neuen Kontext:  
DEF CTX(WdiJms)
2. Wechseln Sie zu diesem neuen Kontext:  
CHG CTX(WdiJms)
3. Definieren Sie eine Warteschlangenverbindungsfactory:  
DEF QCF(HUB60\_QM\_QCF) TRAN(CLIENT) HOST(IP\_COMPUTER\_B)  
PORT(9999) CHAN(java.channel) QMANAGER(HUB60\_QM)
4. Definieren Sie die Warteschlange EDI\_IN\_B:  
DEF Q(EDI\_IN\_B) QMANAGER(HUB60\_QM) QUEUE(EDI\_IN\_B)
5. Definieren Sie die Warteschlange EDI\_OUT\_B:  
DEF Q(EDI\_OUT\_B) QMANAGER(HUB60\_QM) QUEUE(EDI\_OUT\_B)
6. Beenden Sie die JMSAdmin-Sitzung:  
END

## WebSphere Partner Gateway Enterprise Edition konfigurieren

WebSphere Partner Gateway ist die Übertragungsschicht zwischen unterschiedlichen und voneinander unabhängigen externen Partnern und internen Prozessen. Wenn Sie WebSphere Partner Gateway zur Arbeit mit EDI-Dokumenten einrichten, können Sie das Produkt zu folgenden Zwecken konfigurieren:

- Senden und Empfangen von EDI-Dokumenten an und von WebSphere Data Interchange
- Austauschen von EDI-Transaktionen mit externen Handelspartnern über AS2

Im Handbuch *WebSphere Partner Gateway Hubkonfiguration* finden Sie umfassende Informationen zur Konfiguration von WebSphere Partner Gateway Enterprise Edition und Advanced Edition. Dieser Abschnitt enthält ein Beispiel für die Konfiguration von WebSphere Partner Gateway Enterprise Edition, das im Beispielszenario beschrieben ist. Die folgenden Schritte werden beschrieben:

1. „Partner erstellen“ auf Seite 224
2. „B2B-Funktionalität einstellen“ auf Seite 225
3. „Ziele erstellen“ auf Seite 227
4. „Interaktion definieren“ auf Seite 228
5. „Partnerverbindungen erstellen“ auf Seite 229
6. „Empfänger erstellen“ auf Seite 230

**Anmerkung:** Informationen zur Konfiguration von WebSphere Partner Gateway - Express finden Sie im Abschnitt „WebSphere Partner Gateway - Express konfigurieren“ auf Seite 231.

## Partner erstellen

Das Partnerprofil identifiziert Unternehmen für das System. Partner für Partner 1 und Partner 2 werden über die Community Console von WebSphere Partner Gateway Enterprise Edition erstellt.

**Partner für Partner 1 erstellen:** Erstellen Sie ein Partnerprofil, um Computer A und Computer B darzustellen, die die beiden Systeme sind, deren Eigner Partner 1 ist.

Zur Erstellung dieses Partnerprofils können Sie folgende Schritte ausführen:

1. Öffnen Sie die Community Console von WebSphere Partner Gateway.
2. Melden Sie sich als **Hubadmin** an.
3. Klicken Sie auf **Kontenadmin > Profile > Partner**.
4. Klicken Sie auf **Erstellen** und geben Sie die Details wie in folgender Tabelle 96 aufgeführt ein.

Tabelle 96. Partnereigenschaften für Partner 1

Feldname	Wert
Anmeldename des Unternehmens	partner1
Anzeigenname des Partners	Partner 1
Partnertyp	Interner Partner
Status	Aktiviert
Lieferantentyp	Andere
Website	http://IP_COMPUTER_A  Dabei ist IP_COMPUTER_A die Internet Protocol-Adresse (IP-Adresse) von Computer A.
Geschäfts-ID-Typ	Unformatiert
Geschäfts-ID-Kennung	123456789
Zieltyp für IP-Adresse	Produktion
IP-Adresse	IP_COMPUTER_A  Dabei ist IP_COMPUTER_A die Internet Protocol-Adresse (IP-Adresse) von Computer A.

**Anmerkung:** Zur Erstellung des Geschäfts-ID-Typs und der Geschäfts-ID-Kennung klicken Sie zuerst auf die Schaltfläche **Neu** unter Geschäfts-ID. Die Geschäfts-ID muss eindeutig sein. Analog klicken Sie auf die Schaltfläche **Neu** unter **IP-Adresse**, um Details in Bezug auf die IP-Adresse zu erstellen.

5. Klicken Sie auf **Speichern**.

WebSphere Partner Gateway Enterprise Edition verwendet die Geschäfts-ID-Kennung (in Tabelle 96 definiert) zur Identifizierung des Absenders oder Empfängers eines Dokuments. Wenn eine ANSI-X12-EDI-Transaktion empfangen wird, werden

die Daten der Interchange-Absender- und Empfängerkomponente (Interchange Sender und Receiver) gelesen, um die Quelle und das Ziel der Transaktion zu ermitteln.

**Wichtig:** Notieren Sie sich das Kennwort des Administrators für Partner 1, da Sie es später benötigen. Wenn Sie sich an der Community Console als Partner 1 anmelden, werden Sie aufgefordert, das Kennwort einzugeben und es anschließend zu ändern.

**Partner für Partner 2 erstellen:** Erstellen Sie als Nächstes einen externen Partner, um den Partner 2 darzustellen. Zur Erstellung des Partners führen Sie die folgenden Schritte aus:

1. Klicken Sie auf **Kontenadmin > Profile > Partner**.
2. Klicken Sie auf **Erstellen**.
3. Geben Sie die in der folgenden Tabelle 97 aufgeführten Werte ein.

*Tabelle 97. Partnereigenschaften für Partner 2*

Feldname	Wert
Anmeldename des Unternehmens	partner2
Anzeigename des Partners	Partner 2
Partnertyp	Partner
Status	Aktiviert
Lieferantentyp	Andere
Website	http://IP_COMPUTER_C  Dabei ist IP_COMPUTER_C die Internet Protocol-Adresse (IP-Adresse) von Computer C.
Geschäfts-ID-Typ	Unformatiert
Geschäfts-ID-Kennung	987654321
Zieltyp für IP-Adresse	Produktion
IP-Adresse	IP_COMPUTER_C  Dabei ist IP_COMPUTER_C die Internet Protocol-Adresse (IP-Adresse) von Computer C.

4. Klicken Sie auf **Speichern**.

**Wichtig:** Notieren Sie sich das Kennwort des Administrators für Partner 2, da Sie es später benötigen. Wenn Sie sich an der Community Console als Partner 2 angemeldet haben, werden Sie aufgefordert, das Kennwort einzugeben und es anschließend zu ändern.

### **B2B-Funktionalität einstellen**

Die B2B-Funktionalität wird für jeden Partner in WebSphere Partner Gateway Enterprise Edition über die Community Console definiert. Nachdem Sie die B2B-Funktionalität für die Partner definiert haben, können Sie eine gültige Dokumentenflussdefinition definieren, die zur Unterstützung bestimmter Geschäfts-Collaboration-Typen zwischen den Partnern verwendet wird.

**B2B-Funktionalität für Partner 1 einstellen:** Zur Definition der B2B-Funktionalität für Partner 1 führen Sie die folgenden Schritte aus:

1. Klicken Sie auf **Kontenadmin > Profile > Partner**.
2. Klicken Sie auf **Suchen**, um eine Liste aller im System definierten Partner anzuzeigen.
3. Klicken Sie auf das Symbol **Details anzeigen** neben **Partner 1** und anschließend auf **B2B-Funktionalität**.

Die B2B-Funktionalität wird durch Klicken auf das Symbol **Rolle ist nicht aktiv** aktiviert. Für den Zweck dieses Beispiels konfigurieren Sie nur die B2B-Funktionalität, die zur Implementierung dieses Szenarios erforderlich ist.

Zur Festlegung des Quellen- und Zielpakettyps 'None' für Partner 1 würden Sie wie folgt vorgehen:

1. Klicken Sie auf das Symbol **Rolle ist nicht aktiv** unter **Quelle festlegen für Paket: Kein**, um die Option zu aktivieren. Wiederholen Sie diesen Schritt für **Ziel festlegen**.
2. Klicken Sie auf das Symbol **Erweitern**, um weitere Informationen anzuzeigen.
3. Klicken Sie auf das Symbol **Rolle ist nicht aktiv** für **Protokoll: EDI-X12 (ALLE)** für Quelle und Ziel.
4. Klicken Sie auf das Symbol **Erweitern**.
5. Klicken Sie auf das Symbol **Rolle ist nicht aktiv** für **Dokumentenfluss: ISA (ALL)** für Quelle und Ziel.

**B2B-Funktionalität für Partner 2 einstellen:** Zur Definition der B2B-Funktionalität für Partner 2 führen Sie die folgenden Schritte aus:

1. Klicken Sie auf **Kontenadmin > Profile > Partner**.
2. Klicken Sie auf **Suchen**, um eine Liste aller im System definierten Partner anzuzeigen.
3. Klicken Sie auf das Symbol **Details anzeigen** neben **Partner 2** und anschließend auf **B2B-Funktionalität**.

Zur Festlegung des Quellen- und Zielpakettyps 'AS' für Partner 2 würden Sie wie folgt vorgehen:

1. Klicken Sie auf das Symbol **Rolle ist nicht aktiv** unter **Quelle festlegen für Paket: AS**. Wiederholen Sie diesen Schritt für **Ziel festlegen**.
2. Klicken Sie auf das Symbol **Erweitern**, um weitere Informationen anzuzeigen.
3. Klicken Sie auf das Symbol **Rolle ist nicht aktiv** für **Protokoll: EDI-X12 (ALLE)** für Quelle und Ziel.
4. Klicken Sie auf das Symbol **Erweitern**.
5. Klicken Sie auf das Symbol **Rolle ist nicht aktiv** für **Dokumentenfluss: ISA (ALL)** für Quelle und Ziel.

Als Nächstes aktualisieren Sie wie folgt die AS-Definition für Partner 2, um sicherzustellen, dass Message Disposition Notifications (MDNs) für AS2, die an Partner 2 gesendet werden, an die richtige Adresse zurückgegeben werden:

1. Klicken Sie auf das Symbol **Bearbeiten**.
2. Geben Sie eine AS-MDN-E-Mail-Adresse an.  
Diese Adresse wird zum Empfangen von MDNs für AS1 verwendet.
3. Geben Sie eine AS-MDN-HTTP-URL-Adresse ein:  
`http://IP_COMPUTER_B:PORT/bcgreceiver/submit`

**Anmerkung:** Die für AS2 definierte URL-Adresse verwendet die gleichen Parameter, die später in diesem Kapitel für den AS2-Empfänger definiert werden.

## Ziele erstellen

Ein Ziel in WebSphere Partner Gateway definiert einen Netzpunkt, der als Eingang in ein anderes Netz fungiert. Das Ziel enthält die Informationen, die WebSphere Partner Gateway mitteilen, wie Dokumente an die Enterprise Application Integration-Schicht (EAI-Schicht) zuzustellen sind.

**Ziel für Partner 1 erstellen:** Partner 2 sendet EDI-Dokumente an Partner 1 über AS2. Das Ziel von Partner 1 wird zum Senden der über AS2 empfangenen EDI-Dokumente an eine JMS-Warteschlange und schließlich an WebSphere Data Interchange zur Übersetzung verwendet.

Zur Erstellung eines neuen Ziels für Partner 1 führen Sie die folgenden Schritte aus:

1. Klicken Sie auf **Kontenadmin > Profile > Partner**.
2. Klicken Sie auf **Suchen**.
3. Wählen Sie Partner 1 aus, indem Sie auf das Symbol **Details anzeigen** klicken und anschließend **Ziele** auswählen.
4. Klicken Sie auf **Erstellen**, um ein neues Ziel für Partner 1 zu erstellen.
5. Geben Sie die Werte für dieses neue Ziel wie in Tabelle 98 gezeigt ein.

*Tabelle 98. Eigenschaften für das Ziel von Partner 1*

Feldname	Wert
Zielname	JMStoPartner1
Transport	JMS
Empfänger-URI	file:///opt/mqm/java/JNDI/WdiJms
JMS-Factory-Name	HUB60_QM_QCF
JMS-Nachrichtenklasse	TextMessage
JMS-Nachrichtentyp	TextMessage
JMS-Warteschlangenname	EDI_OUT_B
JMS-JNDI-Factory-Name	com.sun.jndi.fscontext.ReffFSContextFactory

6. Klicken Sie auf **Speichern**.

Definieren Sie 'JMStoPartner1' wie folgt als Standardziel für Partner 1:

1. Klicken Sie auf **Standardziele anzeigen**.
2. Wählen Sie in der Liste **Produktion** den Eintrag **JMS2toPartner1** aus.
3. Klicken Sie auf **Speichern**.

**Ziel für Partner 2 erstellen:** Partner 1 sendet EDI-Dokumente an WebSphere Partner Gateway Enterprise Edition über eine JMS-Warteschlange. Das Ziel von Partner 2 wird zum Senden der empfangenen EDI-Dokumente an Partner 2 über AS2 verwendet.

Zur Erstellung eines neuen Ziels für Partner 2 führen Sie die folgenden Schritte aus:

1. Klicken Sie auf **Kontenadmin > Profile > Partner**.
2. Klicken Sie auf **Suchen**.

3. Wählen Sie Partner 2 aus, indem Sie auf das Symbol **Details anzeigen** klicken und anschließend **Ziele** auswählen.
4. Klicken Sie auf **Erstellen**, um ein neues Ziel für Partner 2 zu erstellen.
5. Geben Sie die Werte für dieses Ziel wie in Tabelle 99 gezeigt ein.

Tabelle 99. Eigenschaften für das Ziel von Partner 2

Zielname	AS2toPartner2
Transport	HTTP/1.1
Empfänger-URI	http://IP_COMPUTER_C/input/AS2
Benutzername	partner1
Kennwort	partner1

6. Klicken Sie auf **Speichern**.

**Anmerkung:** Der Benutzername und das Kennwort wie oben eingegeben beziehen sich auf die Zuordnungsmethode von Partnern für eingehende Dokumente für HTTP (Inbound Partner Mapping Method) wie in WebSphere Partner Gateway - Express definiert.

Ein Beispiel für die Einstellung dieser Eigenschaften in WebSphere Partner Gateway - Express finden Sie im Abschnitt „WebSphere Partner Gateway - Express konfigurieren“ auf Seite 231.

Beachten Sie, dass 'AS2toPartner2' als 'online' mit dem Status **Aktiviert** angezeigt wird.

Definieren Sie 'AS2toPartner2' mit Hilfe der folgenden Schritte als Standardziel für Partner 2:

1. Klicken Sie auf **Standardziele anzeigen**.
2. Wählen Sie in der Liste **Produktion** den Eintrag **AS2toPartner2** aus.
3. Klicken Sie auf **Speichern**.

### Interaktion definieren

Eine Dokumentenflussdefinition ist eine Sammlung von "Metainformationen", die die Funktionen zur Dokumentverarbeitung des Partners definiert. Damit ein System ein Geschäftsdokument verarbeiten kann, müssen zwei oder mehr Dokumentenflussdefinitionen verknüpft werden, um eine Interaktion zu erstellen.

Zur Erstellung einer Interaktion zwischen Partner 1 und Partner 2 führen Sie die folgenden Schritte aus:

1. Klicken Sie auf **Hubadmin > Hubkonfiguration > Dokumentdefinition**.
2. Klicken Sie auf **Interaktionen verwalten** und anschließend auf **Interaktion erstellen**.
3. Wählen Sie in der Spalte 'Quelle' folgende Werte aus:
  - a. Paket: **None**
  - b. Protokoll: **EDI-X12 (ALL)**
  - c. Dokumentenfluss: **ISA (ALL)**
4. Wählen Sie in der Spalte 'Empfänger' folgende Werte aus:
  - a. Paket: **AS**
  - b. Protokoll: **EDI-X12 (ALL)**
  - c. Dokumentenfluss: **ISA (ALL)**
5. Legen Sie die Aktion als **Pass-Through** fest.

6. Klicken Sie auf **Speichern**.
7. Klicken Sie erneut auf die Option **Interaktion erstellen**.
8. Wählen Sie in der Spalte 'Quelle' folgende Werte aus:
  - a. Paket: **AS**
  - b. Protokoll: **EDI-X12 (ALL)**
  - c. Dokumentenfluss: **ISA (ALL)**
9. Wählen Sie in der Spalte 'Empfänger' folgende Werte aus:
  - a. Paket: **None**
  - b. Protokoll: **EDI-X12 (ALL)**
  - c. Dokumentenfluss: **ISA (ALL)**
10. Legen Sie die Aktion als **Pass-Through** fest.
11. Klicken Sie auf **Speichern**.

### Partnerverbindungen erstellen

Partnerverbindungen bilden den Mechanismus, der das System in die Lage versetzt, Dokumente zwischen dem internen Partner und seinen verschiedenen Partnern zu verarbeiten und weiterzuleiten. Verbindungen enthalten die erforderlichen Informationen für den ordnungsgemäßen Austausch jedes einzelnen Dokumentenflusses.

Zur Erstellung einer Partnerverbindung zwischen Partner 1 und Partner 2 führen Sie die folgenden Schritte aus:

1. Klicken Sie im Hauptmenü auf **Kontenadmin** und auf **Partnerverbindungen** in der horizontalen Navigationsleiste.
2. Wählen Sie in der Liste **Quelle** den Eintrag **Partner 1** aus.
3. Wählen Sie in der Liste **Ziel** den Eintrag **Partner 2** aus.
4. Klicken Sie auf **Suchen**.
5. Aktivieren Sie die Partnerverbindung, die unten angezeigt wird, indem Sie auf die Schaltfläche **Aktivieren** klicken. Dadurch sollte die B2B-Funktionalität angezeigt werden, die in Tabelle 100 zu sehen ist.

*Tabelle 100. Aktivieren der Partnerverbindung von Partner 1 zu Partner 2*

Dokumentenflusstyp	Quelle	Ziel
Paket	Nein (N/V)	AS (N/V)
Protokoll	EDI-X12 (ALLE)	EDI-X12 (ALLE)
Dokumentenfluss	ISA (ALL)	ISA (ALL)

Zur Erstellung einer Partnerverbindung, bei der Partner 2 die Quelle und Partner 1 das Ziel ist, führen Sie die folgenden Schritte aus:

1. Klicken Sie im Hauptmenü auf **Kontenadmin** und auf **Partnerverbindungen** in der horizontalen Navigationsleiste.
2. Wählen Sie in der Liste **Quelle** den Eintrag **Partner 2** aus.
3. Wählen Sie in der Liste **Ziel** den Eintrag **Partner 1** aus.
4. Klicken Sie auf **Suchen**.
5. Aktivieren Sie die Verbindung mit den in Tabelle 101 auf Seite 230 gezeigten Details.

Tabelle 101. Aktivieren der Partnerverbindung von Partner 2 zu Partner 1

Dokumentenflusstyp	Quelle	Ziel
Paket	AS (N/V)	Nein (N/V)
Protokoll	EDI-X12 (ALLE)	EDI-X12 (ALLE)
Dokumentenfluss	ISA (ALL)	ISA (ALL)

## Empfänger erstellen

Die Anzeige der Empfängerliste stellt Informationen zu Positionen bereit, die es dem Document Manager von WebSphere Partner Gateway ermöglichen, Dokumente aus dem entsprechenden System auf der Basis des Transporttyps des eingehenden Dokuments abzurufen. Sie können separate Empfängerkonfigurationen auf der Basis des Transporttyps erstellen. Der Document Manager kann dann die Positionen von Dokumentrepositorys auf mehreren Web-, FTP- und POP-Mail-Servern, einschließlich interner Verzeichnisse und JMS-Warteschlangen, nach eingehenden Dokumenten abfragen.

Wenn der Document Manager ein Dokument aus der Position auf der Basis eines vordefinierten Empfängers abrufen kann, kann die Routinginfrastruktur das Dokument auf der Basis der Kanalkonfiguration verarbeiten.

Zum Empfangen einer EDI-Transaktion von WebSphere Data Interchange erstellen Sie einen neuen JMS-Empfänger, indem Sie folgende Schritte ausführen:

1. Klicken Sie auf **Hubadmin > Hubkonfiguration > Empfänger**.
2. Klicken Sie auf **Empfänger erstellen**.
3. Ordnen Sie die in Tabelle 102 gezeigten Eigenschaften zu.

Tabelle 102. Empfängereigenschaften für den Empfang über JMS

Empfängereigenschaft	Wert
Empfängername	WdiJmsListener
Transport	JMS
Zieltyp	Produktion
JMS-Provider-URL	file:///opt/mqm/java/JNDI/WdiJms
JMS-Warteschlangenname	EDI_IN_B
JMS-Factory-Name	HUB60_QM_QCF
JNDI-Factory-Name	com.sun.jndi.fscontext.ReffSContextFactory

Ein zweiter Empfänger ist für den Empfang von EDI-Dokumenten von Partner 2 über AS2 erforderlich. Führen Sie die folgenden Schritte aus, um diesen Empfänger zu erstellen:

1. Klicken Sie auf **Hubadmin > Hubkonfiguration > Empfänger**.
2. Klicken Sie auf **Empfänger erstellen**.
3. Ordnen Sie die Eigenschaften aus der folgenden Tabelle 103 zu:

Tabelle 103. Empfängereigenschaften für den Empfang über AS2

Empfängername	HubAS2Listener
Transport	HTTP/S
Zieltyp	Produktion

Tabelle 103. Empfängereigenschaften für den Empfang über AS2 (Forts.)

URI	/bcgreceiver/submit <b>Anmerkung:</b> Die URI für den Empfang von HTTP/S muss immer mit /bcgreceiver beginnen.
-----	-------------------------------------------------------------------------------------------------------------------

4. Klicken Sie auf **Speichern**.

## WebSphere Partner Gateway - Express konfigurieren

Dieser Abschnitt beschreibt die Schritte zur Konfiguration der Umgebung des externen Partners. In diesem Fall wird diese Umgebung mit einem WebSphere Partner Gateway - Express-System hergestellt. In dem Beispielszenario, das in diesem Kapitel vorgestellt wird, verwendet Partner 2 das Produkt WebSphere Partner Gateway - Express zum Senden und Empfangen von EDI-Dokumenten über HTTP-AS2.

Um EDI-Dokumente über HTTP-AS2 erfolgreich empfangen zu können, führen Sie die folgenden Schritte aus:

1. „Mein Profil konfigurieren“
2. „Partner für Partner 1 erstellen“ auf Seite 232
3. „Partner für Partner 1 konfigurieren“ auf Seite 232

### Mein Profil konfigurieren

Im ersten Schritt müssen Sie ein Profil für Partner 2 in WebSphere Partner Gateway - Express erstellen. Zur Erstellung eines Profils für Partner 2 führen Sie die folgenden Schritte aus:

1. Klicken Sie im Hauptmenü auf **Konfiguration**.
2. Klicken Sie auf **Mein Profil** in der horizontalen Navigationsleiste.
3. Geben Sie die Details wie in Tabelle 104 skizziert ein.

Tabelle 104. Details für 'Mein Profil'

Unsichere Domäne der Empfangsadresse	IP_COMPUTER_C  Dabei ist IP_COMPUTER_C die Internet Protocol-Adresse (IP-Adresse) von Computer C, auf dem WebSphere Partner Gateway - Express ausgeführt wird.
Unsicherer Port der Empfangsadresse	80  Dabei ist 80 der Port, der zur Verwendung durch WebSphere Partner Gateway - Express bei der Installation zugeordnet wurde.
AS2-Absender-ID	987654321
Geschäfts-ID-Typ	DUNS
Geschäftskennung	987654321

4. Klicken Sie auf **Speichern**.

## Partner für Partner 1 erstellen

Partner 1 muss als Partner für WebSphere Partner Gateway - Express angegeben werden. Zur Erstellung von Partner 1 als Partner führen Sie die folgenden Schritte aus:

1. Klicken Sie im Hauptmenü auf **Konfiguration**.
2. Klicken Sie auf **Partner** in der horizontalen Navigationsleiste.
3. Klicken Sie auf die Schaltfläche **Partner erstellen**.
4. Ordnen Sie die folgenden Werte zu:
  - a. Partnername: **partner1**
  - b. AS2-Partner-ID: **123456789**
5. Klicken Sie auf **Speichern**.

In der Anzeige für die Partnerverwaltung können Sie die Details für Partner 1 sehen.

## Partner für Partner 1 konfigurieren

Wenn der Partner für Partner 1 vorhanden ist, müssen Sie Partner 1 für AS2 und HTTP konfigurieren. Diese Konfiguration definiert die Parameter, die von WebSphere Partner Gateway - Express zum Senden und Empfangen über HTTP und AS2 an und von Partner 1 benötigt werden.

Zur Konfiguration von Partner 1 für HTTP und AS2 führen Sie die folgenden Schritte aus:

1. Klicken Sie im Hauptmenü auf **Konfiguration**.
2. Klicken Sie auf **AS2** in der horizontalen Navigationsleiste.
3. Wählen Sie **partner1** in der Liste für ausgewählte Partner aus, und klicken Sie auf **Bearbeiten**.
4. Definieren Sie die ausgehende Zieladresse von Partner 1 wie folgt:  
`http://IP_COMPUTER_B:7080/bcgreceiver/submit`  
Dabei ist *IP\_COMPUTER\_B* die IP-Adresse von Computer B.
5. Klicken Sie auf **Speichern**.
6. Klicken Sie auf **HTTP** in der horizontalen Navigationsleiste. (**partner1** sollte immer noch als ausgewählter Partner angezeigt werden.)
7. Klicken Sie auf **Bearbeiten**.
8. Legen Sie den eingehenden Benutzernamen und das Kennwort fest:  
Benutzername: **partner1**  
Kennwort: **partner1**  
Beachten Sie, dass diese Werte zuvor im Beispielschritt zur Erstellung des Standardziels für Partner 2 in WebSphere Partner Gateway Enterprise Edition auf Computer B angegeben wurden.
9. Definieren Sie die ausgehende Zieladresse wie folgt:  
`http://IP_COMPUTER_B:7080/bcgreceiver/submit`
10. Klicken Sie auf **Speichern**.

**Wichtig:** Wenn Sie diese Änderungen in WebSphere Partner Gateway - Express ausgeführt haben, melden Sie sich von der Community Console ab, und stoppen Sie das Ziel. Starten Sie das Ziel und die Community Console erneut, um alle Änderungen in Kraft zu setzen.

---

## Zusammenfassung

In diesem Kapitel wurde der Prozess beschrieben, durch den WebSphere Partner Gateway mit WebSphere Data Interchange interagiert. Darüber hinaus wurden Prozeduren zur Einrichtung des Beispielszenarios erläutert, das im Abschnitt „Beispielszenario für dieses Kapitel“ auf Seite 214 beschrieben wird.

Wie zu Beginn dieses Kapitels erwähnt, können Sie das Lernprogramm *Integrating WebSphere Data Interchange V3.2 with WebSphere Business Integration Connect V4.2* durcharbeiten, um tatsächlich eine Beispielkonfiguration zu erstellen. Das Lernprogramm stellt Beispielscripts und Zuordnungen zur Verfügung, die Ihnen bei der Konfiguration der Umgebung helfen und zeigen, wie ein Beispielaustausch zu testen ist. Für den Zugriff auf das Lernprogramm besuchen Sie die folgende Website: [www.ibm.com/developerworks/websphere/](http://www.ibm.com/developerworks/websphere/)

Suchen Sie dort nach dem Titel des Lernprogramms.

Beachten Sie, dass Sie für den Dokumentenfluss **ISA (ALL)** (und nicht **ALL (ALL)**) auswählen müssen, wenn Sie die B2B-Funktionalität der Partner einrichten. Auch die Interaktionen, die Sie einrichten, und die Verbindungen, die Sie aktivieren, haben **ISA (ALL)** für den Dokumentenfluss.



---

## **Teil 5. Integration mit anderen Produkten**



---

## Kapitel 13. Integration mit WebSphere Transformation Extender

In diesem Kapitel wird die Integration von WebSphere Partner Gateway mit WebSphere Transformation Extender beschrieben.

---

### Einführung

#### Was ist WebSphere Transformation Extender?

WebSphere Transformation Extender ist eine leistungsstarke, transaktionsorientierte Datenintegrationslösung, mit deren Hilfe die Transformation komplexer Transaktionen mit großen Datenmengen automatisiert werden kann, ohne dass eine manuelle Codierung erforderlich ist. Unternehmen können so schnelle Investitionserträge realisieren. Das Produkt unterstützt EDI, XML, SWIFT, HIPAA und weitere, auf Standards basierende B2B-Integrationslösungen, sowie die Echtzeitintegration von Daten aus unterschiedlichen Anwendungen, Datenbanken, Kommunikationstechnologien und der Messaging-Middleware eines Unternehmens.

WebSphere Transformation Extender ist für die Transformation und Weiterleitung von Daten aus Quellen- in Zielsysteme in Batch- und Echtzeitumgebungen zuständig. Zu den Quellen können Dateien, relationale Datenbanken, Messaging-Middleware, Standardsoftware und andere externe Quellen gehören. Nach Abruf der Daten aus der Quelle werden diese von WebSphere Transformation Extender transformiert und nach Bedarf an verschiedene Ziele weitergeleitet, wobei Inhalt und Format an das jeweilige Zielsystem angepasst sind. WebSphere Transformation Extender bietet die folgende Funktionalität:

- Konnektivität mit einer Vielzahl von Mainframe-, Unternehmens- und traditionellen Anwendungen, Datenbanken, Messaging-Systemen und externen Informationsquellen.
- Umfassende Bibliothek mit mehr als 120 vordefinierten Funktionen zur Reduzierung der Entwicklungszeit und Vereinfachung der Spezifikation von Regeln für Validierung, Transformation und Weiterleitung.
- Mehrere Optionen zur Ausführung von Transformationen zum richtigen Zeitpunkt und auf die richtige Weise - egal ob Batch-, Echtzeit- oder integrierte Ausführung.
- Auf Unternehmen abgestimmte Funktionen für Entwicklung, Implementierung, und Verwaltung sowie Plattformentstützung mit hoher Verfügbarkeit. Laufende Verwaltungstasks und Implementierungsrisiken können somit reduziert und Resultate schneller als bei manueller Codierung erzielt werden.

#### Zusammenarbeit zwischen WebSphere Transformation Extender und WebSphere Partner Gateway

WebSphere Transformation Extender wird in WebSphere Partner Gateway zur Ausführung von Dokumenttransformationen verwendet. Für die Integration mit WebSphere Transformation Extender stehen zwei Methoden zur Verfügung.

Bei der einen Methode gibt es keine direkte Interaktion zwischen WebSphere Partner Gateway und WebSphere Transformation Extender. Bei dieser Methode wird WebSphere Transformation Extender wie andere Back-End-Anwendungen behandelt. Ferner wird WebSphere Message Broker verwendet, in das wiederum

WebSphere Transformation Extender integriert ist. Bei dieser Methode wird erwartet, dass das Dokument nicht in WebSphere Partner Gateway transformiert, sondern an WebSphere Message Broker weitergeleitet wird, wo mit Hilfe von WebSphere Transformation Extender die Transformation ausgeführt wird.

Bei der anderen Methode erfolgt eine direkte Interaktion zwischen WebSphere Partner Gateway und WebSphere Transformation Extender. In WebSphere Partner Gateway erfolgt ein Aufruf an WebSphere Transformation Extender, wobei ein Benutzerexit als Handler für die Aktion verwendet wird. Es wird ein Benutzerexit implementiert, der WebSphere Transformation Extender mit Hilfe der Java-RMI-APIs von WebSphere Transformation Extender aufruft, damit die Transformation ausgeführt wird.

---

## Integration mit WebSphere Transformation Extender planen

Wie in der Einführung bereits erwähnt, können für die Integration mit WebSphere Transformation Extender zwei Methoden verwendet werden. Bei der einen Methode wird WebSphere Message Broker und bei der anderen Methode ein Benutzerexit verwendet.

### Integration mit WebSphere Message Broker

#### Voraussetzungen für die Verwendung von WebSphere Message Broker

Zur Verwendung von WebSphere Message Broker mit WebSphere Partner Gateway benötigen Sie Folgendes:

- Die von WebSphere Partner Gateway unterstützte Version von WebSphere Message Broker
- Die von WebSphere Message Broker unterstützte Version von WebSphere Transformation Extender
- WebSphere Transformation Extender für WebSphere Message Broker

#### Verwendung von WebSphere Message Broker planen

WebSphere Partner Gateway interagiert über WebSphere Message Broker indirekt mit WebSphere Transformation Extender. Dies bedeutet, dass die Topologie bei WebSphere Partner Gateway und bei WebSphere Transformation Extender auf WebSphere Message Broker ausgerichtet ist. Die Erstellung, Konfiguration und Implementierung von WebSphere Transformation Extender-Zuordnungen erfolgt im Kontext von WebSphere Message Broker.

### Integration mit einem WebSphere Partner Gateway-Benutzerexit

#### Voraussetzungen für die Verwendung eines WebSphere Partner Gateway-Benutzerexits aus einer Aktion

WebSphere Partner Gateway Version 6.1 unterstützt die Integration mit WebSphere Transformation Extender 8.1 mit Fixpack 1 oder höher. Wenn Sie Fixpack 1 und einen lokalen Aufruf verwenden, muss die Programmkorrektur 105385 bei der WebSphere Transformation Extender-Unterstützung angefordert werden. In zukünftigen WebSphere Transformation Extender-Fixpacks wird die Programmkorrektur möglicherweise bereits enthalten sein. Der lokale Aufruf, auch prozessinterner Aufruf genannt, wird entweder mit den WebSphere Transformation Extender-Java-APIs oder mit den WebSphere Transformation Extender-RMI-APIs verwendet (sofern der RMI-Server nicht eingesetzt wird).

## Verwendung eines WebSphere Partner Gateway-Benutzerexits planen

WebSphere Transformation Extender kann innerhalb von WebSphere Partner Gateway entweder lokal oder fern über die Java-RMI-APIs aufgerufen werden. Die passende Methode hängt von der jeweiligen Implementierungsumgebung ab. Ein ferner Aufruf bietet größere Flexibilität und Sicherheit für WebSphere Partner Gateway.

**Lokaler (prozessinterner) Aufruf:** Die folgenden Informationen beziehen sich auf einen lokalen Aufruf von WebSphere Transformation Extender:

- Ein lokaler Aufruf kann entweder unter Verwendung der Java-APIs oder der RMI-APIs von WebSphere Transformation Extender im lokalen Modus erfolgen (d. h. der Wert für den Hostnamen und die Portnummer ist 0).
- WebSphere Transformation Extender muss auf derselben Maschine wie WebSphere Partner Gateway ausgeführt werden. Dies bedeutet, dass jede Instanz von WebSphere Partner Gateway die Installation von WebSphere Transformation Extender erfordert.
- Die Dokumenttransformation erfolgt im selben WebSphere Partner Gateway-Thread und -Speicher, von wo aus die WebSphere Transformation Extender-API aufgerufen wird.
- WebSphere Partner Gateway und WebSphere Transformation Extender müssen auf einem Betriebssystem installiert sein, das von beiden Produkten unterstützt wird.
- Es gibt eine WebSphere Transformation Extender-Laufzeitbibliothek, die in der Umgebungsvariable PATH von WebSphere Partner Gateway enthalten sein muss.
- Die WebSphere Transformation Extender-Zuordnungen müssen für die WebSphere Transformation Extender-APIs verfügbar sein.
- Unerwartete Fehler während der Transformation können negative Auswirkungen auf WebSphere Partner Gateway haben.

**Ferner Aufruf:** Die folgenden Informationen beziehen sich auf einen fernen Aufruf von WebSphere Transformation Extender:

- WebSphere Partner Gateway kann sich auf einer anderen Maschine als WebSphere Transformation Extender befinden. Dies bedeutet, dass mehrere Instanzen von WebSphere Partner Gateway dieselbe WebSphere Transformation Extender-Instanz gemeinsam nutzen können.
- Die Ausführung von WebSphere Partner Gateway auf einem Windows-Betriebssystem ist nicht erforderlich.
- Der RMI-Server von WebSphere Transformation Extender muss ausgeführt werden.
- Die Dokumenttransformation erfolgt auf dem RMI-Server von WebSphere Transformation Extender.
- Das allgemeine Dateisystem von WebSphere Partner Gateway muss dem RMI-Server von WebSphere Transformation Extender für das Eingabe- und das Ausgabedokument zur Verfügung stehen.
- Die WebSphere Transformation Extender-Zuordnungen müssen für den RMI-Server von WebSphere Transformation Extender verfügbar sein.
- Unerwartete Fehler während der Transformation haben keine negativen Auswirkungen auf WebSphere Partner Gateway.

---

## Integration von WebSphere Message Broker mit WebSphere Transformation Extender

Bei dieser Implementierung wird davon ausgegangen, dass Transformationen von WebSphere Transformation Extender unter Verwendung von WebSphere Message Broker ausgeführt werden, obwohl mit den vorhandenen Funktionen Transformationen in WebSphere Partner Gateway weiterhin möglich sind. Detaillierte Informationen zu WebSphere Message Broker finden Sie in Kapitel 11, „Integration mit WebSphere Message Broker“, auf Seite 191. Informationen zur Integration mit WebSphere Message Broker finden Sie in der WebSphere Transformation Extender-Dokumentation.

---

## Integration mit WebSphere Transformation Extender-Benutzerexit

Vor der Integration mit WebSphere Partner Gateway muss WebSphere Transformation Extender gemäß den Anforderungen und Anweisungen für dieses Produkt installiert werden. Darüber hinaus benötigt die Maschine, auf der WebSphere Transformation Extender ausgeführt wird, Zugriff auf das allgemeine Dateisystem von WebSphere Partner Gateway, um das zu transformierende Eingabedokument lesen und das Ausgabedokument schreiben zu können.

Informationen zu den Konfigurationsanforderungen finden Sie in der WebSphere Transformation Extender-Produktdokumentation. Konfigurationshinweise für den RMI-Server von WebSphere Transformation Extender finden Sie im Abschnitt „WebSphere Transformation Extender - Konfiguration des RMI-Servers“ auf Seite 244.

## WebSphere Partner Gateway-Benutzerexit für WebSphere Transformation Extender erstellen

Der WebSphere Transformation Extender-Benutzerexit sollte die RMI-APIs von WebSphere Transformation Extender verwenden, da diese die höchste Flexibilität für die Installationsposition von WebSphere Transformation Extender bieten und zulassen, dass mehrere Instanzen von WebSphere Partner Gateway dieselbe WebSphere Transformation Extender-Installation verwenden. Ferner muss der Benutzerexit vermutlich in Bezug auf den Hostnamen und den Port konfiguriert werden, für die der RMI-Server von WebSphere Transformation Extender empfangsbereit ist.

Informationen zu den Arbeitsschritten, die zur Erstellung des Benutzerexits erforderlich sind, finden Sie in den folgenden Quellen:

- Informationen zur Implementierung von variablen Workflow-Handlern finden Sie im *WebSphere Partner Gateway Programmer Guide*.
- Informationen zur Verwendung der WebSphere Transformation Extender-APIs finden Sie in der WebSphere Transformation Extender-Dokumentation.
- Informationen zu diesem Benutzerexit finden Sie im Abschnitt zum WebSphere Partner Gateway-Beispielbenutzerexit.

## WebSphere Partner Gateway für den Benutzerexit konfigurieren

WebSphere Partner Gateway muss unter Verwendung von WebSphere Partner Gateway Console für den WebSphere Transformation Extender-Benutzerexit konfiguriert werden. Zuerst müssen Sie den Benutzerexit für WebSphere Partner Gate-

way definieren und die Aktion erstellen, die den Benutzerexit verwendet. Führen Sie insbesondere die folgenden Schritte aus:

1. Laden Sie die Definition des XML-Handlers hoch, die Sie für den Benutzerexit erstellt haben.
2. Erstellen Sie eine Aktion, die den Benutzerexit verwendet.

Nachdem Sie den WebSphere Transformation Extender-Benutzerexit hochgeladen und die Aktion erstellt haben, konfigurieren Sie den Benutzerexit wie für andere WebSphere Partner Gateway-Dokumentenflüsse. Führen Sie insbesondere die folgenden Schritte aus:

1. Konfigurieren Sie die Dokumenttypen.
2. Definieren Sie die zugehörige B2B-Funktionalität.
3. Importieren Sie die WebSphere Partner Gateway-Zuordnungen, die vom Benutzerexit verwendet werden.

**Anmerkung:** Die Zuordnung hat das Format, das Sie für den Benutzerexit definiert haben.

4. Konfigurieren Sie Interaktionen.
5. Verwalten Sie die Verbindungen.

Informationen zur Ausführung dieser Schritte finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

## WebSphere Partner Gateway-Laufzeitkonfiguration für den Benutzerexit

Im Abschnitt „WebSphere Partner Gateway für den Benutzerexit konfigurieren“ auf Seite 240 haben Sie WebSphere Partner Gateway für die Verwendung des WebSphere Transformation Extender-Benutzerexits konfiguriert. Bevor Sie Dokumente über WebSphere Partner Gateway verarbeiten können, müssen Sie den Laufzeitcode verfügbar machen, indem Sie die Java-Klassendateien installieren.

Gehen Sie wie folgt vor, um die Java-Klassendateien zu installieren:

1. Stellen Sie die Java-Klassendateien für den Benutzerexit in das Verzeichnis {WPG-Installationsverz}\router\lib\userexits.
2. Kopieren Sie die JAR-Datei dtxpi.jar aus dem Development Kit vom WebSphere Transformation Extender-Installationsverzeichnis in das Verzeichnis {WPG-Installationsverz}\router\lib\userexits. Diese JAR-Datei enthält die WebSphere Transformation Extender-Laufzeitklassen, die zum Aufrufen der Zuordnungstransformation von WebSphere Transformation Extender erforderlich sind.
3. WebSphere Partner Gateway muss erneut gestartet werden, damit die neuen JAR-Dateien wirksam werden.
4. Bei einer lokalen Ausführung fügen Sie das WebSphere Transformation Extender-Installationsverzeichnis außerdem in den Systempfad ein. WebSphere Partner Gateway muss erneut gestartet werden, damit die neuen Pfadeinstellungen wirksam werden.

## WebSphere Partner Gateway-Beispielbenutzerexit

Der Beispielbenutzerexit soll Sie bei der Integration von WebSphere Partner Gateway mit WebSphere Transformation unterstützen.

## WebSphere Transformation Extender-Beispielbenutzerexit - Übersicht

Der Beispielbenutzerexit stellt die Empfehlungen für das Aufrufen von WebSphere Transformation Extender aus einer WebSphere Partner Gateway-Aktion heraus dar, um eine Transformation auszuführen. Die tatsächliche Implementierung des Benutzerexits hängt davon ab, wie Sie die WebSphere Transformation Extender-Zuordnungen mit den zugehörigen Ein- und Ausgabekarten erstellen. Es handelt sich hier um ein Arbeitsbeispiel, das in WebSphere Partner Gateway ausgeführt werden kann. Die Readme-Dateien für die Installation und Verwendung von Beispiel, Quellcode, Eingabedateien und Zuordnungen befinden sich im Verzeichnis `Integration/WebSphereTransformationExtender` auf der Produkt-CD. In diesen Dateien können Sie weitere Einzelheiten nachlesen.

In diesem Beispiel gibt es zwei Typen von Zuordnungen. Zum einen die WebSphere Transformation Extender-Zuordnung, die mit WebSphere Transformation Extender Map Designer erstellt wird, und zum anderen die WebSphere Partner Gateway-Zuordnung, die Metadaten in Form einer Merkmaldatei enthält, um den Benutzerexit bei der Ausführung der WebSphere Transformation Extender-Zuordnung zu unterstützen. Das Format der WebSphere Partner Gateway-Zuordnung bezieht sich auf die Benutzerexit-Implementierung. Detaillierte Informationen zum Inhalt finden Sie in der WebSphere Partner Gateway-Beispielzuordnung.

### Beschreibung der Klassen des Beispielbenutzerexits

Diese Klassen versorgen die WebSphere Transformation Extender-APIs mit dem erforderlichen Schnittstellencode für den WebSphere Partner Gateway-Benutzerexit, um Operationen innerhalb einer Aktion ausführen zu können. Details zur Implementierung finden Sie im Java-Quellcode. Es folgt eine Beschreibung der Klassen im Beispielbenutzerexit:

#### WTXTransformationFactory

Diese Klasse ist der Benutzerexit-Handler für die Aktion. Sie erstellt und liefert die Instanz der Geschäftsprozess-Schnittstelle (`BusinessProcessInterface`), die in Abhängigkeit von den Handlerattributen verwendet werden muss. Die Handlerattribute selbst werden von der Geschäftsprozessinstanz für weitere Konfigurationsmaßnahmen verwendet.

#### WTXProcess

Diese Klasse ist die Superklasse für die Geschäftsprozessklassen, die eine Schnittstelle mit WebSphere Transformation Extender bilden. Sie enthält die übergreifende Flusslogik für die Aspekte von WebSphere Transformation Extender, die sich nicht auf die APIs beziehen. Details zum API-Aufruf von Transformation Extender sind in die Unterklassen der Methode `callWTX()` eingebunden. Die Methode `setCfg()` wird von den Unterklassen verwendet, um Informationen aus den Handlerattributen festzulegen, die sich auf die jeweilige Unterklasse beziehen.

#### WTXProcessRMI

Diese Klasse ist eine Unterklasse von `WTXProcess`. Diese Geschäftsprozessklasse ruft unter Verwendung des RMI-Servers von WebSphere Transformation Extender eine Zuordnung fern auf, und verwendet dazu die WebSphere Transformation Extender-RMI-APIs. Die Zuordnung kann zur Verarbeitung auch lokal ohne Verwendung des RMI-Servers von WebSphere Transformation Extender aufgerufen werden. Besonderheiten:

- Die Eingaben in die WebSphere Transformation Extender-Zuordnung sollten Dateieingaben sein. Anhand der Informationen aus der WebSphere Partner Gateway-Zuordnung werden der Zuordnungsname sowie weitere Anforderungen für einen Laufzeitaufruf ermittelt.

- Anhand des RMI-Serverwerts aus der Konfiguration des Benutzerexit-Handlers wird festgestellt, ob der RMI-Server von WebSphere Transformation Extender verwendet wird. Bei Verwendung des RMI-Servers von WebSphere Transformation Extender werden der Hostname und die Portnummer aus der Konfiguration des Benutzerexit-Handlers für den Zugriff auf den WebSphere Transformation Extender-RMI-Server verwendet. Die WebSphere Transformation Extender-RMI-APIs können für den lokalen Aufruf einer Zuordnung verwendet werden; wenn die Zuordnung laut Konfiguration des Benutzerexit-Handlers also lokal aufgerufen werden soll, kann derselbe Code verwendet werden. Für einen lokalen Aufruf der Zuordnung muss die Laufzeit-DLL-Datei von WebSphere Transformation Extender im Pfad verfügbar sein.
- Beachten Sie, dass WebSphere Transformation Extender unter Windows ausgeführt wird; wenn also WebSphere Partner Gateway auf einer UNIX-Maschine installiert ist, muss ein ferner Aufruf verwendet werden.

#### **WTXProcessLocal**

Diese Klasse ist eine Unterklasse von WTXProcess. Die Geschäftsprozessklasse ruft eine Zuordnung in WebSphere Transformation Extender lokal auf. Einige Besonderheiten: Bei den Eingaben in die WebSphere Transformation Extender-Zuordnung sollte es sich um Dateieingaben handeln. Anhand der Informationen aus der WebSphere Partner Gateway-Zuordnung werden der Zuordnungsname sowie weitere Anforderungen für einen Laufzeitaufruf ermittelt.

Zur Verwendung dieser Geschäftsprozessklasse muss die Laufzeit-DLL-Datei von WebSphere Transformation Extender im Pfad verfügbar sein.

**Anmerkung:** Die Klasse WTXTransformationFactory instanziiert diese Unterklasse nicht, da die WebSphere Transformation Extender-RMI-APIs in WTXProcessRMI auch für lokale Aufrufe verwendet werden können. Diese Klasse dient als Beispiel für die Verwendung der WebSphere Transformation Extender-Java-APIs.

#### **Grundlegender Laufzeitvorgang des Beispielbenutzerexits**

Die folgenden Informationen beziehen sich auf den grundlegenden Laufzeitvorgang des Beispielbenutzerexits:

- WebSphere Partner Gateway empfängt ein XML-Dokument vom externen Partner.
- Die WebSphere Partner Gateway-Verbindung wird auf der Basis des XML-Formats und der aus dem XML-Dokument abgerufenen Geschäfts-IDs ausgewählt; dann wird das Dokument unter Verwendung dieser Verbindung verarbeitet.
- Die angepasste Aktion in dieser Verbindung ist für die Verwendung des Beispielbenutzerexits von WebSphere Transformation Extender für Transformationen konfiguriert. Bei Aufruf des WebSphere Transformation Extender-Benutzerexits für Transformationen geschieht Folgendes:
  - Der Hostname und die Portnummer des Systems werden abgerufen, auf dem der WebSphere Transformation Extender-RMI-Server ausgeführt wird.
  - Die Informationen für die Eingabe- und die Ausgabekarte der Zuordnung werden basierend auf der Transformation Extender-Zuordnung von WebSphere Partner Gateway festgelegt.
  - Die WebSphere Transformation Extender-Zuordnung wird aufgerufen, und die Ergebnisse werden in das Geschäftsdokumentobjekt übertragen.

## WebSphere Transformation Extender - Konfiguration des RMI-Servers

In diesem Abschnitt werden einige Aspekte der Konfiguration des WebSphere Transformation Extender-RMI-Servers in Bezug auf WebSphere Partner Gateway beschrieben. Detaillierte Informationen zu den Konfigurationseigenschaften finden Sie in der WebSphere Transformation Extender-Dokumentation.

Wenn die WebSphere Transformation Extender-Zuordnung lokal aufgerufen wird, ist die Konfiguration des RMI-Servers nicht relevant. Bei einem lokalen (prozess-internen) Aufruf wird die Zuordnung innerhalb desselben Threads aufgerufen, der in WebSphere Partner Gateway ausgeführt wird. Bei einem fernen Aufruf der Zuordnung wird der RMI-Server verwendet; in diesem Fall sind die Informationen in diesem Abschnitt relevant. Weitere Hinweise zu prozessinternen (lokalen) Aufrufen finden Sie im Abschnitt zur Merkmaldatei `server.mode.multi.process`.

Der RMI-Server wird über die Befehlszeile gestartet. Es wird die Verwendung des Befehlszeilenparameters `'-verbose'` empfohlen, da dieser Parameter einige Merkmale der Laufzeitkonfiguration an die Standardausgabe ausgibt.

Beim Starten des RMI-Servers wird eine Gruppe von Konfigurationsparametern aus einer Merkmaldatei gelesen. Im Folgenden werden einige Aspekte der Konfiguration dieser Parameter in Bezug auf WebSphere Partner Gateway beschrieben.

### **server.mode.multi.process**

**Empfohlene Einstellung:** `true`

**Beschreibung:** Diese Eigenschaft führt den RMI-Server mit der Option für Mehrprozesskonfiguration aus. Jeder Aufruf an den RMI-Server wird in einem separaten Prozess ausgeführt, so dass im Falle von Ausnahmerebedingungen für eine Zuordnung der bestmögliche Schutz gewährleistet ist. Wenn eine Ausnahmerebedingung für eine Zuordnung auftritt, wird der RMI-Server weiterhin ausgeführt. Bei einer Einzelprozesskonfiguration (d. h. `server.mode.multi.process=false`) wird jeder Aufruf an den RMI-Server im selben Prozess, aber in einem anderen Thread ausgeführt. Wenn jedoch eine Ausnahmerebedingung für die Zuordnung auftritt, wird die Ausführung des RMI-Servers möglicherweise gestoppt.

#### **Aspekte der Mehrprozesskonfiguration:**

- Jeder Zuordnungsaufruf wird in einem eigenen Prozess ausgeführt.
- Jeder Prozess verfügt über einen eigenen Java-Heap-Speicher.
- Zusätzliche Sicherheit, da eine Ausnahmerebedingung für eine Zuordnung nicht dazu führt, dass der RMI-Server gestoppt wird.
- Von jedem Prozess wird eine WebSphere Transformation Extender-Instanz in den Speicher geladen, der bis zu 30 MB aufnehmen kann.
- Wenn ein Prozess nicht verfügbar ist, gibt die Eigenschaft `pool.acquire.process.timeout` des RMI-Servers den Zeitraum bis zur Zeitlimitüberschreitung des Zuordnungsaufrufs an.
- Um sicherzustellen, dass genügend Prozesse verfügbar sind, müssen Sie mindestens so viele Prozesse konfigurieren, wie aktive WebSphere Partner Gateway Document Manager-Instanzen multipliziert mit der Anzahl der Threads in den Document Manager-Instanzen vorhanden sind. Der Document Manager besteht aus mehreren Komponenten. Die Komponente, die den Benutzerexit ausführt, ist die Geschäftsprozessengine, die standardmäßig für zwei Threads konfiguriert ist. Wenn also in der Standardkonfiguration zwei Document Manager-Instanzen installiert sind, werden vier Threads ausgeführt (zwei Instanzen x zwei Threads);

der RMI-Server muss deshalb für mindestens vier Prozesse konfiguriert werden. Die Steuerung der Prozessanzahl erfolgt über die Konfigurationseigenschaft `pool.max.process.count` des RMI-Servers.

#### **Aspekte der Einzelprozesskonfiguration:**

- Jeder Zuordnungsaufwurf wird in einem eigenen Thread, aber im selben Prozess wie andere Zuordnungsthreads ausgeführt.
- Wenn eine Ausnahmebedingung für eine Zuordnung auftritt, kann der RMI-Server gestoppt werden.
- Weniger Speicheraufwand, da nur eine WebSphere Transformation Extender-Instanz geladen wird.

#### **Aspekte der prozessinternen Konfiguration:**

- Die Ausführung erfolgt lokal ohne den RMI-Server. Wird für WebSphere Transformation Extender in einer Serverumgebung nicht empfohlen, da die Zuordnung im aufrufenden Thread ausgeführt wird, der in diesem Fall WebSphere Partner Gateway ist. Eine abnormale Ausnahmebedingung kann zu Fehlern in WebSphere Partner Gateway führen.
- WebSphere Transformation Extender muss auf derselben Maschine wie WebSphere Partner Gateway installiert werden.
- Ist auf Windows-Betriebssysteme beschränkt.

### **pool.mode.managed**

**Empfohlene Einstellung:** true

**Beschreibung:** Diese Eigenschaft verwaltet die Anzahl der Prozesse, die gleichzeitig ausgeführt werden können. Ist dieser Wert auf 'false' gesetzt, gibt es keine Einschränkung. Wird zur Vermeidung von Situationen empfohlen, in denen Prozesse unkontrolliert zunehmen.

### **pool.max.process.count**

**Empfohlene Einstellung:** (Gesamtzahl der WebSphere Partner Gateway-Instanzen) x (zugehörige Threadkonfiguration)

**Beschreibung:** Der Document Manager von WebSphere Partner Gateway besteht aus mehreren Komponenten. Die Komponente, die den Benutzerexit ausführt, ist die Geschäftsprozessengine, die standardmäßig für zwei Threads konfiguriert ist. Wenn also in der Standardkonfiguration zwei Document Manager-Instanzen installiert sind, werden vier Threads ausgeführt (zwei Instanzen x zwei Threads); der RMI-Server muss deshalb für mindestens vier Prozesse konfiguriert werden. Diese Eigenschaft stellt sicher, dass genügend Prozesse für die Verarbeitung der Benutzerexitaufrufe an den RMI-Server vorhanden sind. Wenn die Anzahl der Prozesse niedriger ist als die Anzahl der WebSphere Partner Gateway-Threads, muss ein Benutzerexitaufruf an den RMI-Server auf einen verfügbaren Prozess warten, wenn die anderen WebSphere Partner Gateway-Threads diese Prozesse verwenden.

### **pool.max.keep.idle.count**

**Empfohlene Einstellung:** Keine bestimmte Einstellung; der Wert sollte jedoch hoch genug sein, dass nicht ständig Prozesse erstellt werden müssen, da dies die Leistung beeinträchtigt. Diese Einstellung ist ein Wert in Sekunden.

**Beschreibung:** Durch diese Eigenschaft wird gesteuert, wie lange ein Prozess inaktiv ist, bevor er erneut angefordert werden kann. Wird der Prozess erneut benötigt, obwohl der Wert von `pool.max.process.count` noch nicht erreicht wurde, wird er neu erstellt.

**pool.acquire.process.timeout**

**Empfohlene Einstellung:** 10 (Sekunden)

**Beschreibung:** Diese Eigenschaft gibt an, wie lange ein Aufruf für eine Zuordnung warten muss, um einen Prozess abzurufen, damit die Zuordnung ausgeführt werden kann. Wenn pool.max.process.count ordnungsgemäß konfiguriert ist, sollte das Zeitlimit keine Probleme aufwerfen.

**pool.map.auto.unload.timeout**

**Empfohlene Einstellung:** 120 (Sekunden)

**Beschreibung:** Der RMI-Server stellt Zuordnungen in den Cache. Diese Eigenschaft legt fest, wie lange eine Zuordnung im Speicherpool des RMI-Servers verbleibt, bevor die Zuordnung aus dem Pool entfernt wird. Für den Fall, dass die Zuordnung aus dem Pool entfernt und erneut benötigt wird, wird die Zuordnung aus dem Dateisystem gelesen. Da WebSphere Partner Gateway-Zuordnungen in Produktionsumgebungen häufig verwendet werden, ist es vorteilhaft, sie länger im Cache zu behalten. Die optimale Einstellung basiert auf Erfahrungswerten.

---

## Kapitel 14. Integration mit WebSphere Transformation Extender Trading Manager

In diesem Kapitel wird die Integration von WebSphere Partner Gateway mit WebSphere Transformation Extender Trading Manager beschrieben.

---

### Einführung

#### Was ist WebSphere Transformation Extender Trading Manager?

WebSphere Transformation Extender Trading Manager ist ein Client/Server-Produkt für die Verwaltung und Verarbeitung elektronischer Geschäftsdaten, das Funktionen für die Verwaltung und Steuerung der B2B-Integration von Partnerbeziehungen und Nachrichtenflüssen bereitstellt. Trading Manager ermöglicht dem Benutzer die Überprüfung, Steuerung, Überwachung und Anzeige der gesamten B2B-Integrationsumgebung im expandierenden Unternehmen, wobei die Funktion für sicheren Datenaustausch vollständig in die Back-End-Systeme integriert ist.

Weitere Informationen zu Trading Manager und den zugehörigen Komponenten finden Sie in der Trading Manager-Dokumentation.

#### Zusammenarbeit zwischen WebSphere Transformation Extender Trading Manager und WebSphere Partner Gateway

Die Integration mit Trading Manager erfolgt auf ähnliche Weise wie bei anderen Back-End-Anwendungen. Normalerweise wird Trading Manager für die Verarbeitung von EDI-Dokumenten verwendet. Beispiel: Ein AS2-Dokument mit EDI-Nutzdaten wird von einem externen Partner empfangen, und die EDI-Nutzdaten werden zur weiteren Verarbeitung an Trading Manager gesendet. Kompatible Schnittstellen zwischen WebSphere Partner Gateway (Ziele) und Trading Manager (Post Offices) sind Datei, HTTP und JMS.

Integrationsszenarios:

- Externer Partner → WebSphere Partner Gateway → Trading Manager (Interner Partner) → Back-End-Anwendung → Externer Partner
- WebSphere Partner Gateway → Trading Manager (Interner Partner) → Back-End-Anwendung

#### Relevante Dokumentation

Die für WebSphere Partner Gateway relevante Dokumentation ist das Handbuch *WebSphere Partner Gateway Hubkonfiguration*. Für Trading Manager ist es die Trading Manager-Dokumentation.

## Integration mit WebSphere Transformation Extender Trading Manager planen

WebSphere Partner Gateway Version 6.1 unterstützt die Integration mit WebSphere Transformation Extender Trading Manager Version 7.8 oder höher. Bei Verwendung der Dateischnittstelle für Trading Manager müssen WebSphere Partner Gateway und Trading Manager Zugriff auf das Dateisystemverzeichnis haben, das für den Dokumentaustausch verwendet wird.

## Gemeinsame Konfiguration bei WebSphere Partner Gateway und WebSphere Transformation Extender Trading Manager

Bestimmte Aspekte der Konfiguration haben WebSphere Partner Gateway und Trading Manager gemeinsam. Es folgt eine Querverweistabelle für die beiden Produkte.

*Tabelle 105. Konfigurationsbedingungen für WebSphere Transformation Extender Trading Manager*

WebSphere Partner Gateway	Trading Manager	Anmerkungen
Interner Partner	Internal Trading Partner	
Externer Partner	External Trading Partner	
Geschäfts-ID der Partner. Für EDI das Qualifikationsmerkmal und die ID für den EDI-Austausch.	Geschäfts-ID der Partner. Für EDI werden das Qualifikationsmerkmal und die ID für den EDI-Austausch separat konfiguriert.	
Ziel	Get Post Office	
Empfänger	Put Post Office	
Verbindung - Quelle des externen Partners zu Ziel des internen Partners	Inbound Trade Link	Es gibt keine exakte Eins-zu-eins-Beziehung, da in Abhängigkeit von der EDI-Gruppenkonfiguration mehrere eingehende Handelsverbindungen (Inbound Trade Link) vorhanden sein können.
Verbindung - Quelle des internen Partners zu Ziel des externen Partners	Outbound Trade Link	Es gibt keine exakte Eins-zu-eins-Beziehung, da in Abhängigkeit von der EDI-Gruppenkonfiguration mehrere ausgehende Handelsverbindungen (Outbound Trade Link) vorhanden sein können.

## WebSphere Partner Gateway konfigurieren

In diesem Abschnitt werden Elemente beschrieben, die sich auf die Schnittstelle von WebSphere Transformation Extender Trading Manager beziehen.

Dokumentenflüsse werden normal wie bei anderen Integrationen definiert. Die Partner müssen in WebSphere Partner Gateway und in Trading Manager definiert werden, obwohl die WebSphere Partner Gateway-Konfiguration auf die Elemente beschränkt werden kann, die für Pass-Through-Operationen von Dokumenten erforderlich sind.

Die folgenden Abschnitte enthalten besondere Hinweise zur Schnittstelle von Trading Manager hinsichtlich des internen Partners, der externen Partner, der Dokumentdefinition, der Interaktionen, der Verbindungen und des Empfängers.

Weitere Informationen zur Definition von Partnern, Zielen und Dokumentenflüssen finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

### Hinweise zum internen Partner

Es wird davon ausgegangen, dass der interne Partner für Trading Partner verwendet wird; er entspricht dem internen Handelspartner (Internal Trading Partner) in Trading Manager.

*Tabelle 106. Hinweise zur Konfiguration des internen Partners für WebSphere Transformation Extender Trading Manager*

Element	Besondere Hinweise
Geschäfts-ID	Die Geschäfts-ID, die den internen Partner im EDI-Austausch darstellt, muss konfiguriert werden. Wenn Trading Manager der Empfänger des Dokuments ist, ist dies das Qualifikationsmerkmal und die ID für den Empfänger des Austauschs. Wenn Trading Manager der Absender des Dokuments ist, ist dies das Qualifikationsmerkmal und die ID für den Absender des Austauschs. Weitere Details hierzu finden Sie in Kapitel 15, „EDI-Dokumente weiterleiten“, auf Seite 255.
Ziel	Transporttyp und -konfiguration entsprechen dem verwendeten Get Post Office von Trading Manager, wie z. B. Datei, HTTP oder JMS. <ul style="list-style-type: none"><li>• Bei Verwendung einer Datei benötigen WebSphere Partner Gateway und Trading Manager Zugriff auf das gemeinsam genutzte Verzeichnis.</li><li>• Bei Verwendung von JMS müssen WebSphere Partner Gateway und Trading Manager für die Verwendung derselben JMS-Warteschlangen konfiguriert werden.</li></ul>
B2B-Funktionalität	Die B2B-Funktionalität des internen Partners muss für die Quellen- und Zieldokumenttypen aktiviert werden, die mit Trading Manager verwendet werden. Für EDI-Dokumente gilt Folgendes: <ul style="list-style-type: none"><li>• Paket - None</li><li>• Protokoll - EDI-EDIFACT oder EDI-X12</li><li>• Dokumenttyp - UNB (EDIFACT) oder ISA (X12)</li></ul>

## Hinweise zum externen Partner

Der externe Partner entspricht dem externen Handelspartner (External Trading Partner) von Trading Manager.

*Tabelle 107. Hinweise zur Konfiguration des externen Partners für WebSphere Transformation Extender Trading Manager*

Element	Besondere Hinweise
Geschäfts-ID	Die Geschäfts-ID, die den externen Partner im EDI-Austausch darstellt, muss konfiguriert werden. Wenn Trading Manager der Empfänger des Dokuments ist, ist dies das Qualifikationsmerkmal und die ID für den Absender des Austauschs. Wenn Trading Manager der Absender des Dokuments ist, ist dies das Qualifikationsmerkmal und die ID für den Empfänger des Austauschs. Detaillierte Informationen hierzu finden Sie in Kapitel 15, „EDI-Dokumente weiterleiten“, auf Seite 255.
Ziel	Das Ziel des externen Partners muss für Trading Manager nicht speziell konfiguriert werden.
B2B-Funktionalität	Die B2B-Funktionalität des externen Partners muss für die Quellen- und Zieldokumenttypen aktiviert werden, die mit dem externen Partner verwendet werden, und ist nicht spezifisch für Trading Manager. Für EDI-Dokumente, die AS2-Pakete verwenden, gilt beispielsweise Folgendes: <ul style="list-style-type: none"><li>• Paket - AS</li><li>• Protokoll - EDI-EDIFACT oder EDI-X12</li><li>• Dokumenttyp - UNB (EDIFACT) oder ISA (X12)</li></ul>

## Hinweise zur Dokumentdefinition

Wenn Trading Manager der Absender oder Empfänger ist, muss für die Dokumentdefinition das Paket 'None' verwendet werden. Wenn es sich um ein EDI-Dokument handelt, gilt für die Dokumentdefinition Folgendes:

- Paket - None
- Protokoll - EDI-EDIFACT oder EDI-X12
- Dokumenttyp - UNB (EDIFACT) oder ISA (X12)

## Hinweise zu Interaktionen

Wenn Trading Manager der Empfänger ist, wird die im Abschnitt zur Dokumentdefinition beschriebene Zieldokumentdefinition verwendet. Wenn Trading Manager der Absender ist, wird die im Abschnitt zur Dokumentdefinition beschriebene Quellendokumentdefinition verwendet. Als Aktion wird 'Pass-Through' verwendet.

## Hinweise zu Verbindungen

Im Hinblick auf Verbindungen gibt es keine speziellen Konfigurationseinstellungen für Trading Manager.

## Hinweise zum Empfänger

Beim Empfänger entsprechen Transporttyp und -konfiguration dem verwendeten Put Post Office von Trading Manager, wie z. B. Datei, HTTP oder JMS.

- Bei Verwendung einer Datei benötigen WebSphere Partner Gateway und Trading Manager Zugriff auf das gemeinsam genutzte Verzeichnis.
- Bei Verwendung von JMS müssen WebSphere Partner Gateway und Trading Manager für die Verwendung derselben JMS-Warteschlangen konfiguriert werden.

## WebSphere Transformation Extender Trading Manager konfigurieren

Trading Manager wird normal konfiguriert. Obwohl die Partner in WebSphere Partner Gateway definiert sind, müssen sie auch in Trading Manager definiert werden. Die Partnernamen müssen in WebSphere Partner Gateway und Trading Manager zwar nicht identisch sein, ähnliche Benennungen vereinfachen jedoch die Verwaltung der beiden Produkte.

Besondere Elemente der WebSphere Partner Gateway-Schnittstelle:

*Tabelle 108. Konfigurationshinweise für WebSphere Transformation Extender Trading Manager*

Element	Besondere Hinweise
Internal Trading Partner	<p>Der interne Handelspartner (Internal Trading Partner) entspricht dem internen Partner in WebSphere Partner Gateway.</p> <p>Die Geschäfts-ID, die den internen Handelspartner im EDI-Austausch darstellt, muss konfiguriert werden. Wenn WebSphere Partner der Empfänger des Dokuments ist, ist dies das Qualifikationsmerkmal und die ID für die Quelle des Austauschs. Wenn WebSphere Partner Gateway der Absender des Dokuments ist, ist dies das Qualifikationsmerkmal und die ID für den Empfänger des Austauschs.</p>
External Trading Partner	<p>Der externe Handelspartner (External Trading Partner) entspricht dem externen Partner in WebSphere Partner Gateway.</p> <p>Die Geschäfts-ID, die den externen Handelspartner im EDI-Austausch darstellt, muss konfiguriert werden. Wenn WebSphere Partner Gateway der Empfänger des Dokuments ist, ist dies das Qualifikationsmerkmal und die ID für den Empfänger des Austauschs. Wenn WebSphere Partner Gateway der Absender des Dokuments ist, ist dies das Qualifikationsmerkmal und die ID für den Absender des Austauschs.</p>
Get Post Office	<p>Transporttyp und -konfiguration entsprechen dem Ziel des internen Partners von WebSphere Partner Gateway, das für die Schnittstelle mit Trading Manager verwendet wird, z. B. Datei, HTTP oder JMS.</p> <ul style="list-style-type: none"><li>• Bei Verwendung einer Datei benötigen WebSphere Partner Gateway und Trading Manager Zugriff auf das gemeinsam genutzte Verzeichnis.</li><li>• Bei Verwendung von JMS müssen WebSphere Partner Gateway und Trading Manager für die Verwendung derselben JMS-Warteschlangen konfiguriert werden.</li></ul>
Put Post Office	<p>Transporttyp und -konfiguration entsprechen dem verwendeten WebSphere Partner Gateway-Empfänger, wie z. B. Datei, HTTP oder JMS.</p> <ul style="list-style-type: none"><li>• Bei Verwendung einer Datei benötigen WebSphere Partner Gateway und Trading Manager Zugriff auf das gemeinsam genutzte Verzeichnis.</li><li>• Bei Verwendung von JMS müssen WebSphere Partner Gateway und Trading Manager für die Verwendung derselben JMS-Warteschlangen konfiguriert werden.</li></ul>
Inbound Trade Link	Keine spezielle Konfiguration.
Outbound Trade Link	Keine spezielle Konfiguration.

Weitere Konfigurationshinweise finden Sie in der WebSphere Transformation Extender Trading Manager-Dokumentation.



---

## **Teil 6. Andere geschäftsprotokollspezifische Informationen**

Die folgenden Kapitel enthalten geschäftsprotokollspezifische Informationen, wie beispielsweise Informationen zum Weiterleiten von EDI-Dokumenten, sowie Richtlinien, die bei der weiteren Verarbeitung von AS-Paketen, RosettaNet-Paketen und ebMS-Paketen befolgt werden müssen.



---

## Kapitel 15. EDI-Dokumente weiterleiten

In diesem Abschnitt wird das Verfahren beschrieben, mit dem WebSphere Partner Gateway die Informationen für das Routing gesendeter und empfangener EDI-Dokumente (EDI - Electronic Data Interchange) ermittelt. Der Abschnitt enthält eine Beschreibung des allgemeinen Ablaufs dieser Verarbeitung (siehe Übersicht über das EDI-Routing).

Zusätzliche Informationen zur Verwendung der dateibasierten Integration beim Routing von EDI-Dokumenten finden Sie im Abschnitt „Dateisystemprotokoll“ auf Seite 44. Besondere Hinweise zur AS-Paketierung finden Sie im Abschnitt Kapitel 16, „Besondere Hinweise zum AS-Paket“, auf Seite 257.

---

### Übersicht über das EDI-Routing

Ein EDI-Dokument enthält Informationen zum Absender und Empfänger des Dokuments. WebSphere Partner Gateway nutzt diese Informationen zum Routing des EDI-Dokuments. Der allgemeine Ablauf sieht wie folgt aus:

1. WebSphere Partner Gateway überprüft die ersten drei Zeichen des Dokuments und ermittelt damit das verwendete Protokoll. In Tabelle 109 wird dargestellt, welche Dokumenttypprotokolle den einzelnen Knoten zugeordnet sind.

*Tabelle 109. EDI-Codes und zugeordnete Dokumenttypen und -protokolle*

Code	Dokumenttyp	Dokumenttyp-protokoll	Inhaltstyp für ausgehende Dokumente:
ISA	X12	EDI-X12	application/EDI-X12
GS	X12	EDI-X12	application/EDI-X12
UNB	Edifact	EDI-EDIFACT	application/EDIFACT
UNA	Edifact	EDI-EDIFACT	application/EDIFACT
ICS	ICS	EDI-X12	application/EDI-X12
STX	UNTDI	EDI-Consent	application/edi-consent
BG	UCS	EDI-Consent	application/edi-consent

2. WebSphere Partner Gateway extrahiert aus dem EDI-Dokument die Absenderinformationen auf der Grundlage des Elements und der Position für diesen Dokumenttyp, wie in Tabelle 110 beschrieben.

*Tabelle 110. EDI-Codes und die Position der Absender- und Empfängerdaten*

Code	Absendermerkmal	Absender-ID	Empfängermerkmal	Empfänger-ID	Unterstützungshinweise zur EDI-Komponente
ISA	Element 105 an Position 5	Element 107 an Position 6	Element 105 an Position 7	Element 106 an Position 8	Unterstützt
GS	N/V	Element 142 an Position 2	N/V	Element 124 an Position 3	Reine GS-Adressierung (GS-only) wird nicht unterstützt

Tabelle 110. EDI-Codes und die Position der Absender- und Empfängerdaten (Forts.)

Code	Absendermerkmal	Absender-ID	Empfängermerkmal	Empfänger-ID	Unterstützungshinweise zur EDI-Komponente
UNB UNA	Subelement 0007 an Position 2 des zusammengesetzten Elements S002 an Position 20 (2. Teil) des UNB-Segments	Subelement 0004 an Position 2 des zusammengesetzten Elements S002 an Position 20 (2. Teil) des UNB-Segments	Subelement 0007 an Position 2 des zusammengesetzten Elements S003 an Position 30 (3. Teil) des UNB-Segments	Subelement 0010 an Position 1 des zusammengesetzten Elements S003 an Position 30 (3. Teil) des UNB-Segments	Unterstützt
ICS	Element X05 an Position 4	Element X06 an Position 5	Element X05 an Position 6	Element X08 an Position 7	Zwischenversion zu ISA - Nicht unterstützt
STX	Element FROM1 an Position 3	Element FROM2 an Position 3	Element UNT1 an Position 4	Element UNT2 an Position 4	Im aktuellen Release nicht unterstützt
BG	N/V	Element BG03 an Position 3	N/V	Element BG04 an Position 4	Unterstützt

3. WebSphere Partner Gateway ermittelt die Absender-ID aus der Absender-ID und dem Absenderqualifikationsmerkmal des EDI-Dokuments.  
Beachten Sie, dass einige EDI-Umschläge (z. B. GS) keine Qualifikationsmerkmale kennen. In diesem Fall verwendet WebSphere Partner Gateway nur die ID.
4. WebSphere Partner Gateway verkettet das Qualifikationsmerkmal und die ID mit einem Bindestrichzeichen (-), um die Absender-ID aus dem Profilrepository von WebSphere Partner Gateway herauszusuchen. Wenn zum Beispiel in der EDI-Nachricht für den Absender das Qualifikationsmerkmal 'AB' lautet und die ID den Wert 1234567 hat, erwartet WebSphere Partner Gateway, einen externen Partner mit der Kennung 'AB-1234567' im Profilrepository zu finden. Wenn WebSphere Partner Gateway diese ID nicht finden kann, wird das EDI-Dokument nicht weitergeleitet.
5. Zum Ermitteln des empfangenden Partners bestimmt WebSphere Partner Gateway das Qualifikationsmerkmal für den Empfänger und die ID aus der EDI-Nachricht.
6. WebSphere Partner Gateway verkettet das Qualifikationsmerkmal und die ID mit einem Bindestrichzeichen (-), um die Empfänger-ID aus dem Profilrepository herauszusuchen.
7. WebSphere Partner Gateway leitet das Dokument an den beabsichtigten Empfänger weiter.

---

## Kapitel 16. Besondere Hinweise zum AS-Paket

In diesem Abschnitt werden zusätzliche Verarbeitungsschritte beschrieben, die erforderlich sind, wenn ein AS-Paket angegeben wurde.

Wenn das zu erstellende Paket des Dokuments als 'AS-Paket' angegeben ist, führt WebSphere Partner Gateway einige zusätzliche Verarbeitungsschritte aus.

Informationen zu Back-End-Integrationspaketen und AS finden Sie im Abschnitt „Back-End-Integrationspaket“ auf Seite 26.

---

### Routing eingehender Dokumente

Ein Dokument wird von einem externen Partner empfangen:

1. WebSphere Partner Gateway prüft zuerst die AS1- bzw. AS2-Headerinformationen. Insbesondere werden die Absender- und Empfängerinformationen überprüft, um festzustellen, ob Übereinstimmungen mit IDs für gültige externe Partner vorhanden sind.
  - Bei AS1 wird das Headerfeld 'Subject' verwendet, das folgende Form hat:  
*EmpfängerID;AbsenderID*
  - Bei AS2 werden die Headerfelder 'AS2-Absender' (AS2-From) und 'AS2-Empfänger' (AS2-To) verwendet.

Wenn die Werte in den Headerfeldern keinen gültigen IDs entsprechen, leitet WebSphere Partner Gateway das Dokument nicht weiter.

2. WebSphere Partner Gateway führt dann wie gewohnt die Schritte für die Nutzdaten aus (d. h. Protokoll- und Dokumenttyp feststellen, Geschäfts-IDs extrahieren usw.). Weitere Informationen zu EDI-Nutzdaten finden Sie im Abschnitt „Übersicht über das EDI-Routing“ auf Seite 255.

---

### Routing ausgehender Dokumente

Wenn ein Dokument von einem Back-End-System empfangen wird, stellt WebSphere Partner Gateway fest, ob ein AS-Attribut für Geschäfts-ID (BusinessID) für das Quellenpaket (None) und das Empfängerpaket (AS) angegeben wurde.

- Wenn das AS-Attribut für Geschäfts-ID angegeben wurde, verwendet WebSphere Partner Gateway diese Informationen, um die Absender- und Empfänger-IDs im AS1- bzw. AS2-Header zu generieren.
- Wenn das Attribut nicht angegeben wurde, verwendet WebSphere Partner Gateway die Absender- und die Empfänger-ID aus dem Dokument. Für EDI werden die Informationen zur Absender- und Empfänger-ID- mit dem Qualifikationsmerkmal verkettet (siehe „Übersicht über das EDI-Routing“ auf Seite 255).

---

### Beide IDs im Partnerprofil festlegen

Da WebSphere Partner Gateway sowohl die AS1- bzw. AS2-Headerinformationen als auch die aus dem EDI-Dokument abgeleiteten Informationen verwendet, können die IDs für denselben Partner unterschiedliche Formate haben. Zum Beispiel könnten die AS-Headerinformationen für den Absender das Format '123456789' haben, während die aus dem EDI-Dokument abgeleiteten Informationen das Format 'AB-12345678' haben.

Stellen Sie sicher, dass Sie beide IDs im Profil für den externen Partner aufgeführt haben. Weitere Informationen finden Sie im Handbuch *WebSphere Partner Gateway Hubkonfiguration*.

---

## AS3-MDN-Verarbeitung

Wenn WebSphere Partner Gateway eine MDN (Message Disposition Notification) zurück an den Partner sendet, wird im Allgemeinen der Wert der IP-Adresse aus dem Headerattribut `Disposition-notification-to` (Dispositionbenachrichtigung an) des ursprünglichen AS3-Anforderungsdokuments zum Senden der MDN an den Partner verwendet. Das Verhalten hängt von dem Rückkehrziel ab, das für die Verbindung konfiguriert ist, die für das ursprüngliche AS3-Anforderungsdokument vom Partner verwendet wurde.

Für die Verbindung, die für das ursprüngliche AS3-Anforderungsdokument vom Partner verwendet wurde, gibt es eine Konfiguration für Rückkehrziele. Für diesen Konfigurationswert gilt Folgendes:

- **FTP-Ziel** - An Stelle des Werts, der für dieses FTP-Ziel konfiguriert wurde, wird der Wert der IP-Adresse des Headerattributs `"Disposition-notification-to"` aus dem ursprünglichen AS3-Anforderungsdokument verwendet.
- **FTP-Scripting-Ziel** - Der Wert der IP-Adresse des Headerattributs `"Disposition-notification-to"` aus dem ursprünglichen AS3-Anforderungsdokument wird nicht verwendet. Statt dessen wird der für das FTP-Scripting-Ziel konfigurierte Wert verwendet.
- **Anderer Transporttyp für das Ziel** - Das MDN-Dokument schlägt fehl.

---

## Weitere AS-Referenzen

Informationen zu Back-End-Integration und AS finden Sie im Abschnitt „Back-End-Integrationspaket“ auf Seite 26.

---

## Kapitel 17. Besondere Hinweise zum RosettaNet-Paket

---

### Weitere RosettaNet-Referenzen

Informationen zu Back-End-Integration und RosettaNet finden Sie in den folgenden Abschnitten:

- „RosettaNet“ auf Seite 16
- „Back-End-Integrationspaket“ auf Seite 26

Informationen zu WebSphere Process Server und RosettaNet finden Sie im Abschnitt „Integration mit WebSphere Process Server planen“ auf Seite 61.



---

## Kapitel 18. Besondere Hinweise zum ebMS-Paket

---

### Weitere ebMS-Referenzen

Informationen zu Back-End-Integration und ebMS finden Sie in den folgenden Abschnitten:

- „ebMS“ auf Seite 18
- „Back-End-Integrationspaket“ auf Seite 26



---

## Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden.

Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim zuständigen IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. An Stelle der IBM Produkte, Programme oder Services können auch andere, ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder andere Schutzrechte von IBM verletzen. Die Verantwortung für den Betrieb von Produkten, Programmen und Services anderer Anbieter liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

*IBM Director of Licensing  
IBM Europe, Middle East & Africa  
Tour Descartes  
2, avenue Gambetta  
92066 Paris La Defense  
France*

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die Angaben in diesem Handbuch werden in regelmäßigen Zeitabständen aktualisiert. Die Änderungen werden in Überarbeitungen oder in Technical News Letters (TNLs) bekannt gegeben. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter werden lediglich als Service für den Kunden bereitgestellt und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängig voneinander erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

IBM Burlingame Laboratory Director  
IBM Burlingame Laboratory  
577 Airport Blvd., Suite 800  
Burlingame, CA 94010  
USA

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des im Dokument aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung bzw. der Allgemeinen Geschäftsbedingungen von IBM, der IBM Internationalen Nutzungsbedingungen für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer kontrollierten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Gewährleistung, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können davon abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Die oben genannten Erklärungen bezüglich der Produktstrategien und Absichtserklärungen von IBM stellen die gegenwärtige Absicht von IBM dar, unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele von IBM.

Alle von IBM angegebenen Preise sind empfohlene Richtpreise und können jederzeit ohne weitere Mitteilung geändert werden. Händlerpreise können u. U. von den hier genannten Preisen abweichen.

Diese Veröffentlichung dient nur zu Planungszwecken. Die in dieser Veröffentlichung enthaltenen Informationen können geändert werden, bevor die beschriebenen Produkte verfügbar sind.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufes. Sie sollen nur die Funktionen des Lizenzprogramms illustrieren; sie können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden; Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

## COPYRIGHTLIZENZ

Diese Veröffentlichung enthält Musteranwendungsprogramme, die in Quellsprache geschrieben sind. Sie dürfen diese Musterprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle konform sind, für die diese Musterprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten.

Kopien oder Teile der Musterprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

Copyright (c) 1995-2008 International Business Machines Corporation und andere. Alle Rechte vorbehalten.

---

## Informationen zu Programmierschnittstellen

Die ggf. bereitgestellten Informationen zu Programmierschnittstellen sollen Ihnen bei der Erstellung von Anwendungssoftware unter Verwendung dieses Programms helfen. Mit allgemeinen Programmierschnittstellen können Sie Anwendungssoftware schreiben, die die Services aus den Tools dieses Programms abrufen. Diese Informationen können jedoch auch Angaben über Diagnose, Bearbeitung und Optimierung enthalten. Die Informationen zu Diagnose, Bearbeitung und Optimierung sollten Ihnen bei der Fehlerbehebung für die Anwendungssoftware helfen.

**Achtung:** Verwenden Sie diese Informationen zu Diagnose, Bearbeitung und Optimierung nicht als Programmierschnittstelle, da Änderungen vorbehalten sind.

---

## Marken und Servicemarken

Folgende Namen sind Marken oder eingetragene Marken der IBM Corporation in den USA und/oder anderen Ländern:

IBM	DB2	IMS	MQIntegrator	Tivoli
Das IBM Logo	DB2 Universal Database	Informix	MVS	WebSphere
AIX	Domino	iSeries	OS/400	z/OS
CICS	IBMLink	Lotus	Passport Advantage	
CrossWorlds	i5/OS	Lotus Notes	SupportPac	

Microsoft, Windows, Windows NT und das Windows-Logo sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

MMX, Pentium und ProShare sind Marken oder eingetragene Marken der Intel Corporation in den USA und/oder anderen Ländern.

Solaris, Java und alle auf Java basierenden Marken und Logos sind Marken von Sun Microsystems, Inc. in den USA und/oder anderen Ländern.

Linux ist eine Marke von Linus Torvalds in den USA und/oder anderen Ländern.

Weitere Unternehmens-, Produkt- oder Servicenamen können Marken anderer Hersteller sein.

WebSphere Partner Gateway Enterprise Edition und Advanced Edition enthalten Software, die vom Eclipse Project ([www.eclipse.org](http://www.eclipse.org)) entwickelt wurde.



# Index

## Numerische Stichwörter

0A1-Nachrichten 17

## A

Adapter für HTTP  
  an Collaboration binden 168  
  Geschäftsobjektstruktur 161  
  installieren 157  
  konfigurieren 157  
  Payload-Data-Handler 157  
  Protokollhandler 158  
Adapter für JMS  
  an Collaboration binden 187  
  Eingabewarteschlange definieren 181  
  Geschäftsobjektstruktur 182  
  konfigurieren 180  
  Payload-Data-Handler 180  
Adapter für Web-Services 169  
Anhänge  
  Beschreibung 35  
  content-type 35  
  Data-Handler für 133  
  Encoding 35  
  InterChange Server 147  
  Tag 35  
  Tagattribute 35  
  WebSphere Process Server 71  
Anhangscontainer, Geschäftsobjekt  
  InterChange Server 149  
  WebSphere Process Server 71  
Anzahl Threads, Ziel 46  
AS-Paket  
  besondere Hinweise 257  
AS2-Dokumente und Header der  
  Transportebene 30, 31  
Attachment-Data-Handler  
  Anhänge darstellen 148  
  Einführung 133  
  Geschäftsobjektdefinitionen 145  
  konfigurieren 141  
  Position 140  
  Repository-Datei 140  
Austausch, EDI 11  
Authentifizierung für Web-Services 111, 112

## B

B2B-Funktionalität  
  Beschreibung 6  
  einstellen 50, 54  
Back-End-Dokumente 53  
Back-End-Integration, Übersicht 3  
Back-End-System  
  Dokumente empfangen von 52  
  Dokumente senden an 48

Backend Integration, Paket  
  Beispiel 36  
  Beschreibung 26  
  Datenbindung 66  
  Einsatzmöglichkeiten 36, 40, 42  
  HTTP-Transportprotokoll 40  
  JMS-Transportprotokoll 42  
  Transportprotokolle 37  
  Umschlagsmarkierung 34  
BCGBackendIntegrationDataBindingUtil,  
  Klasse  
  getAsByteArray, Methode 80  
  getAsString, Methode 81  
  getDataObject, Methode 81  
  getxAuxHeader, Methode 81  
  Methoden 80  
  Mustercode 85  
  read, Methode 81  
  setBOPrefix, Methode 81  
  setDataObject, Methode 82  
  setDebugLevel, Methode 82  
  setFromByteArray, Methode 82  
  setFromString, Methode 82  
  setOptions, Methode 83  
  setPackagingSchema, Methode 83  
  setTLOTypeName, Methode 84  
  setxAuxHeader, Methode 84  
  Übersicht 79  
  write, Methode 84  
BCGBackendIntegrationJMSData-  
  BindingImpl, Klasse  
  getMessageType, Methode 85  
  isBusinessException, Methode 86  
  Methoden 85  
  read, Methode 86  
  setBusinessException, Methode 86  
  Übersicht 85  
  write, Methode 86  
Benutzerdefinierte Eigenschaften,  
  Geschäftsobjekte 166  
Binärdokumente  
  Paketeinschränkungen 40, 42  
  unterstützte Transportprotokolle 38  
BusinessObjectID, Feld 24

## C

Collaborations  
  Adapter für HTTP 168  
  Adapter für JMS 187  
  Beschreibung 132  
Compute-Knoten 200, 202, 208, 209  
content-length, Attribut 28  
content-type, Attribut 28, 35  
cXML-Dokumente  
  Beschreibung 10  
  Data-Handler für 130  
  Paketerstellung erforderlich 40  
  und WebSphere Process Server 90, 93  
  unterstützte Transportprotokolle 38

## D

Data-Handler  
  Anhang 133  
  Metaobjekt der höchsten Ebene 144  
  untergeordnetes Metaobjekt 141  
Data Interchange Services-Client 5, 11  
Dateiverzeichnis-Transportprotokoll  
  Beschreibung 44  
  und WebSphere Process Server 115  
  Verzeichnisstruktur 44  
Datenbindung  
  Backend Integration, Paket 66  
  HTTP-Transport 67  
  JMS-Transport 66  
  Übersicht 66  
Dokumente  
  AS1 31  
  AS2 30  
  cXML 10, 90, 93  
  ebMS 18, 32  
  EDI 255  
  RosettaNet 16, 29  
  SOAP 9, 109, 169, 201  
  XML-Ereignis (XMLEvent) 21  
Dokumentenflussdefinition  
  Beschreibung 5  
  definieren 49  
Dokumentenverzeichnis 44  
Dokumentverarbeitung, Übersicht 3  
Dynamische Metaobjekte  
  HTTP 166  
  JMS 185

## E

ebMS-Dokumente  
  Pakettyp 38  
  Service-Content 18  
  Übersicht 18  
  unterstützte Transportprotokolle 38  
ebMS-Dokumente und Header der  
  Transportebene 32  
ebMS-Paket  
  besondere Hinweise 261  
ebMS-Service-Content  
  Beschreibung 18  
EDI-Dokumente  
  Paketoptionen 40  
  Position der Nutzdaten 34  
  und WebSphere Data Inter-  
  change 212  
  unterstützte Transportprotokolle 39  
  weiterleiten 255  
Empfänger  
  Beschreibung 4  
  definieren 53  
Empfängerpartner-Ziel, auf dem Web-  
  Service 110  
Encoding, Attribut 35  
Endpunkt-URL, ändern 112

Ereignisbenachrichtigung 16, 20, 21, 22  
Ereignisbenachrichtigungen  
  Beispiel 24  
  Felder 24  
EventMessageID, Feld 24  
Exportbindung  
  Beschreibung 60  
  JMS 66, 97  
  SCA 89  
  Web-Service 109

## G

Geistiges Eigentum 263  
Geschäftsobjektdefinitionen, InterChange Server  
  Attachment-Data-Handler 145  
  erstellen 128  
  HTTP 160  
  JMS 182  
Geschäftsobjekte  
  InterChange Server  
    Anhang 149  
    benutzerdefinierte Eigenschaften 166  
    dynamisches Metaobjekt 166, 185  
    höchste Ebene 162  
    JMS-Eigenschaften 184  
    Metaobjekt für HTTP-Konfiguration 166  
    Nutzdaten 129, 147, 150  
  WebSphere Process Server  
    Anhangscontainer 71  
    höchste Ebene 68  
    Nutzdatencontainer 69  
    Paketheader 71  
Geschäftsprotokoll 9  
getAsByteArray, Methode 77, 80  
getAsString, Methode 77, 81  
getDataObject, Methode 81  
getMethodType, Methode 85  
getxAuxHeader, Methode 81  
Globale eindeutige Kennung (GUID - Global Unique Identifier) 47  
GlobalMessageID, Feld 24  
Gruppen, EDI 11

## H

Header der Transportebene  
  InterChange Server 165  
Höchste Ebene, Geschäftsobjekte  
  InterChange Server 162  
  WebSphere Process Server 68  
HTTP-Protokollhandler 158  
HTTP-Protokollkonfiguration, Metaobjekt 166  
HTTP-Transportprotokoll  
  Beschreibung 40  
  InterChange Server und 123, 153, 169  
  Position der Nutzdaten 34  
  Servlet für WebSphere Process Server 89, 94  
  WebSphere Message Broker 192, 197  
  WebSphere Process Server und 89  
HTTPInput-Knoten 200, 202

HTTPReply-Knoten 200, 202  
HTTPRequest-Knoten 200, 202  
Hubadministrator 4

## I

Importbindung  
  Beschreibung 61  
  JMS 66, 99  
  SCA 92  
  Web-Services 110  
Interaktionen  
  Beschreibung 6  
  erstellen 51, 54  
  zum Senden an das Back-End-System 51  
InterChange Server  
  Anforderungsverarbeitung 157  
  Artefakte erstellen 167, 186  
  Einführung 121  
  erforderliche Komponenten für Empfang  
    über HTTP 154  
    über JMS 173  
  erforderliche Komponenten für Versand  
    über HTTP 154  
    über JMS 173  
  HTTP-Transportprotokoll 153, 169  
  Integration planen 122  
  integrieren mit 173  
  JMS-Transportprotokoll 173  
  konfigurieren 128, 153, 167, 186  
  Muster für Integration 124  
  unterstützte Transportprotokolle 123  
  unterstützte Versionen 122  
Interner Partner 4  
isBusinessException, Methode 86

## J

JMS-Eigenschaften, Geschäftsobjekte 184  
JMS-Transportprotokoll  
  Beschreibung 41  
  Headerinformationen erstellen 183  
  InterChange Server-Geschäftsobjektstruktur 182  
  InterChange Server und 124, 173  
  Position der Nutzdaten 34  
  Warteschlangenmanager 43  
  WebSphere Message Broker 193, 203  
  WebSphere Process Server und 97

## L

Lizenz, Patente 263  
Lizenzierung  
  Adresse 263

## M

Methoden  
  getAsByteArray 77, 80  
  getAsString 77, 81  
  getDataObject 81

Methoden (*Forts.*)

  getMessageType 85  
  getxAuxHeader 81  
  isBusinessException 86  
  read 73, 81, 86  
  setBOPrefix 74, 81  
  setBusinessException 86  
  setDataObject 82  
  setDebugLevel 82  
  setFromByteArray 73, 82  
  setFromString 73, 82  
  setOptions 78, 83  
  setPackagingSchema 78, 83  
  setTLOTypeName 74, 84  
  setxAuxHeader 75, 84  
  write 77, 84, 86  
MO\_DataHandler\_Default, Metaobjekt der höchsten Ebene 144  
MQInput-Knoten 200, 208, 209  
MQOutput-Knoten 200, 208, 209

## N

Nachrichtenfluss  
  Beschreibung 197  
  für HTTP-Transport erstellen 199  
  für JMS-Transport erstellen 208  
None, Paket  
  Beschreibung 26  
  Einsatzmöglichkeiten 36, 40, 42  
  HTTP-Transportprotokoll 40  
Nutzdaten  
  Beschreibung 34  
  content-type 35  
  Encoding 35  
  Ereignisbenachrichtigung 23  
  Tag 35  
  Tagattribute 35  
Nutzdaten, Geschäftsobjekte  
  InterChange Server 150, 161, 182  
  WebSphere Process Server 69  
Nutzdatencontainer, Geschäftsobjekt 69

## O

Objekt der höchsten Ebene  
  Attribute 68  
  Beschreibung 68  
  Standard 74  
Online/Offline, Ziel 46

## P

Paketheader, Geschäftsobjekt 71  
Pakettyp  
  AS 257  
  Backend Integration 26  
  Beschreibung 5, 25  
  ebMS 261  
  None 26  
  RosettaNet 259  
Patente 263  
Payload-Data-Handler  
  Adapter für HTTP 157  
  Adapter für JMS 180

## R

- read, Methode 73, 81, 86
- ROD/flat-Dokumente
  - Position der Nutzdaten 34
  - unterstützte Transportprotokolle 38
- RosettaNet-Dokumente
  - 0A1-Nachrichten 17
  - als Quelle für Header der Transportebene 29
  - InterChange Server-Integrationsmuster 124
  - Pakettyp 38, 40
  - Position der Nutzdaten 34
  - Service-Content 16
  - Übersicht 16
  - unterstützte Transportprotokolle 38
  - WebSphere Process Server-Integrationsmuster 63
- RosettaNet-Paket
  - besondere Hinweise 259
- RosettaNet Service Content
  - Beschreibung 16
  - Geschäftsobjekte erstellen 72

## S

- Schema, XML
  - Nutzdaten der Ereignisbenachrichtigung 23
  - Transportumschläge 35
- Servlet für WebSphere Process Server, HTTP 89, 94
- setBOPrefix, Methode 74, 81
- setBusinessException, Methode 86
- setDataObject, Methode 82
- setDebugLevel, Methode 82
- setFromArray, Methode 73, 82
- setFromString, Methode 73, 82
- setOptions, Methode 78, 83
- setPackagingSchema, Methode 78, 83
- setTLOType, Methode 74, 84
- setxAuxHeader, Methode 75, 84
- SOAP-Dokumente
  - Einführung 9
  - InterChange Server über HTTP 169
  - Paketerstellung erforderlich 40
  - unterstützte Transportprotokolle 38
  - WebSphere Message Broker über HTTP 201
  - WebSphere Process Server über HTTP 109
- StatusCode, Feld 24
- StatusMessage, Feld 24

## T

- Tags
  - Anhang 35
  - Nutzdaten 35
  - Transportumschlag 35
- Timestamp, Feld 24
- Transaktionen, EDI 11
- Transportebene, Header
  - AS1-Quelle 31
  - AS2-Quelle 30
  - ebMS-Quelle 32

- Transportebene, Header (*Forts.*)
  - Inhalt 26
  - InterChange Server 183
  - RosettaNet-Quelle 29
  - WebSphere Process Server 71
- Transportprotokolle
  - Backend Integration, Paket 37
  - erforderlich für XMLevent 21
  - für ebMS erforderliche 18
  - für RosettaNet erforderliche 16
  - Liste 37
- Transportumschläge
  - Tag 35
  - und Anhänge 35
  - XML-Schema 35

## U

- Umschlagsmarkierung 34

## V

- Verarbeitung doppelter Nachrichten 47

## W

- Warteschlangen, Nachrichtenübermittlung 46
- Warteschlangenmanager 43
- Web-Service, Exportbindung 109
- Web-Services
  - Authentifizierung 111, 112
  - Empfängerpartner, Spezifikation 10, 110
  - Endpunkt-URL, ändern 112
  - und WebSphere Process Server 109
  - vom internen Partner bereitgestellt 9, 109
  - von Partnern bereitgestellt 10, 110
- Web-Services, Importbindung 110
- WebSphere Data Interchange 211
- WebSphere Message Broker
  - Einführung 191
  - erforderliche Komponenten für Empfang
    - über HTTP 197
    - über JMS 203
  - erforderliche Komponenten für Versand
    - über HTTP 197
    - über JMS 203
  - HTTP-Transportprotokoll 197
  - Integration planen 192
  - JMS-Transportprotokoll 203
  - konfigurieren 196
  - Muster für Integration 193
  - unterstützte Transportprotokolle 192
  - unterstützte Versionen 192
- WebSphere Partner Gateway
  - Dokumente empfangen von 52
  - Dokumente senden von 48
  - Installation mit WebSphere Process Server 61
  - Integration
    - mit InterChange Server 121

- WebSphere Partner Gateway (*Forts.*)
  - Integration (*Forts.*)

- mit WebSphere Data Interchange 211
  - mit WebSphere Message Broker 191
  - mit WebSphere Process Server 59
  - mit WebSphere Transformation Extender 237
  - mit WebSphere Transformation Extender Trading Manager 247
  - konfigurieren 47
  - WebSphere Partner Gateway - Express 4
  - WebSphere Process Server
    - Anhangscontainer, Geschäftsobjekt 71
    - Back-End-Integrationspaket, Datenbindung 66
    - Dateiverzeichnis-Transportprotokoll 115
    - Datenbindung 66
    - HTTP-Transportprotokoll 89
    - Integration planen 61
    - JMS-Transportprotokoll 97
    - Muster für Integration 63
    - Nutzdatencontainer, Geschäftsobjekt 69
    - Objekt der höchsten Ebene 68
    - Paketheader, Geschäftsobjekt 71
    - Übersicht 59
    - unterstützte Versionen 61
    - Web-Services 109
  - WebSphere Transformation Extender 237
  - WebSphere Transformation Extender Trading Manager 247
  - Wiederholungsintervall, Ziel 46
  - Wiederholungszähler, Ziel 46
  - write, Methode 77, 84, 86
- ## X
- x-aux-create-datetime 27
  - x-aux-event-status-code 27
  - x-aux-in-file-name 27
  - x-aux-IntelligibleCheckRequired 28
    - ebMS-Quelle 34
  - x-aux-msg-id
    - AS1-Quelle 32
    - AS2-Quelle 31
    - Beschreibung 27
    - ebMS-Quelle 33
    - RosettaNet-Quelle 30
  - x-aux-payload-root-tag
    - AS1-Quelle 32
    - AS2-Quelle 31
    - Beschreibung 27
    - RosettaNet-Quelle 29
  - x-aux-process-instance-id
    - Beschreibung 27
    - ebMS-Quelle 33
    - RosettaNet-Quelle 30
  - x-aux-process-type
    - AS1-Quelle 32
    - AS2-Quelle 31
    - Beschreibung 27
    - ebMS-Quelle 33
    - RosettaNet-Quelle 29

- x-aux-process-version
  - AS1-Quelle 32
  - AS2-Quelle 31
  - Beschreibung 27
  - ebMS-Quelle 33
  - RosettaNet-Quelle 29
- x-aux-production
  - Beschreibung 27
  - RosettaNet-Quelle 30
- x-aux-protocol
  - AS1-Quelle 32
  - AS2-Quelle 30
  - Beschreibung 27
  - ebMS-Quelle 33
  - RosettaNet-Quelle 29
- x-aux-protocol-version
  - AS1-Quelle 32
  - AS2-Quelle 30
  - Beschreibung 27
  - ebMS-Quelle 33
  - RosettaNet-Quelle 29
- x-aux-receiver-id
  - AS1-Quelle 31
  - AS2-Quelle 30
  - Beschreibung 26
  - ebMS-Quelle 32
  - RosettaNet-Quelle 29
- x-aux-sender-id
  - AS1-Quelle 31
  - AS2-Quelle 30
  - Beschreibung 26
  - ebMS-Quelle 33
  - RosettaNet-Quelle 29
- x-aux-SyncResponse 28
  - ebMS-Quelle 33
- x-aux-system-msg-id
  - AS1-Quelle 32
  - AS2-Quelle 31
  - Beschreibung 27
  - ebMS-Quelle 33
- x-aux-third-party-bus-id 27
  - ebMS-Quelle 33
- x-aux-TimeToAccept 28
  - ebMS-Quelle 33
- x-aux-transport-retry-count 27
- x-out-filename 28
- XML-Dokumente
  - Paketoptionen 40
  - Position der Nutzdaten 34
  - unterstützte Transportprotokolle 38
- XML-Schema
  - Nutzdaten der Ereignis-  
benachrichtigung 23
  - Transportumschlag 35
- XMLEvent-Dokumente
  - Service-Content 21
  - Übersicht 21
- XMLEvent-Service-Content
  - Beschreibung 21

## Z

- Ziele
  - Beschreibung 5, 49
  - definieren 49



**IBM**