

IBM WebSphere Partner Gateway Enterprise et Advanced  
Editions



# Guide d'intégration entreprise

*Version 6.1.1*



IBM WebSphere Partner Gateway Enterprise et Advanced  
Editions



# Guide d'intégration entreprise

*Version 6.1.1*

**Important**

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section «Remarques», à la page 257.

**Troisième édition - mars 2008**

LE PRESENT DOCUMENT EST LIVRE EN L'ETAT SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE.

Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. Les informations qui y sont fournies sont susceptibles d'être modifiées avant que les produits décrits ne deviennent eux-mêmes disponibles. En outre, il peut contenir des informations ou des références concernant certains produits, logiciels ou services non annoncés dans ce pays. Cela ne signifie cependant pas qu'ils y seront annoncés.

Pour plus de détails, pour toute demande d'ordre technique, ou pour obtenir des exemplaires de documents IBM, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial.

Vous pouvez également consulter les serveurs Internet suivants :

- <http://www.fr.ibm.com> (serveur IBM en France)
- <http://www.can.ibm.com> (serveur IBM au Canada)
- <http://www.ibm.com> (serveur IBM aux Etats-Unis)

*Compagnie IBM France  
Direction Qualité  
Tour Descartes  
92066 Paris-La Défense Cedex 50*

© Copyright IBM France 2008. Tous droits réservés.

© **Copyright International Business Machines Corporation 2004, 2008. All rights reserved.**

---

# Table des matières

<b>A propos de ce guide</b> . . . . .	<b>ix</b>
Public concerné . . . . .	ix
Conventions typographiques . . . . .	ix
Documents associés . . . . .	x
Nouveautés de cette version . . . . .	xi
Nouveautés de l'édition 6.1.1 . . . . .	xi
Nouveautés de la version 6.1. . . . .	xi

---

## Partie 1. Présentation de l'intégration dorsale . . . . . 1

### Chapitre 1. Présentation générale. . . . . 3

Présentation du traitement des documents . . . . .	3
Rôles dans la communauté du concentrateur . . . . .	4
Processus de configuration du concentrateur WebSphere Partner Gateway. . . . .	4
Présentation de l'intégration dorsale . . . . .	7

### Chapitre 2. Planification de l'intégration dorsale . . . . . 9

Quel est le protocole de gestion utilisé ? . . . . .	9
Services Web (SOAP) . . . . .	9
cXML . . . . .	10
EDI . . . . .	10
RosettaNet. . . . .	16
ebMS . . . . .	17
XMLEvent. . . . .	21
Quel emballage allez-vous utiliser ? . . . . .	26
Aucun emballage . . . . .	26
Regroupement d'intégration dorsale . . . . .	27
Quel type d'emballage peut être utilisé pour vos documents ? . . . . .	36
Exemple d'emballage Backend Integration sur HTTP . . . . .	36
Quel mode de transport de messages allez-vous utiliser ? . . . . .	37
Protocole de transport HTTP . . . . .	40
Protocole JMS . . . . .	41
Protocole de système de fichiers . . . . .	44
Comment accéder à l'application dorsale? . . . . .	45
Traitement des messages . . . . .	46
Distribution en file d'attente. . . . .	46
Gestion des erreurs de transmission . . . . .	46
Messages en double . . . . .	47
Configuration de WebSphere Partner Gateway. . . . .	47
Envoi de documents au système dorsal . . . . .	48
Réception de documents en provenance du système dorsal . . . . .	52

---

## Partie 2. Intégration à WebSphere Process Server . . . . . 57

### Chapitre 3. Présentation de l'intégration à WebSphere Process Server . . . . . 59

Présentation . . . . .	59
Fonctionnement de la communication entre WebSphere Process Server et WebSphere Partner Gateway. . . . .	60
Planification de l'intégration à WebSphere Process Server . . . . .	61
Versions de WebSphere Process Server prises en charge par WebSphere Partner Gateway . . . . .	61
Scénarios d'installation pris en charge . . . . .	62
Transports de messages pris en charge par WebSphere Process Server . . . . .	62
Prise en charge de l'intégration à WebSphere Process Server . . . . .	63
Présentation des tâches d'intégration de WebSphere Partner Gateway à WebSphere Process Server . . . . .	64
Sur le système WebSphere Partner Gateway . . . . .	64

Sur le système WebSphere Process Server . . . . .	65
Gestion des messages avec empaquetage Backend Integration . . . . .	66
Transport JMS . . . . .	67
Transport HTTP . . . . .	67
Objets métier enfant et de niveau supérieur . . . . .	68
Fonctionnement de la liaison de données de l’empaquetage Backend Integration . . . . .	72
Classe BCGBackEndIntegrationDataBindingUtil . . . . .	78
Interface DataBinding . . . . .	78
Méthodes . . . . .	78
Exemple de code . . . . .	84
Classe BCGBackEndIntegrationJMSDataBindingImpl . . . . .	84
Interface JMSDataBinding . . . . .	84
Méthodes . . . . .	84
Insertion de classes de liaison de données dans l’implémentation de votre composant . . . . .	85

## **Chapitre 4. Intégration de WebSphere Process Server avec HTTP comme protocole de transport . . . . . 87**

Fonctionnement de l’envoi de messages vers WebSphere Process Server . . . . .	87
Fonctionnement de l’envoi de messages depuis WebSphere Process Server . . . . .	89
Envoi de documents vers WebSphere Process Server. . . . .	91
Configuration de WebSphere Partner Gateway. . . . .	91
Configuration de WebSphere Process Server . . . . .	92
Envoi de documents depuis WebSphere Process Server . . . . .	93
Configuration de WebSphere Partner Gateway. . . . .	93
Configuration de WebSphere Process Server . . . . .	93

## **Chapitre 5. Intégration de WebSphere Process Server avec le protocole de transport JMS . . . . . 95**

Envoi de documents via le protocole de transport JMS . . . . .	95
Réception de documents via le protocole de transport JMS . . . . .	97
Configuration de l’environnement WebSphere Process Server sur WebSphere Application Server . . . . .	98
Création d’un service SCA avec WSDL . . . . .	98
Personnalisation des liaisons d’importation et d’exportation JMS . . . . .	98
Implémentation de la liaison de données JMS . . . . .	99
Personnalisation d’un sélecteur de fonctions . . . . .	100
Configuration de JMS avec WebSphere Partner Gateway installé sur WebSphere Application Server . . . . .	101
Création et configuration des bus, files d’attente JMS et fabriques de connexions . . . . .	102
Création du récepteur JMS . . . . .	105
Création de la destination JMS . . . . .	105
Création d’une file d’attente de destination . . . . .	106

## **Chapitre 6. Intégration de WebSphere Process Server à SOAP/HTTP . . . . . 107**

Description de l’échange des messages SOAP via le protocole HTTP . . . . .	107
De quelle manière les services Web hébergés par WebSphere Process Server sont-ils appelés . . . . .	108
De quelle manière les services Web hébergés par les partenaires sont-ils appelés . . . . .	109
Appel de services Web hébergés par WebSphere Process Server . . . . .	110
Configuration de WebSphere Partner Gateway . . . . .	110
Configuration de WebSphere Process Server . . . . .	110
Appel de services Web hébergés par les partenaires externes . . . . .	110
Configuration de WebSphere Partner Gateway . . . . .	111
Configuration de WebSphere Process Server . . . . .	111

## **Chapitre 7. Intégration de WebSphere Process Server avec le système de fichiers comme protocole de transport . . . . . 113**

Envoi de documents via le protocole système de fichiers . . . . .	113
Réception de documents via le protocole système de fichiers . . . . .	113
Configuration de l’environnement WebSphere Process Server . . . . .	114
Déploiement et configuration de WebSphere Adapter for Flat Files . . . . .	114
Création de composants SCA . . . . .	115

Configuration de l'environnement WebSphere Partner Gateway . . . . .	115
--	-----

---

## **Partie 3. Intégration à WebSphere InterChange Server . . . . . 117**

### **Chapitre 8. Présentation de l'intégration à InterChange Server . . . . . 119**

Planification de l'intégration à InterChange Server . . . . .	120
Versions d'InterChange Server prises en charge par WebSphere Partner Gateway . . . . .	120
Transports de messages pris en charge par InterChange Server . . . . .	121
Prise en charge de l'intégration à InterChange Server . . . . .	122
Configuration de WebSphere Partner Gateway pour InterChange Server . . . . .	123
Prise en charge des documents sortants . . . . .	123
Prise en charge des documents entrants . . . . .	125
Configuration d'InterChange Server . . . . .	127
Création de définitions d'objets métier . . . . .	127
Création des connecteurs . . . . .	130
Création des collaborations . . . . .	131
Déploiement du projet . . . . .	131
Gestion des documents contenant des pièces jointes . . . . .	132
Exécution de la conversion par le gestionnaire de données de pièces jointes . . . . .	133
Configuration de l'environnement pour le gestionnaire de données de pièces jointes . . . . .	138
configuration du gestionnaire de données de pièces jointes . . . . .	139
Création de définitions d'objets métier liées aux pièces jointes . . . . .	144

### **Chapitre 9. Intégration d'InterChange Server via HTTP. . . . . 151**

Utilisation du protocole de transport HTTP avec ICS . . . . .	151
Composants requis pour les documents envoyés à ICS via HTTP . . . . .	151
Configuration de l'environnement pour le transport HTTP avec ICS . . . . .	155
Création de définitions d'objet métier pour ICS via HTTP . . . . .	157
Création d'artefacts ICS pour HTTP . . . . .	164
Envoi de documents SOAP via HTTP/S . . . . .	165
Composants requis pour l'envoi et la réception de documents . . . . .	166
Fonctionnement de l'appel des services Web par les partenaires externes . . . . .	166
Fonctionnement de l'appel des services Web par le partenaire interne . . . . .	167

### **Chapitre 10. Intégration à InterChange Server via JMS. . . . . 169**

Composants requis pour les documents envoyés via JMS. . . . .	169
Envoi de documents via le transport JMS . . . . .	170
Réception de documents via le transport JMS . . . . .	172
Configuration de l'environnement requis pour le transport JMS . . . . .	175
Configuration des files d'attente JMS . . . . .	175
Configuration d'Adapter for JMS. . . . .	176
Création de définitions d'objet métier pour JMS . . . . .	177
Création d'une structure objet métier données utiles pour JMS . . . . .	178
Création des informations d'en-tête JMS . . . . .	179
Création d'artefacts ICS pour JMS . . . . .	182
Création de l'objet connecteur JMS . . . . .	182
Mise en place de liens de collaboration en vue d'une communication avec Adapter for JMS . . . . .	183

---

## **Partie 4. Intégration à d'autres systèmes dorsaux . . . . . 185**

### **Chapitre 11. Intégration à WebSphere Message Broker . . . . . 187**

Planification de l'intégration à WebSphere Message Broker . . . . .	188
Versions de WebSphere Message Broker prises en charge par WebSphere Partner Gateway . . . . .	188
Transports de message pris en charge par WebSphere Message Broker . . . . .	188
Prise en charge de l'intégration à WebSphere Message Broker . . . . .	189
Configuration de WebSphere Partner Gateway pour WebSphere Message Broker . . . . .	189
Prise en charge des documents sortants . . . . .	189
Prise en charge des documents entrants . . . . .	191
Configuration de WebSphere Message Broker . . . . .	192

Création du flux de messages . . . . .	193
Déploiement du projet . . . . .	193
Utilisation du protocole de transport HTTP avec WebSphere Message Broker . . . . .	193
Composants requis pour les documents envoyés via HTTP . . . . .	193
Création du flux de messages pour le transport HTTP. . . . .	195
Envoi de documents SOAP. . . . .	197
Utilisation du protocole de transport JMS avec WebSphere Message Broker. . . . .	198
Composants requis pour les documents envoyés via JMS. . . . .	198
Configuration de l'environnement requis pour le transport JMS . . . . .	202
Création du flux de messages pour le transport JMS . . . . .	203

## **Chapitre 12. Intégration à WebSphere Data Interchange . . . . . 205**

A qui s'adresse ce chapitre ? . . . . .	205
Ressources pouvant être utilisées dans ce chapitre . . . . .	205
Introduction. . . . .	206
Envoi de documents à WebSphere Data Interchange . . . . .	206
Réception de documents de WebSphere Data Interchange . . . . .	207
Exemple de scénario utilisé dans ce chapitre . . . . .	208
Planification de l'intégration à WebSphere Data Interchange. . . . .	209
Versions de WebSphere Data InterChange prises en charge par WebSphere Partner Gateway . . . . .	209
Configuration de votre environnement pour l'échange de messages . . . . .	209
Configuration de l'intercommunication WebSphere MQ . . . . .	209
Configuration de WebSphere Data Interchange . . . . .	211
Configuration de l'environnement JMS . . . . .	216
Configuration de WebSphere Partner Gateway Enterprise Edition . . . . .	217
Configuration de WebSphere Partner Gateway - Express . . . . .	224
Configuration de mon profil . . . . .	225
Création d'un partenaire pour Partenaire numéro un . . . . .	225
Configuration du partenaire correspondant à Partenaire numéro un . . . . .	225
Récapitulatif. . . . .	227

---

## **Partie 5. Intégration à d'autres produits . . . . . 229**

### **Chapitre 13. Intégration à WebSphere Transformation Extender. . . . . 231**

Introduction. . . . .	231
Qu'est-ce que WebSphere Transformation Extender ? . . . . .	231
Comment WebSphere Transformation Extender s'interface-t-il avec WebSphere Partner Gateway ? . . . . .	232
Planification de l'intégration à WebSphere Transformation Extender . . . . .	232
Intégration via WebSphere Message Broker . . . . .	232
Intégration à l'aide d'exits utilisateur WebSphere Partner Gateway . . . . .	233
Intégration de WebSphere Message Broker à WebSphere Transformation Extender . . . . .	234
Intégration de l'exit utilisateur de WebSphere Transformation Extender . . . . .	234
Création de l'exit utilisateur de WebSphere Partner Gateway pour WebSphere Transformation Extender . . . . .	235
Configuration de WebSphere Partner Gateway pour l'exit utilisateur . . . . .	235
Configuration de la phase d'exécution de WebSphere Partner Gateway pour l'exit utilisateur. . . . .	236
Exemple d'exit utilisateur fourni pour WebSphere Partner Gateway . . . . .	236
WebSphere Transformation Extender - configuration du serveur RMI. . . . .	238

### **Chapitre 14. Intégration à WebSphere Transformation Extender Trading Manager . . . 241**

Introduction. . . . .	241
Qu'est-ce que WebSphere Transformation Extender Trading Manager ? . . . . .	241
Comment WebSphere Transformation Extender Trading Manager s'interface-t-il avec WebSphere Partner Gateway ? . . . . .	241
Documentation applicable . . . . .	241
Planification de l'intégration à WebSphere Transformation Extender Trading Manager . . . . .	242
Configuration commune entre WebSphere Partner Gateway et WebSphere Transformation Extender Trading Manager . . . . .	242
Configuration de WebSphere Partner Gateway . . . . .	242
Problèmes liés au partenaire interne. . . . .	243
Problèmes liés au partenaire externe. . . . .	243

Problèmes liés à la définition du document . . . . .	244
Problèmes liés aux interactions . . . . .	244
Problèmes liés à la connexion . . . . .	244
Problèmes liés au récepteur . . . . .	244
Configuration de WebSphere Transformation Extender Trading Manager . . . . .	244

---

**Partie 6. Autres informations spécifiques aux protocoles de gestion. . . . . 247**

**Chapitre 15. Routage de documents EDI. . . . . 249**

Présentation du routage EDI . . . . .	249
---------------------------------------	-----

**Chapitre 16. Considérations spécifiques à l’empaquetage AS. . . . . 251**

Routage des documents entrants . . . . .	251
Routage des documents sortants . . . . .	251
Configuration des deux ID dans le profil de partenaire . . . . .	251
Traitement des documents AS3 MDN . . . . .	252
Autres références AS . . . . .	252

**Chapitre 17. Considération spécifiques à l’empaquetage RosettaNet . . . . . 253**

Autres références RosettaNet . . . . .	253
--	-----

**Chapitre 18. Considérations spécifiques à l’empaquetage ebMS . . . . . 255**

Autres références ebMS . . . . .	255
----------------------------------	-----

**Remarques . . . . . 257**

Informations relatives aux interfaces de programmation . . . . .	259
Marques . . . . .	260

**Index . . . . . 261**



---

## A propos de ce guide

Ce guide décrit l'interface de Backend Integration, qui correspond au mécanisme utilisé par les systèmes dorsaux et IBM WebSphere Partner Gateway pour communiquer. Il décrit ensuite comment intégrer WebSphere Process Server, WebSphere InterChange Server, WebSphere Message Broker et WebSphere Data Interchange à WebSphere Partner Gateway à l'aide de cette interface.

Les informations de ce guide concernent uniquement les éditions Enterprise et Advanced de WebSphere Partner Gateway.

---

## Public concerné

Ce document est destiné à la personne en charge de l'intégration de WebSphere Partner Gateway aux systèmes dorsaux.

---

## Conventions typographiques

Ce document utilise les conventions typographiques suivantes.

Tableau 1. Convention typographiques

Convention	Description
Police monospace	Le texte dans cette police indique qu'il s'agit de texte que vous tapez, de valeurs pour des arguments ou des options de commande, d'exemples et d'exemples de code ou d'informations que le système imprime à l'écran (texte de message ou invite).
<b>gras</b>	Le texte en gras correspond aux commandes de l'interface graphique (par exemple, les noms des boutons en ligne, les noms ou les options de menu) et aux en-têtes de colonne dans des tables et du texte.
<i>italique</i>	Le texte en italique permet la mise en évidence de texte telle que des titres de manuels, des nouveaux termes ou des termes définis dans le texte, des noms de variables ou des lettres de l'alphabet utilisées comme lettres.
<i>Police monospace en italique</i>	Le texte figurant dans cette police signale les noms de variable dans le texte de police monospace.
<i>ProductDir</i>	<i>rép_produit</i> représente le répertoire dans lequel le produit est installé. Tous les noms de chemins du produit IBM WebSphere Partner Gateway se rapportent au répertoire dans lequel IBM WebSphere Partner Gateway est installé sur le système.
<i>%texte%</i> et <i>\$texte</i>	Le texte placé entre des signes de pourcentage (%) indique la valeur de la variable système ou utilisateur <i>texte</i> de Windows. La notation équivalente dans un environnement UNIX est <i>\$texte</i> , ce qui indique la valeur de la variable d'environnement <i>texte</i> d'UNIX.
Texte en couleur souligné	Le texte en couleur souligné indique une référence croisée. Cliquez sur le texte pour atteindre l'objet de référence.

Tableau 1. Convention typographiques (suite)

Convention	Description
<i>Texte encadré en bleu</i>	(Dans les fichiers PDF uniquement) Un texte encadré de bleu indique une référence croisée. Cliquez sur le texte avec contour pour atteindre l'objet de référence. Cette convention est l'équivalent pour les fichiers PDF de la convention "texte en couleur souligné" mentionnée dans ce tableau.
" " (guillemets)	(Dans les fichiers PDF uniquement) Les guillemets sont placés de part et d'autre des références croisées à d'autres sections du document.
{ }	Dans une ligne de syntaxe, des accolades entourent un jeu d'options parmi lesquelles une seule doit être sélectionnée.
[ ]	Dans une ligne de syntaxe, les crochets entourent des paramètres facultatifs.
< >	Des crochets en chevron entourent les éléments variables d'un nom pour les distinguer les uns des autres. Par exemple, <nom_serveur><nom_connecteur>tmp.log.
/, \	Des barres obliques inversées (\) servent de séparateurs dans les chemins d'accès aux répertoires dans les installations Windows. Pour les installations UNIX, remplacez les barres obliques inverses par des barres obliques standard (/).

## Documents associés

L'ensemble de la documentation disponible pour ce produit comprend des informations exhaustives sur l'installation, la configuration, l'administration et l'utilisation de WebSphere Partner Gateway Enterprise Edition et WebSphere Partner Gateway Advanced Edition.

Vous pouvez télécharger cette documentation ou la consulter en ligne sur le site suivant : <http://www.ibm.com/software/integration/wspartnergateway/library/>

**Remarque :** Des informations importantes relatives à ce produit peuvent être disponibles dans les notes techniques et les lettres d'informations du service d'assistance qui sont émises après la publication de ce document. Ces informations sont disponibles sur le site Web suivant du service de support de WebSphere Business Integration, <http://www.ibm.com/software/integration/wspartnergateway/support>. Sélectionnez le domaine de composant qui vous intéresse et parcourez la section Technotes and Flashes.

---

## Nouveautés de cette version

Cette section décrit les nouvelles fonctions d'IBM WebSphere Partner Gateway.

### Nouveautés de l'édition 6.1.1

WebSphere Partner Gateway 6.1.1 prend en charge les nouvelles fonctions suivantes :

- Dans les versions précédentes, la prise en charge de l'authentification standard était uniquement disponible pour les messages de services Web. Cette fonction a été étendue à tous les protocoles. Il est conseillé d'utiliser une connexion sécurisée HTTP, soit HTTPS et non HTTP.
- Prise en charge de la compression et de la décompression de données fournies pour les messages RNIF, à l'exception de la signature et du chiffrement.
- Prise en charge de la validation des éléments SOAP Body et SOAP Envelope. Par ailleurs, il est possible de désenvelopper un élément SOAP Envelope.
- Le délai d'attente maximum synchrone et les connexions maximales synchrones peuvent être contrôlés localement pour chaque récepteur HTTP.
- Le serveur FTP est intégré à WebSphere Partner Gateway pour prendre en charge le protocole AS3, le destinataire de script FTP, le récepteur de script FTP, et le récepteur et le destinataire FTP / FTPS.
- Le document d'erreur peut être envoyé au partenaire émetteur, au partenaire destinataire, ou à ces deux partenaires. Le flux de documents d'erreur peut être configuré dans la console WebSphere Partner Gateway et peut être envoyé au format WebSphere Partner Gateway ou au format Services Web.
- Les performances du programme d'archivage ont été améliorées.
- Prise en charge de plusieurs partenaires internes.
- Possibilité de renvoyer simultanément plusieurs documents entrants et sortants.
- Prise en charge du mode FIPS. Possibilité de configurer le produit pour l'exécuter en mode FIPS ou en mode par défaut.
- Fonctionnalités Supprimer et Emplacement d'utilisation fournies pour la destination, les mappes de validation, les définitions de document, les interactions et les utilisateurs.
- Prise en charge de la compression de fichiers volumineux pour les documents AS2 et AS3.
- Prise en charge du chiffrement et de la signature.
- Les dépendances du type de configuration pour la migration comprennent également les codes événement et les notifications d'alertes. Par ailleurs, la fonction de migration de partenaire a été améliorée pour la prise en charge des définitions d'importation/exportation des événements pouvant faire l'objet d'une alerte.
- Prise en charge du téléchargement de plusieurs certificats. Nouvel assistant inclus dans la console pour télécharger et configurer les certificats.
- Le produit prend désormais en charge AIX 6.1, RHEL 5 (32 et 64 bits), SLES 10 (64 bits) et Windows Server 2003 64 bits.

### Nouveautés de la version 6.1

WebSphere Partner Gateway V6.1 prend en charge les nouvelles fonctions suivantes :

- Nouveaux protocoles métier : AS3, SOAP avec pièces jointes, CIDX et prise en charge de ebMS (eXML Message Service) 2.0

- Prise en charge améliorée des documents XML personnalisés, comprenant une meilleure organisation, la prise en charge de l'expression XPath complète, des zones de recherche, des attributs définis par l'utilisateur et une prise en charge synchrone
- Prise en charge d'IPv6 et fonction améliorées de création de scripts FTP pour la prise en charge d'AS3
- Réorganisation des attributs de définition de document
- Nouveaux attributs de définition de document à utiliser avec les exits utilisateur
- Irréfutabilité configurable par type de document et par niveau de partenaire commercial
- L'afficheur de documents dispose de nouvelles zones de recherche définies par l'utilisateur.
- Prise en charge améliorée de l'afficheur AS avec états de retour MDN
- Assistant de configuration d'EDI et Assistant d'importation d'EIF (livrés auparavant dans le pack support GA02)
- Nouveau mode de notification d'alerte permettant d'envoyer des notifications à toutes les parties concernées (partenaires source et cible ou tous les contacts inscrits), ce qui réduit les opérations de configuration
- Autorisations d'accès à Gateway et à la fonction de renvoi désormais disponibles pour les utilisateurs autres que l'administrateur du concentrateur
- Nouveau groupe d'utilisateurs permettant à plusieurs utilisateurs d'assumer la fonction d'administrateur du concentrateur
- Prise en charge LDAP pour l'authentification d'ouverture de session
- Utilisation des fonctions de consignation et de traçage WebSphere Application Server pour les composants WebSphere Partner Gateway
- Les données de configuration du fichier de propriétés sont désormais stockées et gérées de manière centralisée par la console WebSphere Partner Gateway
- WebSphere MQ n'est plus obligatoire ; WebSphere Platform Messaging est utilisé pour les communications internes
- Archivage sélectif en fonction du type de document et/ou de partenaire
- Migration de la configuration de WebSphere Partner Gateway en exportant et en important des définitions depuis une instance de WebSphere Partner Gateway vers une autre instance
- Option d'installation simplifiée sur une seule machine (mode simple)
- WebSphere Application Server Network Deployment est désormais utilisé pour les déploiements sur plusieurs machines, ce qui permet la mise en cluster et une gestion centrale de l'infrastructure
- Prise en charge de l'utilisation de WebSphere Process Server Version 6.1 en tant que système d'intégration dorsale

**Remarques :**

1. L'API d'administration XML est dépréciée dans la version 6.1.
2. WebSphere Partner Gateway Version 6.1 ne prend pas en charge l'algorithme RC5.

---

## **Partie 1. Présentation de l'intégration dorsale**

Les chapitres suivants présentent l'intégration de WebSphere Partner Gateway à un système dorsal. Vous y trouverez également des instructions pour la planification de l'intégration au système dorsal.



---

## Chapitre 1. Présentation générale

Ce chapitre fournit les informations générales suivantes sur l'intégration un système dorsal :

- «Présentation du traitement des documents»
- «Présentation de l'intégration dorsale», à la page 7

---

### Présentation du traitement des documents

Avec WebSphere Partner Gateway, vous échangez des documents de gestion avec vos partenaires externes. L'objectif de cet échange de documents est de communiquer des informations, ce qui implique généralement le traitement de données et le renvoi d'un résultat. Lorsque vous recevez des données d'un partenaire externe, leur traitement intervient généralement dans le système dorsal de votre entreprise. Dans cette communauté de concentrateur, WebSphere Partner Gateway est le point autour duquel gravitent les messages entrants et sortants de l'entreprise.

L'accès à l'entreprise s'effectue au travers du système dorsal auquel WebSphere Partner Gateway se connecte.

La figure 1 montre la façon dont les documents circulent via WebSphere Partner Gateway Enterprise et Advanced Editions. Un partenaire envoie un document à WebSphere Partner Gateway (le concentrateur). WebSphere Partner Gateway reçoit le document et exécute toutes les actions qui ont été prédéfinies (telles que la validation ou la transformation du document). WebSphere Partner Gateway envoie alors le document à une application dorsale dans laquelle le document est traité.

**Remarque :** Comme l'indique l'illustration, la communication circule dans la direction opposée également. L'application dorsale peut générer un document et l'envoyer au concentrateur qui le traite et le transfère au partenaire.

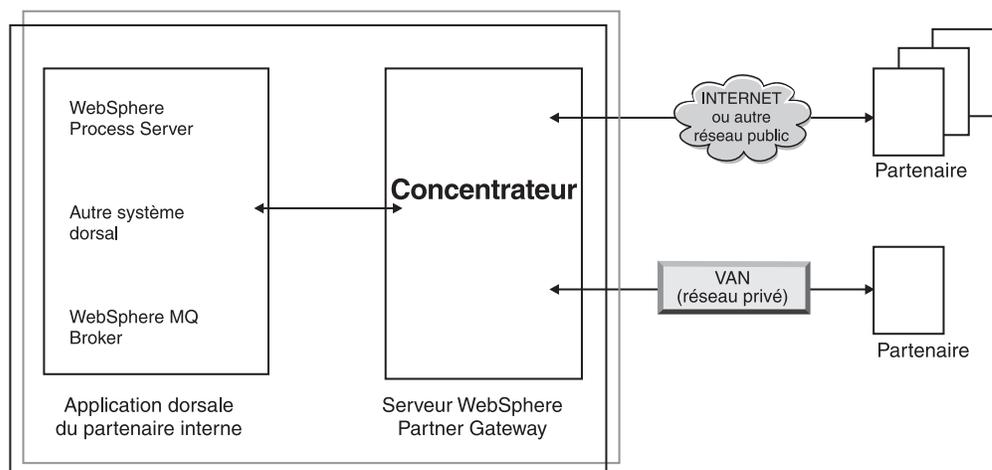


Figure 1. Flux de document complet

Ce guide porte sur l'intégration entre le concentrateur et l'application dorsale.

**Remarque :** Les informations de ce document concernent uniquement WebSphere Partner Gateway Enterprise et Advanced Editions. WebSphere Partner Gateway - Express, un outil de connectivité business-to-business souple et simple à utiliser diffère de WebSphere Partner Gateway Enterprise et Advanced Editions. Il fournit une solution d'intégration de communauté (alors que WebSphere Partner Gateway Enterprise et Advanced Editions constitue une solution de concentrateur de passerelle pour un partenaire interne). Pour plus d'informations sur WebSphere Partner Gateway - Express, voir son *Guide d'utilisateur WebSphere Partner Gateway*.

## Rôles dans la communauté du concentrateur

WebSphere Partner Gateway Enterprise et Advanced Editions comptent trois types de partenaires : l'administrateur du concentrateur, le partenaire interne et les partenaires externes. Un administrateur du concentrateur est créé automatiquement à l'installation de WebSphere Partner Gateway. L'administrateur du concentrateur est responsable de la configuration du concentrateur et de la création des partenaires qui interagiront avec ce concentrateur.

Le partenaire interne, qui est généralement le propriétaire du concentrateur, est considéré en fait comme l'un des partenaires au sein du concentrateur. L'administrateur du concentrateur crée un profil pour le partenaire interne, fournissant les informations dont il a besoin pour envoyer et recevoir des documents des partenaires. Lorsque le concentrateur envoie des documents au système dorsal, il utilise les informations (URL ou file d'attente JMS, par exemple) configurées pour le partenaire interne. L'administrateur du concentrateur crée également des profils pour les partenaires, qui peuvent être nombreux.

## Processus de configuration du concentrateur WebSphere Partner Gateway

L'administrateur du concentrateur est l'utilisateur en charge de l'administration du concentrateur. Il configure le concentrateur pour envoyer et recevoir des documents de gestion du partenaire interne et de ses partenaires externes. Pour recevoir des documents de gestion du partenaire interne, l'administrateur du concentrateur crée les récepteurs pour les modes de transport que le partenaire interne va utiliser pour envoyer les documents. Par exemple, si le partenaire interne utilise les protocoles de transport Répertoire de fichiers et JMS, l'administrateur du concentrateur configure un récepteur en mode de transport Répertoire de fichiers et un récepteur en mode de transport JMS pour le partenaire interne. De la même manière, si des partenaires externes utilisent les protocoles de transport HTTP et FTP, l'administrateur du concentrateur configure un récepteur HTTP et un récepteur FTP pour eux.

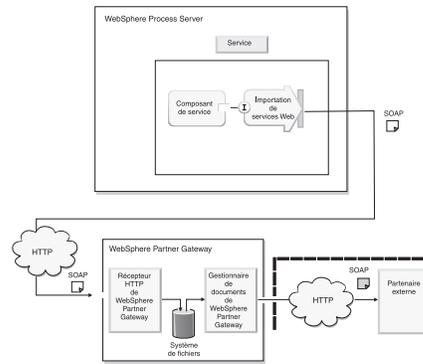


Figure 2. Récepteurs du partenaire interne et des partenaires externes

Les destinations sont créées pour le partenaire interne et les partenaires externes pour chacun des modes de transport qu'ils utiliseront pour recevoir des documents de gestion envoyés par le concentrateur.

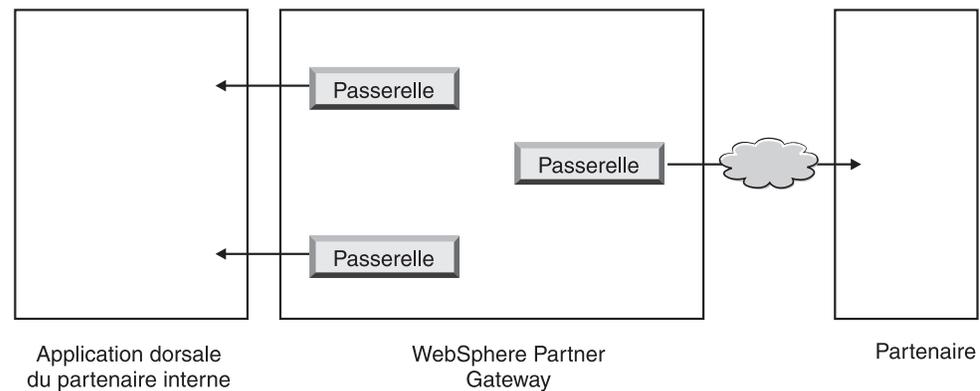


Figure 3. Destinations pour le partenaire interne et les partenaires externes

Dans le cadre de la configuration du concentrateur, l'administrateur du concentrateur établit des définitions de flux de documents qui définissent, comme il se doit, les caractéristiques d'un flux de document, telles que :

- L'empaquetage, qui donne les informations pour le routage du document
- Le protocole, qui est le protocole de gestion suivi par le document ;
- Le flux de document, qui représente le document lui-même.

Lorsque WebSphere Partner Gateway est installé, un ensemble de définitions de flux de document est disponible à l'utilisation. Vous pouvez également contribuer aux définitions de flux de document en créant vos propres définitions ou en téléchargeant des définitions vers le système. Par exemple, des définitions de flux de document pour divers PIP RosettaNet sont incluses sous forme de fichiers ZIP sur le support d'installation. Vous pouvez télécharger ces fichiers vers le système pour les rendre disponibles. Si vous échangez des fichiers EDI, vous pouvez importer des définitions de flux de document et des mappes associées à partir du client Data Interchange Services.

Prenons l'exemple suivant : un partenaire externe envoie un message RNIF 2.0 contenant un document de commande d'achat RosettaNet PIP 3A4 au récepteur HTTP de WebSphere Partner Gateway. Le message est libellé à l'attention du partenaire interne. Le partenaire interne est doté d'un système dorsal qui traite les commandes d'achat et s'attend à recevoir un bon de commande, constitué en substance des données utiles du message RNIF envoyées par le partenaire externe. Avant la configuration des connexions du partenaire externe dans WebSphere Partner Gateway, il est convenu que :

- Le partenaire externe va envoyer un message RNIF contenant le bon de commande RosettaNet PIP 3A4 via HTTP.
- WebSphere Partner Gateway va extraire les données utiles de gestion ou le contenu du service RosettaNet du message entrant.
- Le document sera routé vers le système dorsal via JMS. L'empaquetage Backend Integration sera utilisé.
- L'application dorsale traitera alors le document reçu.

Lorsque l'empaquetage Backend Integration est utilisé, les en-têtes de niveau transport définis par WebSphere Partner Gateway sont ajoutés au document pour transmettre les informations utiles à l'échange de documents.

Dans l'exemple précédent, l'administrateur du concentrateur téléchargeait vers le système le module PIP approprié, ce qui configurait les définitions de flux de document suivantes pour l'échange de document RosettaNet PIP 3A4 :

- un flux qui comprend l'empaquetage RNIF, le protocole RosettaNet et le processus PIP 3A4 ;
- un flux qui comprend l'empaquetage Backend Integration, le protocole RNSC et le processus PIP 3A4.

Lorsque l'administrateur du concentrateur a établi les définitions de flux de document, il crée les interactions pour les définitions de flux de document. Par exemple, l'administrateur du concentrateur peut indiquer que la définition du flux de document RNIF/RosettaNet/3A4 peut parvenir au concentrateur depuis une source.

L'administrateur du concentrateur (ou les partenaires externes) sélectionne les fonctions business-to-business appropriées pour l'échange de documents. Dans cet exemple, les fonctions business-to-business ci-après sont activées pour le partenaire interne :

- Empaquetage : Backend Integration
- Protocole : RNSC
- Flux de document : 3A4

Les fonctions business-to-business ci-après sont activées pour le partenaire externe :

- Empaquetage : RNIF
- Protocole : RosettaNet
- Flux de document : 3A4

L'administrateur du concentrateur crée alors les connexions entre partenaires externes.

Dans l'illustration suivante, l'administrateur du concentrateur a créé des profils pour le partenaire interne et les partenaires externes, des récepteurs pour recevoir les documents et des destinations pour les envoyer, les définitions de flux de document répertoriées ci-dessus, il a défini les fonctions business-to-business du partenaire externe et du partenaire interne et a créé une connexion entre les deux.

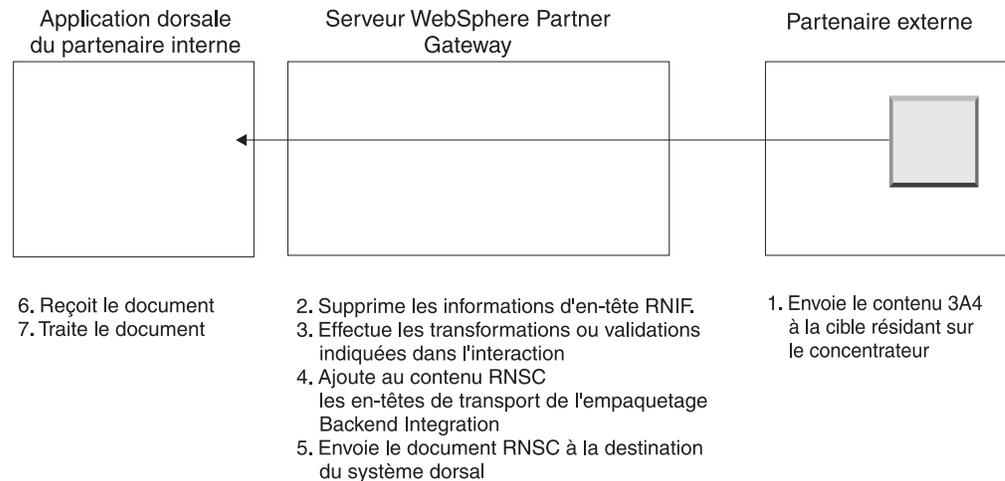


Figure 4. Comment circule un document jusqu'au système dorsal

Pour obtenir des informations sur la configuration du concentrateur, voir le *guide de configuration du concentrateur WebSphere Partner Gateway*.

## Présentation de l'intégration dorsale

Chaque édition de WebSphere Partner Gateway permet la connexion de systèmes dorsaux. Seuls les protocoles de transport pris en charge par ces éditions diffèrent, comme suit :

- WebSphere Partner Gateway - Express fournit une intégration basée sur les fichiers.
- Les WebSphere Partner Gateway Enterprise et Advanced Editions proposent une intégration basée sur des fichiers. Elles permettent également l'intégration de protocoles HTTP, HTTPS, et JMS.

Les documents échangés entre le partenaire externe et WebSphere Partner Gateway peuvent se présenter sous différents formats. Les documents peuvent être au format SOAP, cXML, XML, EDI, ROD (record-oriented data)/fichier à plat, des formats binaires ou un format personnalisé convenu avec les partenaires externes. Dans le *Guide d'administration WebSphere Partner Gateway*, vous trouverez une liste complète des différents types de documents pris en charge ainsi que les divers protocoles de transport (ex: HTTP) permettant l'envoi de documents.

Une liste de documents échangeables entre WebSphere Partner Gateway et le système dorsal du partenaire interne associés aux types de protocoles de transport nécessaires à l'envoi de ces documents vous est proposée dans le tableau 16, à la page 38, le tableau 17, à la page 38 et le tableau 18, à la page 39.

La figure 5 explique comment WebSphere Partner Gateway utilise l'interface d'intégration dorsale pour communiquer avec l'application dorsale au niveau du partenaire interne. Notez que les flèches sont bidirectionnelles, par conséquent le document peut également être émis par le système dorsal du partenaire interne.

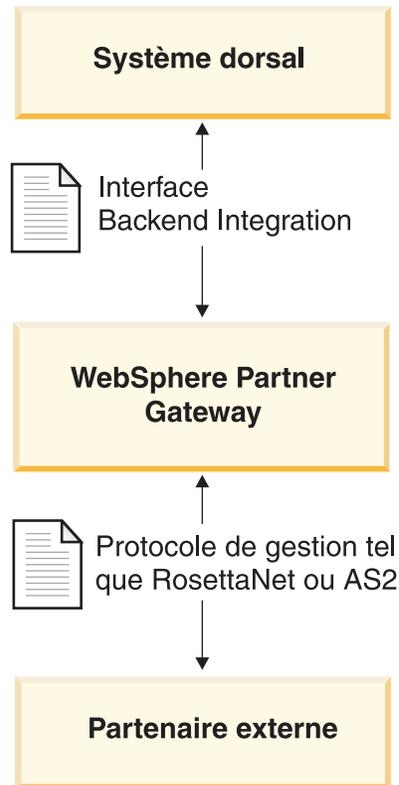


Figure 5. Rôle du protocole de gestion et de l'emballage dans le flux de document

s

---

## Chapitre 2. Planification de l'intégration dorsale

Le présent chapitre définit les méthodes de planification nécessaires à l'intégration du logiciel WebSphere Partner Gateway sur une application dorsale. Il décrit les types de décisions que vous serez amené à prendre lors de la planification de l'intégration dorsale :

- «Quel est le protocole de gestion utilisé ?»
- «Quel emballage allez-vous utiliser ?», à la page 26
- «Quel mode de transport de messages allez-vous utiliser ?», à la page 37
- «Comment accéder à l'application dorsale?», à la page 45

Ce chapitre contient également les informations suivantes :

- «Traitement des messages», à la page 46
- «Configuration de WebSphere Partner Gateway», à la page 47

---

### Quel est le protocole de gestion utilisé ?

Le protocole de gestion de votre message détermine le format du document. Le protocole de gestion influence sur les décisions à prendre lors de la planification de l'application dorsale. Le choix du protocole de gestion détermine la méthode d'emballage à utiliser, qui, à son tour, affecte le type de protocole de transport des messages pouvant être utilisé.

Pour une description complète des protocoles de gestion, reportez-vous au *Guide de configuration du concentrateur WebSphere Partner Gateway*. Cette section fournit des informations nécessaires à l'intégration spécifiques aux protocoles de gestion suivants:

- «Services Web (SOAP)»
- «cXML», à la page 10
- «EDI», à la page 10

**Remarque :** La section relative à EDI décrit également comment sont traités les documents XML et ROD (Record-Oriented-Data)/fichiers à plat.

- «RosettaNet», à la page 16

### Services Web (SOAP)

WebSphere Partner Gateway peut mettre les services Web suivants à la disposition des membres de la communauté du concentrateur :

- Les services Web proposés par le partenaire interne sont mis à la disposition des partenaires externes.

Vous procurerez à vos partenaires externes le WSDL public généré par WebSphere Partner Gateway. Il est important de noter que l'URL utilisée par le partenaire externe pour appeler le service Web correspond à l'URL publique du service Web spécifié au chargement du service Web. WebSphere Partner Gateway agit en tant que proxy. Il reçoit un message SOAP en provenance du partenaire et détermine quel est le service Web privé correspondant. Il appelle ensuite le service Web privé (fourni par le partenaire interne) au moyen du même message SOAP. La réponse renvoyée par le partenaire interne est ensuite transmise au partenaire externe.

- Les services Web fournis par les partenaires externes peuvent être mis à la disposition du partenaire interne.

Il est important de noter que la même interface de service Web peut être fournie par plusieurs partenaires. WebSphere Partner Gateway met le service Web à la disposition du partenaire interne au niveau de l'URL du service Web spécifié dans la console lors du chargement du service Web. De plus, le partenaire interne devra fournir le paramètre d'URL pour identifier la valeur "Vers le partenaire". Pour plus d'informations, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*. WebSphere Partner Gateway agit en tant que proxy. Il reçoit un message SOAP en provenance du partenaire interne et détermine quel est le service Web privé correspondant, ainsi que la valeur de "Vers le partenaire". Il appelle ensuite le service Web fourni par le partenaire au moyen du même message SOAP. Le message en réponse renvoyé par le partenaire est ensuite transmise au partenaire interne.

Pour plus d'informations, notamment sur la configuration des définitions de flux de document pour les services Web, reportez-vous au *Guide de configuration du concentrateur WebSphere Partner Gateway*.

## cXML

Vous pouvez envoyer ou recevoir des documents cXML en provenance ou à destination de vos partenaires externes. Lorsque WebSphere Partner Gateway reçoit un document cXML provenant d'un partenaire externe, il le valide et le convertit (le cas échéant) avant de l'envoyer au système dorsal chez le partenaire interne. Notez que la conversion ne doit pas être utilisée pour les messages cXML synchrones. Dans un échange synchrone, le système dorsal génère une réponse, que WebSphere Partner Gateway renvoie au partenaire externe (si c'est pertinent pour le message).

Un système dorsal chez le partenaire interne qui doit envoyer un document cXML peut procéder de deux manières :

- générer et envoyer un document cXML transmis par WebSphere Partner Gateway au partenaire externe ;
- générer et envoyer un document XML converti par WebSphere Partner Gateway au format cXML avant de l'envoyer au partenaire externe.

**Remarque :** Si la conversion du document XML est utilisée pour les transactions de demande/réponse synchrones avec le partenaire externe, la réponse est renvoyée en mode asynchrone au système dorsal.

Pour plus d'informations, notamment sur la configuration des définitions de flux de document pour les services Web, reportez-vous au *Guide de configuration du concentrateur WebSphere Partner Gateway*.

## EDI

WebSphere Partner Gateway accepte les documents EDI émanant des partenaires des réseaux à valeur ajoutée comme d'Internet. Les documents EDI envoyés à un réseau à valeur ajoutée ou reçus par celui-ci utilisent le transport de script FTP. Ce mode de transport peut également être utilisé pour envoyer des documents à Internet ou en recevoir de ce dernier. Pour plus d'informations sur le transfert de scripts FTP, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

Un document EDI entre dans le concentrateur et quitte le concentrateur dans une enveloppe EDI, appelée *Interchange*. L'échange contient des transactions ou des groupes de transactions EDI.

Dans le cas où l'échange EDI est transmis via le concentrateur (sans être désenveloppé), vous devez créer une connexion entre celui-ci et le partenaire interne.

Cependant, dans le cas où l'échange EDI sera désenveloppé, le processus de création d'interactions et de connexions est différent de celui d'autres protocoles de gestion. L'Interchange doit être désenveloppé et les transactions individuelles doivent être traitées. Les transactions sont généralement converties dans un autre format, selon une mappe de transformation importée à partir du client Data Interchange Services. Si les transactions EDI sont converties en documents XML ou ROD (données orientées enregistrement)/fichiers à plat, ces documents sont envoyés au partenaire interne ou au partenaire externe. Si les transactions sont converties dans d'autres formats EDI, les transactions sont enveloppées avant d'être envoyées au partenaire interne ou au partenaire externe.

### Application dorsale vers les flux partenaires

Une application dorsale peut envoyer les types de document suivants :

- un seul Interchange EDI contenant une ou plusieurs transactions ;  
WebSphere Partner Gateway désenveloppe les transactions EDI individuelles et les convertit. Si les transactions sont converties en EDI, elles sont enveloppées puis routées vers le partenaire. L'application dorsale peut n'utiliser aucun emballage ou un emballage dorsal et envoyer l'échange sur divers protocole de transport, comme cela est défini dans tableau 17, à la page 38.

La figure 6 présente un échange X12 consistant en le désenveloppement de trois transactions. Les transactions sont transformées au format EDIFACT puis enveloppées et envoyées au partenaire.

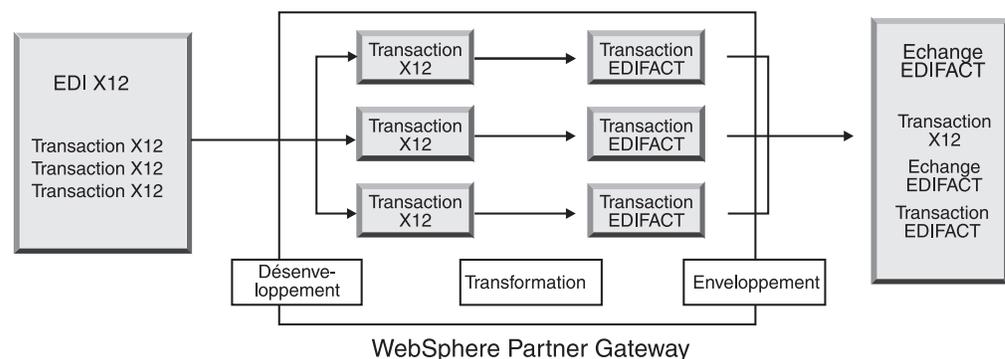


Figure 6. Echange EDI de l'application dorsale au partenaire

Chacune des transactions est associée à une mappe de transformation qui spécifie comment la transaction est convertie. La transaction peut être convertie dans une seule transaction ou, si le chaînage de mappes a été utilisé pour créer la mappe, en plusieurs transactions.

Si la transaction est convertie en document XML ou ROD/fichier à plat, elle est routée comme spécifié dans la configuration de la vue Connexion du partenaire pour cette transaction.

La figure 7 montre un échange EDI X12 désenveloppé et converti en documents XML, qui sont envoyés au partenaire.

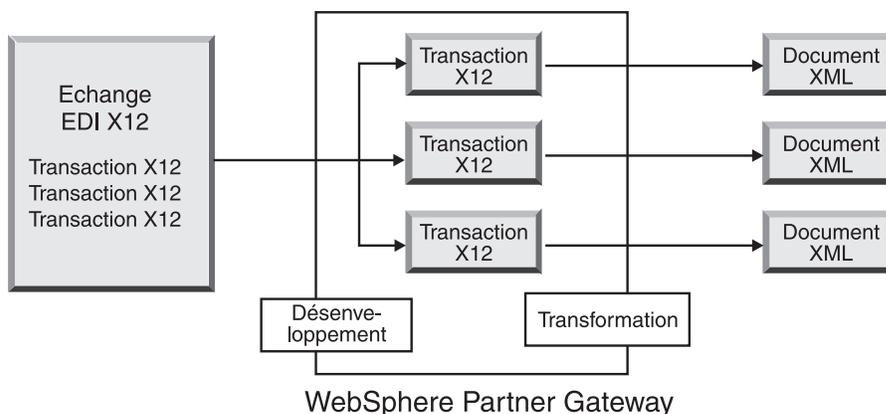


Figure 7. Echange EDI envoyé de l'application dorsale au partenaire (sous forme de documents XML)

La transaction peut être transformée en un seul document ou, si le chaînage de mappes a été utilisé pour créer la mappe, en plusieurs documents.

- Un seul document, tel que fichier XML ou ROD/à plat.

WebSphere Partner Gateway convertit le document en une transaction EDI, l'enveloppe et l'envoie au partenaire. L'application dorsale peut n'utiliser aucun empaquetage ou un empaquetage Backend Integration et peut envoyer le document sur divers protocoles de transport, comme cela est défini dans le tableau 17, à la page 38.

La figure 8, à la page 13 montre un document XML converti en transactions X12, puis enveloppé.

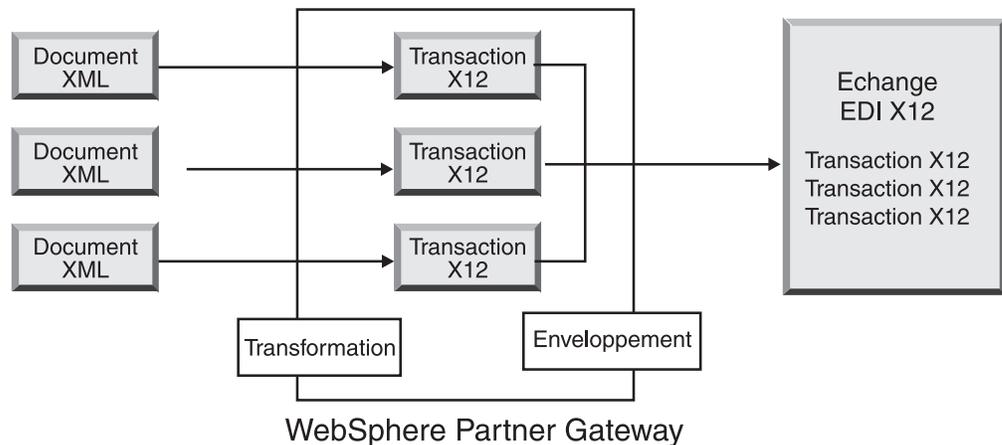


Figure 8. Documents XML envoyés de l'application dorsale au partenaire (sous forme d'échange EDI)

Un document peut être transformé en plusieurs transactions (si le chaînage de mappes a été utilisé pour créer la mappe) et les transactions peuvent être enveloppées dans différents échanges.

La figure 9 montre un document XML converti en trois transactions X12. Deux d'entre elles sont enveloppées ensemble. Une des deux est placée dans une enveloppe distincte.

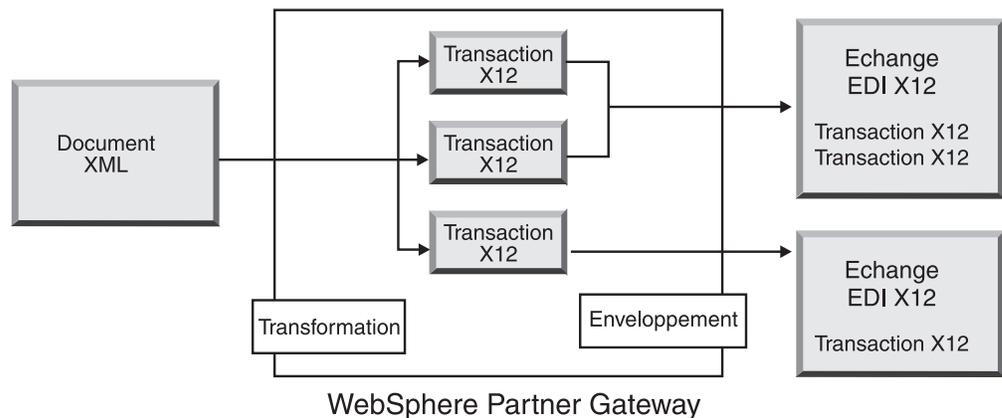


Figure 9. Document XML envoyé de l'application dorsale au partenaire (sous forme d'échange EDI)

Si le document est converti en document XML ou un autre fichier ROD/fichier à plat, elle est routée comme spécifié dans la configuration affichés dans la vue Connexion du partenaire pour ce document.

- Un seul fichier contenant plusieurs documents XML ou ROD/à plat ;  
WebSphere Partner Gateway divise les documents et les convertit. Si les documents sont convertis en transactions EDI, WebSphere Partner Gateway les enveloppe et envoie l'enveloppe au partenaire. Si des ID de lot ont été affectés aux documents XML ou ROD/à plat, WebSphere Partner Gateway tente d'envoyer les transactions EDI dans une enveloppe (sous forme de lot). L'application dorsale peut n'utiliser aucun empaquetage ou un empaquetage Backend Integration et peut envoyer le document sur divers protocoles de transport, comme cela est défini dans le tableau 17, à la page 38.

La figure 10 montre un ensemble de documents XML en cours de séparation. Des documents XML distincts sont donc obtenus. Les documents XML sont transformés en transactions X12 qui sont ensuite enveloppées.

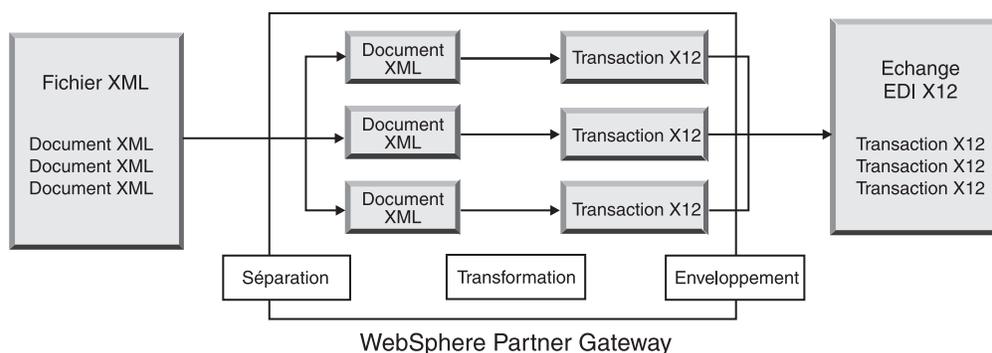


Figure 10. Plusieurs documents XML envoyés depuis une application dorsale, scindés, puis envoyés au partenaire (sous forme d'échange EDI)

La figure 10 montre comment les documents sont scindés et les transactions converties enveloppées. Pour pouvoir scinder les documents, vous devez configurer un gestionnaire pour scinder les documents (dans ce cas, XML Splitter Handler) pour le récepteur utilisé pour l'envoi de documents. Pour que ce scénario ait lieu, l'option BCG\_BATCHDOCS du gestionnaire de séparation XML doit être activé (valeur par défaut). BCG\_BATCHDOCS affecte un ID de lot aux documents XML de sorte que les transactions résultantes seront placées dans la même enveloppe. Pour plus d'informations sur le gestionnaire de séparation XML et l'attribut BCG\_BATCHDOCS, voir *le guide de configuration du concentrateur WebSphere Partner Gateway*.

Si les documents sont convertis dans d'autres documents XML ou ROD/fichiers à plat, ils sont routés comme spécifié dans la configuration de la connexion du partenaire pour ces documents.

- un seul fichier contenant plusieurs échanges EDI.

WebSphere Partner Gateway divise le fichier en échanges individuels. Il désenveloppe alors les échanges en transactions individuelles et les convertit. Si les documents sont convertis en transactions EDI, WebSphere Partner Gateway les enveloppe et envoie l'enveloppe au partenaire. L'application dorsale peut n'utiliser aucun emballage ou un emballage Backend Integration et peut envoyer le document sur divers protocoles de transport, comme cela est défini dans le tableau 17, à la page 38.

Si les documents sont convertis dans d'autres documents XML ou ROD/fichiers à plat, ils sont routés comme spécifié dans la configuration de la connexion du partenaire pour ces documents.

## Flux partenaire-application dorsale

Un partenaire peut envoyer les types de document suivants :

- un seul Interchange EDI contenant une ou plusieurs transactions ;  
WebSphere Partner Gateway désenveloppe les transactions EDI individuelles et les convertit. Si les transactions sont converties en EDI, elles sont enveloppées puis routées vers l'application dorsale. L'application dorsale peut n'utiliser aucun emballage ou un emballage Backend Integration et les transactions peuvent être envoyées via divers protocoles de transport, comme cela est défini dans le tableau 18, à la page 39.  
Si les transactions sont converties dans d'autres documents XML ou ROD/fichiers à plat, ils sont routés comme spécifié dans la configuration de la connexion du partenaire pour ces transactions.
- Un seul document, tel que fichier XML ou ROD/à plat.  
WebSphere Partner Gateway convertit le document en une transaction EDI, l'enveloppe et envoie l'enveloppe à l'application dorsale. Aucun emballage ou un emballage Backend Integration peut être utilisé.  
Si le document est converti en document XML ou un autre fichier ROD/fichier à plat, elle est routée comme spécifié dans la configuration affichés dans la vue Connexion du partenaire pour ce document.
- Un seul fichier contenant plusieurs documents XML ou ROD/à plat ;  
WebSphere Partner Gateway divise les documents et les convertit. Si les documents sont convertis en transactions EDI, WebSphere Partner Gateway les enveloppe et envoie l'enveloppe à l'application dorsale. Si des ID de lot ont été affectés aux documents XML ou ROD/à plat, WebSphere Partner Gateway tente d'envoyer les transactions EDI dans une enveloppe (sous forme de lot). Aucun emballage ou un emballage Backend Integration peut être utilisé.  
Si les documents sont convertis dans d'autres documents XML ou ROD/fichiers à plat, ils sont routés comme spécifié dans la configuration de la connexion du partenaire pour ces documents.
- un seul fichier contenant plusieurs échanges EDI.  
WebSphere Partner Gateway divise le fichier en échanges individuels. Il désenveloppe alors les échanges en transactions individuelles et les convertit. Si les documents sont convertis en transactions EDI, WebSphere Partner Gateway les enveloppe et envoie l'enveloppe à l'application dorsale. Aucun emballage ou un emballage Backend Integration peut être utilisé.  
Si les documents sont convertis dans d'autres documents XML ou ROD/fichiers à plat, ils sont routés comme spécifié dans la configuration de la connexion du partenaire pour ces documents.

## Accusés de réception fonctionnels

Un accusé de réception fonctionnel spécifie qu'un échange EDI a été reçu. Il est toujours enveloppé avant d'être envoyé.

**Remarque :** Les accusés de réception fonctionnels s'appliquent uniquement aux échanges qui sont désenveloppés ou générés par WebSphere Partner Gateway. Ils ne s'appliquent pas aux échanges transmis via WebSphere Partner Gateway.

Pour les échanges reçus par WebSphere Partner Gateway

- Si l'échange est reçu d'une application dorsale, WebSphere Partner Gateway peut envoyer des accusés de réception fonctionnels à celle-ci.
- Si l'échange est reçu d'un partenaire, WebSphere Partner Gateway peut lui envoyer en retour des accusés de réception fonctionnels.

Pour les échanges générés par WebSphere Partner Gateway :

- Si l'échange est envoyé à un partenaire, celui-ci peut envoyer en retour un accusé de réception fonctionnel à WebSphere Partner Gateway. WebSphere Partner Gateway ne l'enverra pas au système dorsal.
- Si l'échange est envoyé à l'application dorsale, celle-ci peut envoyer un accusé de réception fonctionnel à WebSphere Partner Gateway. WebSphere Partner Gateway ne l'enverra pas au partenaire.

## RosettaNet

WebSphere Partner Gateway prend en charge les envois et réceptions de documents conformes aux normes RosettaNet 1.1 et 2.0. Lorsqu'un partenaire envoie un message RosettaNet au concentrateur, Backend Integration doit être spécifié du côté cible de la connexion du partenaire. Le concentrateur convertit les données utiles du message au format RNSC et envoie le message au système dorsal. En raison de l'utilisation de l'empaquetage Backend Integration, le concentrateur ajoute des en-têtes de niveau transport au message. Ce message est envoyé via le protocole de transport HTTP ou JMS. L'en-tête de niveau transport conserve les méta-informations qui ne font pas partie du PIP et active WebSphere Partner Gateway afin qu'il route correctement le message.

De façon similaire, lorsque le système dorsal du partenaire interne envoie un message RNSC au concentrateur, l'empaquetage Backend Integration doit être spécifié du côté source de la connexion du partenaire et le système dorsal doit fournir les en-têtes de niveau transport.

Par exemple, imaginons qu'une application souhaite envoyer un message à un partenaire externe au moyen de RosettaNet via HTTP. L'application fournit le contenu du service RosettaNet et ajoute l'en-tête de niveau transport. L'en-tête identifie le partenaire externe chargé de répondre à la demande, le PIP à envoyer et sa version, ainsi que d'autres informations. Ces informations permettent alors à WebSphere Partner Gateway d'envoyer le bon PIP au partenaire externe.

Les informations relatives à l'installation de la prise en charge de RosettaNet et à la configuration des PIP se trouvent dans le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

### Notification d'événement

WebSphere Partner Gateway exécute des processus RNIF PIP avec les partenaires externes pour le compte des applications dorsales du partenaire interne. Par conséquent, WebSphere Partner Gateway fournit une *notification d'événement* afin d'informer l'application dorsale des divers aspects de l'exécution du processus RNIF PIP. La notification d'événement permet à WebSphere Partner Gateway, par exemple, de notifier l'application si WebSphere Partner Gateway n'est pas en mesure d'envoyer un PIP au partenaire. L'application peut ainsi prendre en charge cet échec.

Un message de notification d'événement est un document XML qui comporte des informations sur les événements qui sont intervenus dans WebSphere Partner Gateway ou dans une application. Ces messages ont la même structure que tout autre message entrant ou sortant de WebSphere Partner Gateway : en d'autres termes, ils comportent un en-tête de niveau de transfert et des données utiles. WebSphere Partner Gateway peut être configuré pour envoyer ou non des messages de notification d'événement car ces messages sont facultatifs.

Le tableau 2 résume les messages de notification d'événements que WebSphere Partner Gateway est susceptible d'envoyer aux systèmes dorsaux.

Tableau 2. Notification d'événement envoyée au système dorsal

Condition d'événement	Message de notification d'événement
WebSphere Partner Gateway fournit un document RosettaNet à un partenaire externe et reçoit un accusé de réception.	Code de statut d'événement 100
Erreur générale lors de l'exécution de l'action RNIF. WebSphere Partner Gateway annule un PIP en générant un message 0A1 et en le délivrant au partenaire externe.  S'il s'agit de la dernière action dans le PIP, ce code d'action entraîne l'annulation par WebSphere Partner Gateway de ce PIP en générant un message 0A1 et en le délivrant au partenaire externe. Autrement, WebSphere Partner Gateway envoie une "exception générale" au partenaire.	Code de statut d'événement 800
Erreur lors de l'exécution de l'action RNIF. Validation du contenu de service au regard du dictionnaire spécifié dans le PIP. WebSphere Partner Gateway annule un PIP en générant un message 0A1 et en le délivrant au partenaire externe.	Code de statut d'événement 801
WebSphere Partner Gateway reçoit une exception d'accusé de réception ou une exception générale d'un partenaire externe.	Code de statut d'événement 900

WebSphere Partner Gateway peut envoyer des messages 0A1 à l'application cible comme il le ferait avec n'importe quel autre PIP, s'il a été configuré pour envoyer ces messages à l'aide de la Procédure de gestion des listes d'exclusion. Voir "Gestion des listes d'exclusion" dans le *Guide de l'administrateur WebSphere Partner Gateway*.

Une application peut envoyer un message de notification d'événement à WebSphere Partner Gateway pour annuler un PIP RosettaNet.

Pour toute information sur les codes d'événement employés par WebSphere Partner Gateway, voir «Codes d'événement employés par WebSphere Partner Gateway», à la page 25.

## ebMS

WebSphere Partner Gateway prend en charge les envois et réceptions de documents conformes aux normes ebMS 2.0. Lorsqu'un partenaire envoie un message ebMS au concentrateur, l'empaquetage Backend Integration doit être spécifié du côté cible de la connexion du partenaire. Le concentrateur extrait les données utiles du message et envoie le message au système dorsal. En raison de l'utilisation de l'empaquetage Backend Integration, le concentrateur ajoute des en-têtes de niveau transport au message. Ce message est envoyé via le protocole de transport HTTP ou JMS. L'en-tête de niveau transport conserve les méta-informations qui ne font pas partie du message ebMS et permet à WebSphere Partner Gateway de router correctement le message.

De façon similaire, lorsque le système dorsal du partenaire interne envoie un message RNSC au concentrateur, l’empaquetage Backend Integration doit être spécifié du côté source de la connexion du partenaire et le système dorsal doit fournir les en-têtes de niveau transport.

Par exemple, supposons qu’une application souhaite envoyer un message à un partenaire externe au moyen d’ebMS via HTTP. L’application indique le service ebMS, l’action, les ID de partenaires (source et cible), les informations de protocole et d’empaquetage, l’ID du message et l’ID de conversation, puis ajoute les en-têtes de niveau transport. L’en-tête identifie le partenaire externe chargé de répondre à la demande. Ces informations permettent à WebSphere Partner Gateway d’élaborer le message ebMS et de mettre les données utiles issues du dorsal en données utiles ebMS, pour envoyer ensuite le message au partenaire externe.

La définition des en-têtes Backend dans tableau 3 est indispensable dans le cas de flux ebMS.

Tableau 3. En-têtes Backend indispensables pour les flux ebMS

Nom de l’en-tête	Description	Valeur d’exemple	Obligatoire
x-aux-sender-id	ID de l’expéditeur du message, tel qu’un numéro DUNS	987654321	Oui
x-aux-receiver-id	ID du récepteur du message, tel qu’un numéro DUNS	123456789	Oui
x-aux-protocol	Protocole du contenu du message. Cette valeur doit concorder avec la valeur de protocole pour le transport vers la source sur la vue Connexion du partenaire.	ebMS	Oui
x-aux-protocol-version	Version du protocole du contenu du message Cette valeur doit concorder avec la version de protocole pour le transport vers la source sur la vue Connexion du partenaire.	2.0	Oui
x-aux-process-type	Processus à réaliser ou type de message à envoyer. Cette valeur doit concorder avec la valeur de type de document pour le transport vers la source sur la vue Connexion du partenaire.	Demande de commande d’achat	Oui
x-aux-process-version	Version du processus. Cette valeur doit concorder avec la version du type de document pour le transport vers la source sur la vue Connexion du partenaire. Cette valeur est la même que le type Service dans le message ebXML.	TOUT	Oui
x-aux-activity	Activité à réaliser sur le type du message envoyé. Cette valeur doit concorder avec la valeur de l’activité du côté source sur la vue Connexion du partenaire. Dans le cas des flux ebMS, elle est identique à x-aux-process-type.	Demande de commande d’achat	Oui

Tableau 3. En-têtes Backend indispensables pour les flux ebMS (suite)

Nom de l'en-tête	Description	Valeur d'exemple	Obligatoire
x-aux-action	Action à réaliser sur le message ebMS. Sa valeur est la même que pour l'élément Action dans le message ebXML Soap. En outre, cette valeur doit concorder avec la valeur de l'action du côté source sur la vue Connexion du partenaire.	Action de commande d'achat	Oui
x-aux-system-msg-id	Identificateur global unique (GUID) du message, utilisé pour la vérification de la présence de doublons. Cet ID est utilisé par WebSphere Partner Gateway pour la vérification de la présence de doublons. Il ne sert pas au flux ebMS.	28282828282828	Oui
x-aux-production	Routage du message. Les valeurs valides sont les suivantes : Production et Test. Ces valeurs sont renseignées pour les requêtes effectuées dans les deux sens.	Production	Oui
x-aux-msg-id	Identificateur unique de message. Défini en tant que messageId dans le message ebXML Soap	123454321@abc.com	Non
contentType	Type de contenu du message. Non obligatoire si envoyez un message encapsulé en wbiPackaging à partir de backend.	text/xml	Non
x-aux-process-instance-id	ProcessId ou ConversationId du message ebXML.	329878738@abc.com	Non
x-aux-seq-number	Numéro de séquence du message, si "Sémantique de l'ordre des messages" est défini comme "Garanti" du côté cible de la définition de flux de documents. Il s'agit de la séquence du message émis lors de cette conversation.	2	Non
x-aux-request-msg-id	Cet en-tête ne doit être défini que pour le document en réponse. La valeur de cet en-tête sera l'IDMessage du document de demande associé au document en réponse courant.	123455521@abc.com	Non
x-aux-role	Il n'est pas obligatoire de renseigner cette zone si c'est WBIPackaging1.2 qui est employé, étant donné que l'utilisateur peut définir cet attribut dans le doc XML lui-même. Dans le cas contraire, la valeur de cette zone sera la valeur de l'attribut de rôle dans l'élémentManifest/Reference du message ebXML Soap.	<a href="http://reqrep.org/gci/purchaseOrder">http://reqrep.org/gci/purchaseOrder</a>	Non

Tableau 3. En-têtes Backend indispensables pour les flux ebMS (suite)

Nom de l'en-tête	Description	Valeur d'exemple	Obligatoire
x-aux-role-schema	Schéma servant à la validation du rôle. Cette validation n'est pas effectuée par WebSphere Partner Gateway. La valeur de cette zone est placée dans le message ebXML Soap en tant qu'attribut Schema.	http://reqrep.org/gci/purchaseOrder/po.xsd	Non
x-aux-role-schema-version	Version du schéma ci-dessus	2.0	Non
x-aux-description	Description des données utiles. Cette valeur est mise comme valeur de l'élément Description dans le message ebXML Soap.	Demande de commande d'achat	Non

Les informations relatives à la mise en place de la prise en charge d'ebMS et à la configuration des connexions ebMS se trouvent dans le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

### Notification d'événement

WebSphere Partner Gateway apporte *notification d'événement* comme mécanisme servant à aviser l'application dorsale des divers aspects de l'exécution du processus ebMS. Pour plus de précisions sur la notification d'événement, reportez-vous à la section EventNotification/XMLEvent.

tableau 4 récapitule les messages de notification d'événement pour ebMS envoyés par et à WebSphere Partner Gateway par et au dorsal.

Tableau 4. Notification d'événement envoyée au système dorsal

Condition d'événement	Message de notification d'événement
WebSphere Partner Gateway génère cet événement après avoir reçu l'accusé de réception du partenaire.	Code de statut d'événement 100
WebSphere Partner Gateway reçoit une exception d'accusé de réception ou une exception générale d'un partenaire externe.	Code de statut d'événement 900
WebSphere Partner Gateway génère cet événement lorsque toutes les relances ont été faites ou que la durée de vie du message est dépassée et qu'aucun accusé de réception n'a été reçu pour le message.	Code de statut d'événement 902
WebSphere Partner Gateway génère cet événement en cas d'échec de distribution du message. C'est à dire, lorsque WebSphere Partner Gateway n'est pas parvenu à envoyer le message à un partenaire car la destination de ce partenaire est hors service et que le nombre de relances au niveau transport et destination a été dépassé.	Code de statut d'événement 903

Pour toute information sur les codes d'événement employés par WebSphere Partner Gateway, voir «Codes d'événement employés par WebSphere Partner Gateway», à la page 25.

## XMLEvent

WebSphere Partner Gateway offre une manière de notifier le dorsal du statut d'un message. Par exemple, si WebSphere Partner Gateway envoie un message à un partenaire et que celui-ci lui envoie en retour un accusé de réception, WebSphere Partner Gateway générera un événement XML contenant un code d'événement 100 (la liste des codes d'événement figure dans «Codes d'événement employés par WebSphere Partner Gateway», à la page 25) et l'enverra au dorsal. La structure des événements XML (XML (Event)) est décrite à la «Structure des messages d'événement», à la page 22. Les XMLEvents ne sont employés qu'entre WebSphere Partner Gateway et une application dorsale. Il ne sont jamais envoyés aux partenaires. Dans WebSphere Partner Gateway 6.1, les XMLEvents ne sont générés que pour les flux RosettaNet et ebMS.

Vous pouvez activer ou désactiver un XMLEvent créé entre WebSphere Partner Gateway et une application dorsale. Voir «activation ou désactivation des XMLEvents» pour plus d'informations.

### activation ou désactivation des XMLEvents

Pour activer un XMLEvent, procédez de la manière suivante :

1. Activez les fonctions business-tu-business pour le flux de définition de document RosettaNet ou ebMS, selon le cas.

La fonction business-to-business source doit être la suivante :

Package : Backend Integration (1.0)

Protocole : XMLEvent (1.0)

Type de document : XMLEvent

La fonction business-to-business cible doit être la suivante :

Package : Backend Integration (1.0)

Protocole : XMLEvent (1.0)

Type de document : XMLEvent

2. Créez une interaction entre la source et la cible, de la manière suivante :
  - a. Cliquez sur **Administrateur du concentrateur > Configuration du concentrateur > Définition de document**.
  - b. Cliquez sur **Gestion des interactions**.
  - c. Cliquez sur **Création d'une interaction**.
  - d. Sous **Source**, développez **Package : Backend Integration (1.0) > Protocole : XMLEvent**.
  - e. Sélectionnez **Type de document : XMLEvent (1.0)**.
  - f. Répétez les étapes précédentes (étapes d et e) dans la colonne **Cible**.
  - g. Sélectionnez **Passe-système** dans la liste **Action** et cliquez sur **Sauvegarder**.
3. Procédez de la manière suivante pour activer le XMLEvent :
  - a. Cliquez sur **Administrateur du compte > Connexions**. La page Gestion des connexions s'affiche.
  - b. Sélectionnez la **source** et la **cible**.
  - c. Cliquez sur **Rechercher** pour identifier les connexions qui répondent à vos critères.
  - d. Pour activer un XMLEvent, cliquez sur **Activer**. La page Gestion des connexions s'affiche de nouveau, avec les connexions requises en vert. Cette page affiche l'emballage, le protocole et le type de document pour la source et la cible. Vous pouvez également afficher et modifier les paramètres et l'état de la connexion partenaire.

Pour désactiver un XMLEvent, vous pouvez :

- **Désactiver** le Protocole : XMLEvent dans **Administrateur du concentrateur > Définition de document > Gestion des définitions de document**. OU
- Procéder de la manière suivante :
  1. Cliquez sur **Administrateur du compte > Liste d'exclusion**.
  2. Sélectionnez le **nom du partenaire**.
  3. Cliquez sur **Edition**.
  4. Cochez la case **Ne pas renvoyer de notification d'événement au système dorsal**.

### **Notification d'événement**

Pour le traitement des messages RosettaNet et ebMS, il existe des circonstances où un XML Event est envoyé au dorsal ou bien reçu du dorsal. Pour les circonstances spécifiques dans lesquelles vous êtes amené à utiliser un XML Event dans ces protocoles, consultez «Notification d'événement», à la page 16 pour le traitement RosettaNet et «Notification d'événement», à la page 20 pour le traitement ebMS.

### **Structure des messages d'événement**

Tout message de notification d'événement dispose d'un en-tête de niveau de transfert standard et d'une zone x-aux-process-type définie par la valeur XMLEvent. Cependant, les données utiles du message ont une structure spécifique, comme le montre schéma XML dans la figure 11.

```

<?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>

```

Figure 11. Schéma XML pour un message de notification d'événement

Le tableau 5 décrit chaque zone contenue dans les données utiles de l'événement.

*Tableau 5. Zones de la notification d'événement XML*

<b>Zone</b>	<b>Description</b>
StatusCode	Type de message. Pour connaître quelles sont les valeurs valides, consultez «Codes d'événement employés par WebSphere Partner Gateway», à la page 25.
StatusMessage	Description alphanumérique de ce message de notification d'événement
EventMessageID	ID alphanumérique de ce message de notification d'événement particulier.
BusinessObjectID	Identificateur x-aux-msg-id contenu dans l'en-tête de niveau de transfert du message affecté par ce message de notification d'événement. Cette zone relie les données utiles du message d'origine à cet événement.
GlobalMessageID	Identificateur x-aux-system-msg-id contenu dans l'en-tête de niveau de transfert du message ayant déclenché ce message de notification d'événement.
Timestamp	Utilisé lorsqu'un événement se produit à l'aide du format d'horodatage UTC: CCYY-MM-DDThh:mm:ssZ  comprenant la précision fractionnée des secondes (...ss.ssssZ). L'horodatage doit être conforme au type de données du schéma XML pour la zone dateTime (w3.org/TR/2001/REC-xmlschema-2-20010502#dateTime)

## Exemple de notification d'événement

La figure 12 montre un exemple de message de notification d'événement envoyé par le protocole HTTP.

```
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: evtmf: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"?>
<evtmf:EventNotification xmlns:evtmf=
  "http://www.ibm.com/websphere/bcg/2003/v1.0/xmleventnotification">
  <evtmf:StatusCode>100</evtmf:StatusCode>
  <evtmf:StatusMessage>The message was delivered</evtmf:StatusMessage>
  <evtmf:EventMessageID>12345</evtmf:EventMessageID>
  <evtmf:BusinessObjectID>34234</evtmf:BusinessObjectID>
  <evtmf:GlobalMessageID>98732</evtmf:GlobalMessageID>
  <evtmf:Timestamp>2001-01-31T13:20:00Z</evtmf:Timestamp>
</evtmf:EventNotification>
```

Figure 12. Exemple de message de notification d'événement envoyé via le protocole HTTP

## Codes d'événement employés par WebSphere Partner Gateway

Cette section présente une synthèse de tous les codes d'événement employés dans WebSphere Partner Gateway. Pour savoir comment ces codes d'événement sont utilisés dans les différents protocoles, consultez «Notification d'événement», à la page 16 pour les protocoles RosettaNet et «Notification d'événement», à la page 20 pour les protocoles ebMS.

tableau 6 récapitule les codes d'événement employés par WebSphere Partner Gateway.

Tableau 6. Codes d'événement employés par WebSphere Partner Gateway

Code de statut d'événement	Protocole utilisé dans	Envoyé par	Envoyé à
100	RosettaNet ebMS	WebSphere Partner Gateway	Dorsal
800	RosettaNet	Dorsal	WebSphere Partner Gateway
801	RosettaNet	Dorsal	WebSphere Partner Gateway
900	RosettaNet ebMS	WebSphere Partner Gateway	Dorsal
902	ebMS	WebSphere Partner Gateway	Dorsal

Tableau 6. Codes d'événement employés par WebSphere Partner Gateway (suite)

Code de statut d'événement	Protocole utilisé dans	Envoyé par	Envoyé à
903	ebMS	WebSphere Partner Gateway	Dorsal

## Quel emballage allez-vous utiliser ?

Le type d'emballage détermine le format dans lequel WebSphere Partner Gateway envoie le message au système dorsal et format dans lequel ce dernier envoie le message à WebSphere Partner Gateway.

Vous utiliserez la console de communauté pour établir la connexion avec vos partenaires externes et spécifier l'emballage utilisé entre WebSphere Partner Gateway et le système dorsal. Pour déterminer le type d'emballage à utiliser, vous devez être en mesure de répondre aux questions suivantes :

- Quels types d'emballage peuvent être utilisés avec un système dorsal?
- Quels sont les types d'emballage adaptés à un message dans un protocole de gestion spécifique ?

Pour plus d'informations sur la configuration de connexions de partenaires, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

Tous les types d'emballage ne sont pas adaptés lors de l'utilisation de WebSphere Partner Gateway pour l'intégration. Le tableau 7 répertorie les types d'emballage appropriés lorsque WebSphere Partner Gateway échange des documents ou des messages avec une application dorsale du partenaire interne.

Tableau 7. Types d'emballage appropriés pour l'intégration dorsale

Type d'emballage	Description
Aucun emballage	Envoie le message au système dorsal ou au concentrateur <i>sans</i> données d'en-tête
Emballage Backend Integration	Ajoute des attributs supplémentaires à l'en-tête du message et regroupe le contenu du message dans une enveloppe XML

**Remarque :** D'autres types d'emballage (tels que AS) sont disponibles dans WebSphere Partner Gateway. Toutefois, pour l'intégration dans des applications dorsales, il est recommandé d'utiliser uniquement les valeurs "emballage Backend Integration" et "aucun emballage".

## Aucun emballage

Lorsque l'emballage a la valeur Aucun, WebSphere Partner Gateway n'ajoute aucun en-tête de niveau transport lorsqu'il envoie un message à une application dorsale et ne s'attend pas non plus à en obtenir un lorsqu'il reçoit un message provenant d'une application dorsale. Au lieu de cela, WebSphere Partner Gateway envoie uniquement le message au système dorsal. Les informations contenues dans le document contrôlent le routage des données.

## Regroupement d'intégration dorsale

Lorsque l'empaquetage a la valeur Backend Integration, les messages envoyés à ou reçus par une application dorsale contiennent les éléments suivants :

- un en-tête de niveau transport qui contient des méta-informations relatives au message (obligatoire) ;
- des données utiles qui contiennent le corps du message (obligatoire) ;
- une pièce jointe (facultatif)

L'en-tête et les données utiles sont obligatoires tandis que les pièces jointes sont facultatives. Les sections suivantes décrivent chaque élément constitutif d'un document utilisant l'empaquetage Backend Integration.

### Contenu de l'en-tête de niveau transport

L'en-tête de niveau transport contient les informations utilisées par WebSphere Partner Gateway pour traiter et router le message vers la destination appropriée. L'en-tête de niveau transport est bidirectionnel de sorte que tous les messages qui entrent et sortent de WebSphere Partner Gateway comportent les zones obligatoires et facultatives qui s'appliquent.

Le tableau 8 répertorie les zones de l'en-tête de niveau transport.

Tableau 8. Zones de l'en-tête de niveau transport

Zone d'en-tête	Description	Obligatoire
x-aux-sender-id	ID de l'expéditeur du message, tel qu'un numéro DUNS	Oui
x-aux-receiver-id	ID du récepteur du message, tel qu'un numéro DUNS	Oui
x-aux-protocol	Protocole du contenu du message. Les valeurs valides sont RNSC pour le contenu du service RosettaNet, XMLEvent et Binary. Pour WebSphere Partner Gateway, la valeur contenue dans cette zone a priorité sur n'importe quelle autre zone du protocole dans les données utiles.	Oui
x-aux-protocol-version	Version du protocole du contenu du message	Oui
x-aux-process-type	Processus à réaliser ou type de message à envoyer. Pour les messages RosettaNet, il s'agit du code PIP (par exemple, 3A4). Pour les messages d'événement, il s'agit de la valeur XMLEvent et pour les messages binaires, il s'agit de la valeur Binary. Pour WebSphere Partner Gateway, la valeur contenue dans cette zone a priorité sur n'importe quelle autre zone du processus dans les données utiles.	Oui
x-aux-process-version	Version du processus. Pour les messages RosettaNet, il s'agit du numéro de version du PIP.	Oui
x-aux-create-datetime	Lorsqu'un message a été correctement posté avec le format d'horodatage UTC (CCYY-MM-DDThh:mm:ssZ)	
x-aux-msg-id	ID du contenu des données utiles. Il peut, par exemple, s'agir de l'ID d'instance RNPIPSERVICECONTENT d'un message RosettaNet ou de l'ID document du propriétaire. Il relie les données utiles du message à un élément contenu dans le système de l'expéditeur du message pour des besoins de traçage.	
x-aux-production	Routing du message. Les valeurs valides sont les suivantes : Production et Test. Ces valeurs sont renseignées pour les requêtes effectuées dans les deux sens. Notez que lorsque le message est une réponse à un PIP bidirectionnel lancé par un partenaire externe, WebSphere Partner Gateway utilise la valeur du GlobalUsageCode de la demande et ignore la valeur contenue dans l'en-tête de niveau transport.	
x-aux-system-msg-id	Identificateur global unique (GUID) du message, utilisé pour la vérification de la duplication	Oui

Tableau 8. Zones de l'en-tête de niveau transport (suite)

Zone d'en-tête	Description	Obligatoire
x-aux-payload-root-tag	Elément de balise racine des données utiles. Par exemple, pour le contenu du service RosettaNet 3A4, la valeur de cette zone doit être Pip3A4PurchaseOrderRequest. Pour les messages de notification d'événement, la valeur de cette zone doit être EventNotification.	
x-aux-process-instance-id	ID qui relie les documents contenus dans un processus métier multiple de messages à une instance de processus unique. Pour RosettaNet, cette valeur doit être unique pour les processus RosettaNet pendant les 30 derniers jours. Tous les messages échangés en tant que partie intégrante de l'instance d'un processus RosettaNet, notamment les tentatives de relance, utilisent le même ID d'instance de processus.	
x-aux-event-status-code	Code d'état de la notification d'événement. Voir la zone StatusCode à la section «Structure des messages d'événement», à la page 22.	
x-aux-third-party-bus-id	ID tel que le numéro DUNS de la partie à l'origine de l'envoi du message. Il peut être différent des identificateurs x-aux-sender-id et x-aux-receiver-id si un tiers héberge WebSphere Partner Gateway pour le compte du propriétaire de communauté.	
x-aux-transport-retry-count	Nombre de tentatives infructueuses pour transmettre ce message avant cette tentative. Si un message est transmis à la première tentative, la valeur de cette zone sera 0.	
x-aux-in-file-name	Nom du fichier d'origine pour les messages envoyés à WebSphere Partner Gateway via JMS ou HTTP. (Voir la remarque 3, à la page 29.)	Non
x-out-filename	Nom du fichier d'origine pour les messages envoyés via JMS avec l'emballage Backend Integration. (Voir la remarque 2, à la page 29.)	Non
content-type	Type de contenu du message	
content-length	Longueur du message (en octets)	
x-aux-SyncResponse	Cet attribut est employé pour informer le dorsal que SyncResponse est nécessaire pour cette demande. Si cette réponse synchrone n'est pas requise du côté dorsal, elle sera absente. Les valeurs admises sont les suivantes : <b>Remarque :</b> Toutes les valeurs de cet attribut doivent être en minuscules. <ul style="list-style-type: none"> <li>• <b>signalonly</b> : le signal peut être un signal d'acceptation ou un signal d'exception. Déterminer le type du signal à émettre relève du dorsal. Si la valeur de l'attribut est signalonly, le partenaire expéditeur ne s'attend à recevoir que des signaux via une connexion synchrone.</li> <li>• <b>responseonly</b>: un document réponse constitue une réponse métier pour le document reçu. Si la valeur de l'attribut est responseonly, le partenaire expéditeur s'attend à recevoir une réponse via une connexion synchrone.</li> <li>• <b>signalsandresponse</b>: dans ce cas, les signaux comme la réponse devront être envoyés via une connexion synchrone. Cet attribut est un attribut d'objet routage. Il est présent dans CPA et peut être défini au niveau de la connexion du partenaire si aucun CPA n'est présent.</li> </ul>	Non
x-aux-TimeToAccept	Cet attribut, utilisé dans le flux ebMS, contiendra une valeur de "timeToAcknowledgeAcceptance" dans CPA. Lorsque CPA est chargé, cette valeur est définie comme attribut TimeToPerform as Source Document Definition. La valeur de cet attribut est un nombre en minutes. Il indique la période au terme de laquelle la partie réceptrice doit avoir accusé réception de l'acceptation sur la forme d'un document de gestion (ce qui signifie qu'il a réussi la validation au regard des règles de gestion).	Non

Tableau 8. Zones de l'en-tête de niveau transport (suite)

Zone d'en-tête	Description	Obligatoire
x-aux-IntelligibleCheckRequired	Cet attribut est utilisé dans le flux ebMS. Les valeurs possibles sont <b>oui</b> et <b>non</b> . Si la valeur est "oui", le dorsal doit vérifier qu'un document de gestion n'est pas tronqué (qu'il a donc réussi la validation de schéma) avant de renvoyer un signal d'accusé de réception.	

**Remarques :**

1. Pour des raisons de compatibilité avec IBM WebSphere MQ (un fournisseur JMS), on utilisera des traits de soulignements au lieu de traits d'union dans les zones des messages transmis via le protocole JMS. Par exemple, dans un message JMS, la zone x\_aux\_sender\_id remplace la zone x-aux-sender-id.
2. Si la passerelle est spécifiée comme étant HTTP et que la valeur du package est Aucun, le nom de fichier d'origine est défini dans les en-têtes HTTP en tant que "Content-Disposition: attachment;po.xml".  
Si le récepteur est spécifié comme JMS et que la valeur de l'emballage est Backend Integration, le nom de fichier d'origine est écrit dans x-out-filename avec d'autres en-têtes x-aux-\*.
3. Si le récepteur est spécifié comme HTTP et que la valeur de package est Aucun, le nom de fichier d'origine est défini dans les en-têtes HTTP comme "Content-Disposition: attachment;po.xml,"  
Si le récepteur est spécifié comme JMS et que la valeur de package est Aucun, le nom de fichier d'origine est défini dans les en-têtes HTTP comme "x: attachment;po.xml." Dans le cas du récepteur HTTP, il est défini sur "Content-Disposition: attachment;po.xml."

Le tableau 8 présente les informations de l'en-tête de niveau transport. Les sections suivantes fournissent les informations de l'en-tête de niveau transport spécifiques à certains protocoles de gestion :

- «En-tête de niveau transport et message RosettaNet»
- «En-tête de niveau transport et message AS2», à la page 30
- «En-tête de niveau transport et message AS1», à la page 31

**En-tête de niveau transport et message RosettaNet :** Le tableau 9 décrit l'emplacement où WebSphere Partner Gateway extrait les valeurs pour les zones de l'en-tête de niveau transport à partir d'un message RosettaNet.

Tableau 9. Zones de l'en-tête de niveau transport et contenu RosettaNet

Zone d'en-tête	Source de la valeur : RosettaNet 2.0	Source de la valeur : RosettaNet 1.1
x-aux-sender-id	<(DeliveryHeader)> <messageSenderIdentification> <PartnerIdentification> <GlobalBusinessIdentifier>	<ServiceHeader> <ProcessControl> <TransactionControl> <ActionControl> or <SignalControl> <PartnerRouter> <fromPartner> <PartnerDescription> <BusinessDescription> <GlobalBusinessIdentifier>

Tableau 9. Zones de l'en-tête de niveau transport et contenu RosettaNet (suite)

Zone d'en-tête	Source de la valeur : RosettaNet 2.0	Source de la valeur : RosettaNet 1.1
x-aux-receiver-id	<(DeliveryHeader)> <messageReceiverIdentification> <PartnerIdentification> <GlobalBusinessIdentifier>	<ServiceHeader> <ProcessControl> <TransactionControl> <ActionControl> or <SignalControl> <PartnerRouter> <toPartner> <PartnerDescription> <BusinessDescription> <GlobalBusinessIdentifier>
x-aux-protocol	Définir une valeur pour RosettaNet : RNSC	Identique pour RosettaNet 2.0
x-aux-protocol-version	Définir une valeur : 1.0	Identique pour RosettaNet 2.0
x-aux-process-type	La variable XPath source est : /ServiceHeader/ProcessControl/ pipCode/GlobalProcessIndicatorCode	La variable XPath source est : /ServiceHeader/ProcessControl/ ProcessIdentity/GlobalProcessIndicatorCode
x-aux-process-version	La variable XPath source est : /ServiceHeader/ProcessControl/ pipVersion/VersionIdentifier  La valeur de l'ID de version de chaque PIP se trouve dans la spécification PIP qui lui est associée.	La variable XPath source est : /ServiceHeader/ProcessControl/ ProcessIdentity/VersionIdentifier  La valeur de l'ID de version de chaque PIP se trouve dans la spécification PIP qui lui est associée.
x-aux-payload-root-tag	Nom du PIP, par exemple Pip3A4PurchaseOrderRequest	Identique pour RosettaNet 2.0
x-aux-process-instance-id	Pour les processus lancés par l'application, cette valeur correspond à l'ID d'instance du processus. Pour les processus lancés par un partenaire externe qui ne correspondent pas à des flux de travaux passe-système, cette valeur correspond à l'ID processus de la demande RosettaNet initiale :  <ServiceHeader> <ProcessControl> <pipInstanceId> <InstanceIdentifier>	<ServiceHeader> <ProcessControl> <ProcessIdentity> <InstanceIdentifier>
x-aux-msg-id	<(RNPipServiceContent)> <thisDocumentIdentifier> <ProprietaryDocumentIdentifier>	Identique pour RosettaNet 2.0
x-aux-production	<ServiceHeader> <ProcessIndicator> <GlobalUsageCode>	<Preamble> <GlobalUsageCode>

**En-tête de niveau transport et message AS2 :** Le tableau 10 décrit l'emplacement où WebSphere Partner Gateway extrait les valeurs pour les zones de l'en-tête de niveau transport à partir d'un message AS2.

**Remarque :** Les valeurs respectent les majuscules.

Tableau 10. Zones de l'en-tête de niveau transport à partir d'un contenu AS2

Zone d'en-tête	Source de la valeur lorsqu'un partenaire externe envoie un message AS/2 au concentrateur	Source de la valeur lorsqu'un message AS2 est envoyé à un partenaire externe
x-aux-sender-id	La zone d'en-tête AS2-From du message AS2 est définie dans la zone x-aux-sender-id du message d'intégration dorsale envoyé au partenaire interne.	La zone x-aux-sender-id du message d'intégration dorsale entrant est utilisée en tant que valeur d'en-tête AS2-From du message AS2.
x-aux-receiver-id	La zone d'en-tête AS2-To du message AS2 est définie dans la zone x-aux-receiver-id du message d'intégration dorsale envoyé au partenaire interne.	La zone x-aux-receiver-id du message d'intégration dorsale entrant est utilisée en tant que valeur d'en-tête AS2-To du message AS2.
x-aux-protocol	La zone ToProtocol de la connexion du partenaire est définie dans la zone x-aux-protocol du message d'intégration dorsale envoyé au partenaire interne.	La zone x-aux-protocol du message d'intégration dorsale entrant est utilisée pour déterminer la zone FromProtocol de la connexion du partenaire.
x-aux-protocol-version	La zone ToProtocolVersion de la connexion du partenaire est définie dans la zone x-aux-protocol-version du message d'intégration dorsale envoyé au partenaire interne.	La zone x-aux-protocol-version du message d'intégration dorsale entrant est utilisée en tant que zone FromProtocolVersion de la connexion du partenaire.
x-aux-process-type	La zone ToProcessCode de la connexion du partenaire est utilisée pour définir la zone x-aux-process-type du message d'intégration dorsale envoyé au partenaire interne.	La zone x-aux-process-type du message d'intégration dorsale entrant est utilisée en tant que zone FromProcessCode de la connexion du partenaire.
x-aux-process-version	La zone ToProcessVersion de la connexion du partenaire est définie dans la zone x-aux-process-version du message d'intégration dorsale envoyé au partenaire interne.	La zone x-aux-process-version du message d'intégration dorsale entrant est utilisée en tant que zone FromProcessVersion de la connexion du partenaire.
x-aux-payload-root-tag	Pour un protocole XML personnalisé uniquement, la balise racine indiquée dans la variable XPATH du message est analysée et utilisée dans la zone x-aux-payload-root-tag.	Il est inutile de définir cette zone dans le message d'intégration dorsale entrant.
x-aux-process-instance-id	Cette zone n'est pas utilisée pour AS2.	Cette zone n'est pas utilisée pour AS2.
x-aux-msg-id	Pour un protocole XML personnalisé uniquement, l'ID document indiqué dans la variable XPATH du message est analysé et utilisé dans la zone x-aux-msg-id.	Il est inutile de définir cette zone dans le message d'intégration dorsale entrant.
x-aux-system-msg-id	Cette zone est définie par l'ID unique généré en interne pour ce message.	Il est inutile de définir cette zone dans le message d'intégration dorsale entrant.
x-aux-production	Cette zone n'est pas utilisée pour AS2.	Cette zone n'est pas utilisée pour AS2.

**En-tête de niveau transport et message AS1 :** Le tableau 11, à la page 32 décrit l'emplacement où WebSphere Partner Gateway extrait les valeurs pour les zones de l'en-tête de niveau transport à partir d'un message AS1.

**Remarque :** Les valeurs respectent les majuscules.

Tableau 11. Zones de l'en-tête de niveau transport à partir d'un contenu AS1

Zone d'en-tête	Source de la valeur lorsqu'un partenaire externe envoie un message AS/1 au concentrateur	Source de la valeur lorsqu'un message AS/1 est envoyé à un partenaire externe
x-aux-sender-id	L'élément <i>FromID</i> contenu dans la zone d'en-tête "Objet : <i>ToID;FromID</i> " du message AS1 est défini dans la zone x-aux-sender-id du message d'intégration dorsale envoyé au partenaire interne.	La zone x-aux-sender-id du message d'intégration dorsale entrant est utilisée en tant qu'élément <i>FromID</i> dans la valeur d'en-tête "Objet : <i>ToID;FromID</i> " du message AS1.
x-aux-receiver-id	L'élément <i>ToID</i> contenu dans la zone d'en-tête "Objet : <i>ToID;FromID</i> " du message AS1 est défini dans la zone x-aux-receiver-id du message d'intégration dorsale envoyé au partenaire interne.	La zone x-aux-receiver-id du message d'intégration dorsale entrant est utilisée en tant qu'élément <i>ToID</i> dans la valeur d'en-tête "Objet : <i>ToID;FromID</i> " du message AS1.
x-aux-protocol	La zone <i>ToProtocol</i> de la connexion du partenaire est définie dans la zone x-aux-protocol du message d'intégration dorsale envoyé au partenaire interne.	La zone x-aux-protocol du message d'intégration dorsale entrant est utilisée en tant que zone <i>FromProtocol</i> de la connexion du partenaire.
x-aux-protocol-version	La zone <i>ToProtocolVersion</i> de la connexion du partenaire est définie dans la zone x-aux-protocol-version du message d'intégration dorsale envoyé au partenaire interne.	La zone x-aux-protocol-version du message d'intégration dorsale entrant est utilisée en tant que zone <i>FromProtocolVersion</i> de la connexion du partenaire.
x-aux-process-type	La zone <i>ToProcessCode</i> de la connexion du partenaire est définie dans la zone x-aux-process-type du message d'intégration dorsale envoyé au partenaire interne.	La zone x-aux-process-type du message d'intégration dorsale entrant est utilisée en tant que zone <i>FromProcessCode</i> de la connexion du partenaire.
x-aux-process-version	La zone <i>ToProcessVersion</i> de la connexion du partenaire est définie dans la zone x-aux-process-version du message d'intégration dorsale envoyé au partenaire interne.	La zone x-aux-process-version du message d'intégration dorsale entrant est utilisée en tant que zone <i>FromProcessVersion</i> de la connexion du partenaire.
x-aux-payload-root-tag	Pour un protocole XML personnalisé uniquement, la balise racine indiquée dans la variable <i>XPATH</i> du message est analysée et définie dans la zone x-aux-payload-root-tag.	Il est inutile de définir cette zone dans le message d'intégration dorsale entrant.
x-aux-process-instance-id	Cette zone n'est pas utilisée pour AS1.	Cette zone n'est pas utilisée pour AS1.
x-aux-msg-id	Pour un protocole XML personnalisé uniquement, l'ID document indiqué dans la variable <i>XPATH</i> du message est analysé et utilisé dans la zone x-aux-msg-id.	Il est inutile de définir cette zone dans le message d'intégration dorsale entrant.
x-aux-system-msg-id	Cette zone est définie par l'ID unique généré en interne pour ce message.	Il est inutile de définir cette zone dans le message d'intégration dorsale entrant.
x-aux-production	Cette zone n'est pas utilisée pour AS1.	Cette zone n'est pas utilisée pour AS1.

**En-tête de niveau transport et message ebMS :** Le tableau 11 décrit l'emplacement d'où WebSphere Partner Gateway extrait les valeurs des zones d'en-tête de niveau transport au sein d'un message ebMS.

**Remarque :** Les valeurs respectent les majuscules.

Tableau 12. Zones de l'en-tête de niveau transport issues du contenu ebMS

Zone d'en-tête	Source de la valeur lorsqu'un partenaire externe envoie un document ebMS au concentrateur	Source de la valeur lorsqu'un document ebMS est envoyé à un partenaire externe
x-aux-receiver-id	(En-tête SOAP) <eb:MessageHeader><eb:To><eb:PartyId>	La valeur de cet attribut provient du dorsal en tant qu'en-têtes du dorsal
x-aux-sender-id	(En-tête SOAP) <eb:MessageHeader><eb:From><eb:PartyId>	La valeur de cet attribut provient du dorsal en tant qu'en-têtes du dorsal
x-aux-SyncResponse	Provient de l'attribut Source Document Definition bcg.ro.ebxml.SYNCREPLYMODE  Les valeurs possibles sont : <ul style="list-style-type: none"> <li>• responseOnly</li> <li>• signalAndResponse</li> <li>• signalOnly</li> </ul>	Provient de l'attribut Target Document Definition
x-aux-process-instance-id	(En-tête SOAP) <eb:MessageHeader><eb:ConversationId>	Provient du dorsal en tant qu'en-têtes du dorsal. Si le dorsal ne le définit pas, c'est WebSphere Partner Gateway qui le générera  L'élément obligatoire ConversationId du flux ebMS est une chaîne qui identifie le jeu des messages liés les uns aux autres qui constituent la conversation entre les deux parties. Les messages d'une même conversation auront le même ID de conversation, mais des ID de messages différents.
x-aux-process-type	(En-tête SOAP) <eb:MessageHeader><eb:Service>	Provient du dorsal en tant qu'en-têtes du dorsal.
x-aux-process-version	(En-tête SOAP) <eb:MessageHeader><eb:Service> Texte de la valeur de l'élément après le caractère "\$". Si la valeur de l'élément ne contient pas de caractère "\$" ou n'est pas un URI, ce sera par défaut "ALL".	Provient du dorsal en tant qu'en-têtes du dorsal.
x-aux-protocol	"EBMS"	Provient du protocole de la cible sur la vue Connexion du partenaire
x-aux-protocol-version	(En-tête SOAP) <eb:MessageHeader><eb:version>	Provient du protocole de la cible sur la vue Connexion du partenaire
x-aux-third-party-bus-id	<b>Console &gt; Administration Système&gt; Propriétés DocMgr</b>	N/A
x-aux-TimeToAccept	Provient de l'attribut Source Document Definition	
x-aux-system-msg-id	Cette zone est définie par l'ID unique généré en interne pour ce message.	Il est inutile de définir cette zone dans le message d'intégration dorsale entrant.

Tableau 12. Zones de l'en-tête de niveau transport issues du contenu ebMS (suite)

Zone d'en-tête	Source de la valeur lorsqu'un partenaire externe envoie un document ebMS au concentrateur	Source de la valeur lorsqu'un document ebMS est envoyé à un partenaire externe
x-aux-msg-id	<soapenv:Header><eb:MessageHeader> <eb:MessageData><eb:MessageId>	Il peut être défini par le dorsal comme en-têtes de dorsal. Si le dorsal ne le définit pas, c'est WebSphere Partner Gateway qui le générera en interne.
x-aux-IntelligibleCheckRequired	Provient de l'attribut Source Document Definition bcg.ro.ebxml.ISINTELLIGIBLECHECKREQUIRED	N/A

Exemple d'en-tête de transport ebMS :

```

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 : 1158164092823000D606BBA3900133200000000000000021
ReferenceId : 1158164101686000D606BBA3900133200000000000000003
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
    
```

**Données utiles**

Les données utiles du message correspondent au contenu réel du message. L'emplacement des données utiles dépend du protocole de transport qui envoie le message, comme le montre le tableau 13.

Tableau 13. Emplacement des données utiles

Protocole de transport	Emplacement des données utiles
Messages du protocole HTTP	Dans le corps de la requête HTTP Post
Messages du protocole JMS	Dans le corps du message JMS
messages RosettaNet	Dans le contenu du service à partir du PIP
EDI	Enveloppe EDI
document ROD/à plat ou XML	Le document ROD/à plat ou XML

Les données utiles peuvent être encodées en Base64 et regroupées dans une *enveloppe de transport XML* dans les cas suivants :

- Si le document contient une pièce jointe  
Un document doté de pièces jointes *doit* être inséré dans une enveloppe XML. Pour plus d'informations sur les pièces jointes, voir «Pièces jointes», à la page 35.

- Si l'indicateur de l'enveloppe d'empaquetage Backend Integration est positionné sur Oui

Pour encapsuler un document dans une enveloppe XML *sans tenir compte* s'il contient des pièces jointes, positionner l'indicateur de l'enveloppe Backend integration sur Oui dans la vue Fonctions business-to-business du profil. Par exemple, pour définir l'indicateur dans le profil du partenaire interne, effectuez les tâches suivantes :

1. Cliquez sur **Administrateur du compte > Profils**.
2. Entrez le nom du partenaire interne (ou effectuez une recherche sur tous les partenaires externes).
3. Cliquez sur l'icône **Afficher les détails** en regard du nom du partenaire interne.
4. Cliquez sur **Fonctions business-to-business**.
5. Cliquez sur l'icône **Edition** à côté d'**Intégration Dorsale**.
6. Affectez à l'**indicateur d'enveloppe** la valeur **Oui**.

Cette enveloppe XML insère le document concerné dans la balise racine <transport-enveloppe>. A l'intérieur de cette balise racine se trouve une balise <payload> qui contient les données utiles du document. Si ce document contient des pièces jointes, celles-ci se trouvent dans une étiquette <attachment>. Pour plus d'informations sur la structure de ces étiquettes, voir «Pièces jointes».

WebSphere Partner Gateway inclut le fichier schéma XML W3C suivant qui décrit la structure d'enveloppe XML de l'intégration dorsale :

wbipackaging\_v1.1\_ns.xsd

**Remarque :** dans le cas des flux ebMS, c'est l'empaquetage wbipackaging\_v1.2\_ns.xsd qu'il faut utiliser.

Ce fichier schéma se trouve dans le répertoire suivant sur le support d'installation :

B2BIntegrate\packagingSchemas

Vous pouvez utiliser n'importe quel outil d'édition XML pour valider l'intégration dorsale XML à la place de ce fichier schéma afin de garantir la validité du document avant de l'envoyer au Gestionnaire de documents.

## Pièces jointes

Si le protocole de messagerie le permet, chaque document peut avoir une ou plusieurs pièces jointes. Si le document contient des pièces jointes, celui-ci *doit* être inséré dans une enveloppe XML, comme stipulé dans la section «Données utiles», à la page 34. Le tableau 14 décrit les attributs XML dans les balises de données utiles et de pièces jointes.

Tableau 14. Attributs XML des balises de données utiles et de pièces jointes

Attributs XML	Description	Obligatoire
Content-Type	Identifie le type/sous-type MIME, tel que text/xml ou image/gif.	Oui

Tableau 14. Attributs XML des balises de données utiles et de pièces jointes (suite)

Attributs XML	Description	Obligatoire
Encoding	Identifie le codage. Dans la mesure où les pièces jointes et les données utiles doivent être cryptées en Base64, la seule valeur valide pour cet attribut est "Base64".	Non

La figure 13 présente un exemple de document inséré dans une enveloppe XML comprenant des données utiles et une pièce jointe.

**Remarque :** L'espace de nom est requis dans cet exemple :

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

```
<?xml version="1.0" encoding="utf-8"?>
<transport-enveloppe
  xmlns="http://www.ibm.com/websphere/bcg/2003/v1.0/wbipackaging">
  <payload encoding="base64" contentType="application/xml">
    ...base64 encoded XML message...
  </payload>
  <attachment encoding="base64" Content-Type="text/xml">
    ...base64 encoded XML attachment...
  </attachment>
</transport-enveloppe>
```

Figure 13. Exemple d'enveloppe XML comprenant des données utiles et une pièce jointe

**Remarques :**

1. Pour traiter les documents encapsulés dans une enveloppe XML via le WebSphere Interchange Server, WebSphere Partner Gateway inclut un gestionnaire de pièces jointes. Pour plus d'informations, voir «Gestion des documents contenant des pièces jointes», à la page 132.
2. Pour traiter les documents avec pièces jointes sur WebSphere Process Server, WebSphere Partner Gateway offre la liaison de données de l'empaquetage Backend Integration. Voir «Gestion des messages avec empaquetage Backend Integration», à la page 66.

## Quel type d'empaquetage peut être utilisé pour vos documents ?

Les documents de certains protocoles de gestion peuvent uniquement utiliser certains types d'empaquetage. Par exemple, un document RosettaNet ou ebMS peut *uniquement* être traité lorsque l'empaquetage Backend integration a été spécifié. Voir le tableau 16, à la page 38, le tableau 17, à la page 38 et le tableau 18, à la page 39 pour obtenir la liste complète des types de document et des types d'empaquetage auxquels ils peuvent être associés.

## Exemple d'empaquetage Backend Integration sur HTTP

La figure 14 présente un exemple de message provenant de WebSphere Partner Gateway à destination d'une application utilisant le protocole de transport HTTP. Notez que le message ne contient aucune pièce jointe.

```

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>

```

Figure 14. Exemple de message utilisant le protocole de transport HTTP

---

## Quel mode de transport de messages allez-vous utiliser ?

Lorsque l'application dorsale et WebSphere Partner Gateway s'envoient des messages entre eux, chacun doit utiliser le même protocole de transport des messages. Le *protocole de transport des messages* définit le protocole de communication utilisé pour l'envoi des messages.

WebSphere Partner Gateway communique avec une application dorsale via son interface d'intégration dorsale. Le tableau 15 fournit la liste des protocoles de transport compatibles avec l'interface Backend Integration.

Tableau 15. Protocoles de transport compatibles avec l'intégration dorsale

Protocole de transport	Pour plus d'informations
HTTP ou HTTPS	«Protocole de transport HTTP», à la page 40
Fichiers de système de fichiers	«Protocole de système de fichiers», à la page 44
JMS	«Protocole JMS», à la page 41

Le tableau 16, à la page 38 indique les protocoles de transport pris en charge pour les types d'empaquetage et les protocoles de gestion lorsque le concentrateur envoie des documents au système dorsale.

Tableau 16. Protocoles de transport pris en charge de WebSphere Partner Gateway au système dorsal

Type d’empaquetage	Protocole de gestion	HTTP ou HTTPS ?	JMS ?	Système de fichiers ?
Backend Integration	RosettaNet (RNSC)	Oui	Oui	Non
	ebMS	Oui	Oui	Non
	Binaire	Oui	Oui	Non
	EDI (voir le tableau 18, à la page 39 pour plus d’informations sur EDI)			
	XML	Oui	Oui	Non
	ROD/à plat	Oui	Oui	Non
Aucun	EDI (voir le tableau 18, à la page 39 pour plus d’informations sur EDI)			
	cXML uniquement	Oui	Non	Non
	SOAP uniquement	Oui	Non	Non
	Binaire	Oui	Oui	Oui
	XML	Oui	Oui	Oui
	ROD/à plat	Oui	Oui	Oui

Le tableau 17 indique les protocoles de transport pris en charge pour les types d’empaquetage et les protocoles de gestion lorsque le système dorsal envoie des documents au concentrateur.

Tableau 17. Protocoles de transport pris en charge du système dorsal à WebSphere Partner Gateway

Type d’empaquetage	Protocole de gestion	HTTP ou HTTPS ?	JMS ?	Système de fichiers ?
Backend Integration	RosettaNet (RNSC)	Oui	Oui	Non
	ebMS	Oui	Oui	Non
	XML	Oui	Oui	Non
	Binaire	Oui	Oui	Non
	ROD/à plat	Oui	Oui	Non
Aucun	XML uniquement	Oui	Oui	Oui
	EDI (voir le tableau 18, à la page 39 pour plus d’informations sur EDI)			
	cXML uniquement	Oui	Non	Non
	SOAP uniquement	Oui	Non	Non
	Binaire uniquement	Non	Non	Non
	ROD/à plat uniquement	Oui	Oui	Oui

Le tableau 18, à la page 39 montre les protocoles de transport et les types d’empaquetage pris en charge pour divers documents EDI, XML et ROD (données orientées enregistrement)/fichiers à plat.

Tableau 18. Protocoles de transport pris en charge entre WebSphere Partner Gateway et le système dorsal pour EDI

Type d'emballage	Document	HTTP ou HTTPS ?	JMS ?	Système de fichiers ?
Backend Integration	Echange simple contenant une seule transaction (telle qu'une transaction X12 850 dans une enveloppe)	Oui	Oui	Non
	Interchange simple contenant plusieurs transactions (telles qu'une transaction X12 850 et une transaction X12 890 dans la même enveloppe)	Oui	Oui	Non
	Echanges multiples contenant une transaction (telles que deux enveloppes X12 dans le même fichier, contenant chacune une transaction)	Oui	Oui	Non
	Plusieurs échanges contenant plusieurs transactions (telles que deux enveloppes X12 dans le même fichier, contenant chacune au moins deux transactions)	Oui	Oui	Non
	Transaction EDI (par exemple, une transaction X12 850), qui ne peut pas être envoyée seule car elle doit se trouver dans un échange EDI	Non	Non	Non
	Document (par exemple, XML) ultérieurement converti en transaction EDI	Oui	Oui	Non
Aucun	Echange unique contenant une seule transaction	Oui	Oui	Oui
	Interchange unique contenant plusieurs transactions	Oui	Oui	Oui
	Plusieurs échanges contenant une seule transaction	Oui	Oui	Oui
	Plusieurs échanges contenant plusieurs transactions	Oui	Oui	Oui
	La transaction EDI (non prise en charge ; doit avoir une enveloppe Interchange)	Non	Non	Non
	Document (par exemple, XML) ultérieurement converti en transaction EDI	Oui	Oui	Oui

Les tableaux précédents répertorient les protocoles de transport valides entre le concentrateur et le système dorsal. Le concentrateur peut utiliser d'autres protocoles de transport pour envoyer des documents aux partenaires et en recevoir de ces derniers. Par exemple, le concentrateur peut envoyer un document à un serveur FTP distant via le transport de scriptsFTP. Il peut également recevoir des documents via le transport de scripts FTP. Le transport de scripts FTP, décrit dans

le *guide de configuration du concentrateur WebSphere Partner Gateway*, peut être utilisé pour envoyer et recevoir des documents via Internet mais il doit être utilisé pour envoyer et recevoir des documents depuis des réseaux à valeur ajoutée.

## Protocole de transport HTTP

Pour envoyer des messages à l'aide d'un protocole HTTP, WebSphere Partner Gateway utilise HTTP/S 1.1. Pour recevoir des messages en provenance des systèmes dorsaux, WebSphere Partner Gateway prend en charge HTTP/S en version 1.0 et 1.1.

Le message HTTP peut inclure des attributs d'empaquetage d'intégration. Ces attributs seront inclus dans ce message en fonction du type d'empaquetage associé à la connexion du partenaire, tel que décrit ci-dessous :

- Si la connexion du partenaire spécifie que le message HTTP inclut Empaquetage Backend Integration, l'en-tête de niveau transport du message HTTP inclut des attributs supplémentaires contenant des informations sur le message, telles que le protocole du contenu, l'ID ou encore l'expéditeur du message. Pour obtenir une liste complète des zones contenues dans l'en-tête, voir «Contenu de l'en-tête de niveau transport», à la page 27.

Les messages RosettaNet doivent utiliser l'empaquetage Backend Integration.

- Si la connexion du partenaire spécifie Aucun empaquetage n'a été utilisé, le message HTTP ne possède *aucun* attribut supplémentaire et WebSphere Partner Gateway doit analyser le message pour obtenir ces informations.

Les messages SOAP et cXML ne doivent utiliser Aucun empaquetage.

**Remarque :** Les messages XML peuvent à la fois utiliser l'empaquetage Backend Integration ou n'en utiliser aucun. De façon similaire, les documents EDI peuvent utiliser l'empaquetage Backend Integration ou n'en utiliser aucun. Les messages binaires reçus du système dorsal doivent utiliser l'empaquetage Backend Integration. Toutefois, l'inverse n'est pas vrai car WebSphere Partner Gateway prend en charge l'envoi des messages binaires vers l'application dorsale au moyen de n'importe quel type d'empaquetage.

## Processus

Lorsque des messages HTTP ou HTTPS sont envoyés entre WebSphere Partner Gateway et une application dans le cadre d'échanges asynchrones, les étapes suivantes se produisent :

1. Le système source (WebSphere Partner Gateway ou le système dorsal) poste un message HTTP au système récepteur au moyen d'un URL spécifique.
2. Le système récepteur reçoit le message et envoie un accusé de réception de niveau protocole, HTTP 200 ou 202, pour signifier le changement de droits de propriété. Le système source ignore le corps de ce message d'accusé de réception. Si une erreur se produit au cours du traitement, le système récepteur renvoie un message HTTP 500 au système source.
3. Si WebSphere Partner Gateway est le système récepteur (cas de figure où WebSphere Partner Gateway reçoit le message), il conserve alors le message et libère la connexion au système source.
4. Le système récepteur peut alors traiter le message de façon asynchrone.

Lorsque l'échange est synchrone (par exemple, pour un document SOAP ou cXML), une réponse est renvoyée avec le message HTTP 200 au sein de la même connexion HTTP.

## Envoi de messages depuis le système dorsal via le protocole HTTP

Pour envoyer un message à WebSphere Partner Gateway à l'aide du protocole HTTP, l'application dorsale effectue les opérations suivantes :

1. Elle crée le message.  
L'attribut Content-Type de l'en-tête de niveau transport indique le codage utilisé pour le message.
2. Elle groupe le message conformément à l'empaquetage défini pour la connexion.  
Pour l'empaquetage Backend Integration, l'application dorsale ajoute les attributs de l'en-tête du protocole requis par WebSphere Partner Gateway.
3. Elle soumet le message à l'URL utilisée par WebSphere Partner Gateway pour recevoir ces messages.
4. Si l'échange est synchrone, l'application dorsale attend de recevoir une réponse dans la même connexion que celle utilisée pour la requête.

Pour activer les échanges de message HTTP dans cette direction, sur la page Caractéristiques récepteur de la Console de communauté, configurez un récepteur sur le concentrateur destiné aux documents entrants. Ce récepteur spécifie un URL. Le système dorsal doit connaître cette adresse pour envoyer des documents au concentrateur.

## Réception de messages sur le système dorsal via le protocole HTTP

Pour recevoir un message de WebSphere Partner Gateway à l'aide du protocole HTTP, l'application dorsale effectue les opérations suivantes :

1. Elle détecte l'arrivée d'un message sur une URL spécifique.
2. Dès réception du message, elle le traite :
  - Si la connexion ne demande aucun empaquetage, l'application dorsale doit analyser le message pour déterminer sa méthode de gestion.
  - Si la connexion utilise l'empaquetage Backend Integration, l'application peut utiliser les attributs d'intégration dorsale pour déterminer la méthode de gestion du message.
3. Si l'échange est synchrone, l'application dorsale renvoie une réponse dans la même connexion que celle utilisée pour la requête.

Pour activer les échanges de messages HTTP dans cette direction, sur la page Destination de la Console de communauté, configurez une destination sur le concentrateur indiquant où les documents devront être distribués au système dorsal.

## Protocole JMS

Le protocole JMS est basé sur le service JMS (Java<sup>TM</sup> Message Service) et transfère les messages par l'intermédiaire de files d'attente JMS transactionnelles et permanentes fournies, par exemple, par IBM WebSphere MQ. Le protocole JMS prend en charge les types de message JMS suivants :

- StreamMessage (sous forme de tableau d'octets)
- BytesMessage (sous forme de tableau d'octets)
- TextMessage

Dans le protocole JMS, un système envoie un message JMS à un autre. Une fois que le second système reçoit le message, il le supprime de la file d'attente. A partir de ce moment, le système de réception peut traiter le message de façon asynchrone

Le message JMS peut inclure des attributs d'empaquetage d'intégration. Ces attributs seront inclus dans ce message en fonction du type d'empaquetage associé à la connexion du partenaire, tel que décrit ci-dessous :

- Si la connexion du partenaire spécifie que le message JMS inclut : Empaquetage Backend Integration, le message JMS détient les informations de niveau transport (telles que le protocole du contenu, l'ID ou encore l'expéditeur du message) en tant que propriétés JMS dans le message. Pour obtenir une liste complète des propriétés, voir «Contenu de l'en-tête de niveau transport», à la page 27.

**Remarque :** Pour des besoins de compatibilité avec WebSphere MQ JMS, les propriétés des messages JMS utilisent des traits de soulignement dans les noms de propriété à la place des traits d'union. Par exemple, la propriété `x_aux_system_msg_id` d'un message JMS a pour équivalent la zone d'en-tête HTTP `x-aux-system-msg-id`. Lorsque WebSphere Partner Gateway traite un message JMS, il convertit les traits de soulignement en traits d'union dans ces propriétés.

- Si la connexion du partenaire indique Aucun empaquetage, le message JMS ne possède *aucun* des attributs complémentaires suivants.

A l'exception des messages binaires, WebSphere Partner Gateway prend en charge l'envoi et la réception des messages JMS utilisant n'importe quel type de empaquetage. Les messages binaires reçus d'une application doivent utiliser l'empaquetage Backend Integration. L'inverse n'est pas vrai car WebSphere Partner Gateway prend en charge l'envoi des messages binaires vers l'application au moyen de n'importe quel type d'empaquetage.

## Configuration de l'environnement JMS

Pour configurer votre environnement JMS, les fournisseurs ci-après sont requis.

- Fournisseur JMS

Un fournisseur JMS fournit l'implémentation du support d'API JMS pour la messagerie. Le système dorsal avec lequel vous échangez des documents détermine le fournisseur JMS à utiliser.

- Si vous échangez des documents avec WebSphere Interchange Server, vous utiliserez le fournisseur JMS WebSphere MQ.
- Si vous échangez des documents avec WebSphere Process Server, vous utiliserez WebSphere Platform Messaging et un bus d'intégration de service comme fournisseur JMS. Vous trouverez tous les détails sur les fournisseurs JMS utilisables avec WebSphere Process Server dans Chapitre 5, «Intégration de WebSphere Process Server avec le protocole de transport JMS», à la page 95.

Le fournisseur JMS est celui qui vous offre un programme pour installer votre environnement JMS. Par exemple, WebSphere MQ fournit le programme JMSAdmin, qui permet de construire des objets requis par JMS (la fabrique de connexion JMS et les objets de file d'attente JMS). Lorsque ces objets sont construits, les références à ceux-ci sont stockées dans JNDI.

**Remarque :** Pour la messagerie, WebSphere Partner Gateway prend en charge le modèle point à point uniquement.

- Fournisseur JNDI

Le fournisseur JNDI fournit l'implémentation de JNDI, qui est utilisé pour stocker les références aux objets JMS.

Pour qu'une application dorsale envoie des documents de gestion à WebSphere Partner Gateway via le protocole JMS, un récepteur JMS doit être configuré. Le récepteur JMS reçoit des messages d'une file d'attente JMS et les documents sont introduits dans le flux de travaux WebSphere Partner Gateway. La configuration du récepteur JMS inclut les paramètres requis pour l'accès à la JNDI ainsi que les noms des objets JMS. A des fins d'intégration avec le système dorsal, la file d'attente configurée dans le récepteur JMS est celle depuis laquelle ce système envoie le message JMS.

De façon similaire, une destination JMS est utilisée par WebSphere Partner Gateway pour envoyer des documents de gestion à une file d'attente dans laquelle les partenaires s'attendent à les recevoir. Par conséquent, pour envoyer des messages au système dorsal, vérifiez qu'une destination JMS est configurée dans le profil du partenaire interne. Cette destination doit être configurée pour effectuer des envois dans la file d'attente dans laquelle le système dorsal reçoit les messages. La configuration de la destination JMS inclut les paramètres requis pour l'accès à la JNDI ainsi que les noms des objets JMS.

### **Présentation de l'installation d'un environnement JMS**

Pour communiquer sur le protocole de transport JMS, WebSphere Partner Gateway et l'application dorsale requièrent une file d'attente JMS pour *chaque* direction de la communication. Par conséquent, vous devez procéder aux opérations suivantes pour fournir les files d'attente JMS :

- Configurer votre environnement JMS.
- Créer un gestionnaire de files d'attente ainsi que les autres files d'attente requises (file d'attente transmission, éloignée et réception).

Le Le gestionnaire de file d'attente JMS peut se trouver sur n'importe quel ordinateur, y compris ceux cités ci-dessous :

- l'ordinateur sur lequel réside le système dorsal ;
- l'ordinateur sur lequel réside WebSphere Partner Gateway.

De plus, un gestionnaire de files d'attente peut être installé à *la fois* sur l'ordinateur sur lequel réside le système dorsal et l'ordinateur qui héberge WebSphere Partner Gateway. Dans ce cas, utiliser des canaux de configuration destinés à relier les deux gestionnaires de files d'attente entre eux. A l'aide de cette méthode, aucune connexion client n'est nécessaire sur le réseau.

Pour plus d'informations sur les instructions de configuration du mécanisme de protocole de transport JMS avec WebSphere MQ version 6.0, (et dernier groupe de correctifs) voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*. Les instructions de configuration de l'environnement JMS pour les échanges de documents avec WebSphere Process Server sont exposées dans Chapitre 5, «Intégration de WebSphere Process Server avec le protocole de transport JMS», à la page 95.

### **Envoi de messages depuis le système dorsal via le protocole JMS**

Pour envoyer un message à WebSphere Partner Gateway à l'aide du protocole JMS, l'application dorsale effectue les opérations suivantes :

1. Elle crée le message.

2. Elle groupe le message conformément à l’empaquetage défini pour la connexion.  
Pour l’empaquetage Backend Integration, l’application ajoute les attributs d’en-tête JMS requis.
3. Elle envoie le message à la file d’attente JMS utilisée par l’application dorsale pour envoyer des messages à WebSphere Partner Gateway.

### **Réception de messages sur le système dorsal via le protocole JMS**

Pour recevoir un message de WebSphere Partner Gateway à l’aide du protocole JMS, l’application dorsale effectue les opérations suivantes :

1. Elle détecte l’arrivée d’un message sur la file d’attente JMS.
2. Dès réception du message, elle le traite :
  - Si la connexion ne demande aucun empaquetage, l’application dorsale doit analyser le message pour déterminer sa méthode de gestion.
  - Si la connexion utilise l’empaquetage Backend Integration, l’application peut utiliser les attributs d’intégration dorsale pour déterminer la méthode de gestion du message.

### **Protocole de système de fichiers**

Le protocole de système de fichiers permet à WebSphere Partner Gateway d’envoyer des messages en les plaçant dans une arborescence de répertoires définie. WebSphere Partner Gateway reçoit les messages en les lisant à partir de la structure de répertoires. Le protocole de système de fichiers prend en charge uniquement le type Aucun empaquetage.

### **Envoi de messages depuis le système dorsal via le protocole de système de fichiers**

Pour envoyer un message à WebSphere Partner Gateway à l’aide du protocole de système de fichiers, l’application effectue les opérations suivantes :

1. Elle crée le fichier de messages dans un répertoire temporaire.
2. Une fois le fichier de messages prêt, elle déplace le fichier dans le répertoire approprié interrogé par WebSphere Partner Gateway.

Pour activer les échanges de messages via le système de fichiers, sur la page Caractéristiques récepteur de la Console de communauté, installez un récepteur destiné aux documents entrants. Le récepteur du message détermine le répertoire interrogé par WebSphere Partner Gateway. Lorsque vous créez un récepteur, WebSphere Partner Gateway crée un répertoire de documents et ses sous-répertoires pour le récepteur, de la manière suivante :

```
<doc_root>  
  Documents  
    Production  
    Test  
<other destination types>
```

WebSphere Partner Gateway interroge régulièrement les répertoires de stockage des documents et leurs sous-répertoires pour détecter les fichiers de messages. S’il trouve un message, WebSphere Partner Gateway le conserve puis le supprime du répertoire. WebSphere Partner Gateway traite ensuite le message normalement. Voir le *Guide de configuration du concentrateur WebSphere Partner Gateway* pour obtenir des informations sur la création du récepteur.

## Réception de messages sur le système dorsal via le protocole de système de fichiers

Pour recevoir des messages à l'aide du protocole de système de fichiers, l'application effectue les opérations suivantes :

1. Elle interroge le répertoire approprié à la recherche de fichiers de messages.
2. Chaque fois qu'un message se présente, elle le conserve.
3. Elle supprime le message du répertoire.
4. Elle traite le message.

Pour activer les échanges de message via le système de fichiers, utilisez la page Destination de la Console de communauté pour installer une destination spécifiant l'emplacement où les documents devront être distribués. WebSphere Partner Gateway placera le fichier message dans le répertoire Documents défini par la destination. Définir le répertoire de destination en fonction de la destination permet d'avoir un répertoire différent pour chaque connexion de partenaire. Pour plus d'informations sur les destinations, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

---

## Comment accéder à l'application dorsale?

WebSphere Partner Gateway permet l'intégration de nombreux types d'applications dorsales. En règle générale, une application dorsale est accessible via un système dorsal du type courtier d'intégrations. Ce guide traite des différents types d'intégration dans les systèmes dorsaux présentés dans le tableau 19.

Tableau 19. Applications dorsales prises en charge pour WebSphere Partner Gateway

Application dorsale	Pour plus d'informations
WebSphere Process Server	Chapitre 3, «Présentation de l'intégration à WebSphere Process Server», à la page 59
WebSphere Interchange Server	Chapitre 8, «Présentation de l'intégration à InterChange Server», à la page 119
WebSphere Message Broker	Chapitre 11, «Intégration à WebSphere Message Broker», à la page 187
WebSphere Data Interchange	Chapitre 12, «Intégration à WebSphere Data Interchange», à la page 205

---

## Traitement des messages

Cette section décrit comment WebSphere Partner Gateway gère les situations suivantes qui affectent la livraison des messages :

- «Distribution en file d'attente»
- «Gestion des erreurs de transmission»
- «Messages en double», à la page 47

### Distribution en file d'attente

WebSphere Partner Gateway poste dans une file d'attente les informations relatives à tous les documents qu'il souhaite envoyer à une destination spécifique. Le Gestionnaire de documents traite ces messages dans l'ordre dans lequel la file d'attente les reçoit (premier entré, premier sorti) et utilise l'unité d'exécution de chaque message pour les envoyer. Notez que si la destination (par exemple, l'URL si le protocole de transport est HTTP ou la destination JMS si le protocole de transport est JMS) a été configurée pour être hors ligne (voir Gestion des erreurs de transmission), les messages restent dans la file d'attente jusqu'à ce que la destination soit activée (mise en ligne). Si le Gestionnaire de documents reçoit une erreur dans une unité d'exécution, il empêche les autres unités d'exécution d'essayer de transmettre leurs messages. Le Gestionnaire de documents replace ces messages dans la file d'attente jusqu'à ce qu'il soit en mesure de transmettre le message à l'origine de l'erreur.

Si le nombre d'échecs de tentatives excède le nombre maximal de tentatives autorisé, le Gestionnaire de documents place le message dans un répertoire d'échec puis tente de distribuer le message suivant dans la file d'attente sauf si la destination est hors ligne.

### Gestion des erreurs de transmission

Lorsque WebSphere Partner Gateway représente l'expéditeur et que l'application renvoie une erreur (par exemple, un message de réponse HTTP qui n'est ni un message 200 ni un 202 pour une utilisation du protocole HTTP), WebSphere Partner Gateway peut tenter une nouvelle fois d'envoyer le message en fonction de sa configuration pour cette destination spécifique. Chaque destination (un URL dans le cas d'un protocole HTTP) dispose des options suivantes qui affectent le nombre de relances autorisé et la méthode d'envoi des messages :

Tableau 20. Options de configuration de la destination

Options de configuration	Description
Nombre de relances	Définit le nombre de relances de document autorisé en cas de réception d'une erreur
Intervalle de relance	Définit l'intervalle entre chaque tentative de relance
En ligne/Hors ligne	Démarre et arrête les tentatives de livraison
Nombre d'unités d'exécution	Nombre d'unités d'exécution de postage traitant les messages par destination

Si WebSphere Partner Gateway n'est pas configuré pour effectuer une nouvelle tentative d'envoi du message ou que toutes les tentatives de livraison échouent, WebSphere Partner Gateway signale l'incident en procédant à l'une ou l'ensemble des opérations suivantes :

- en présentant les erreurs sur différentes vues de la Console de communauté tels que l'Afficheur de documents et l'Afficheur RosettaNet ;

- en envoyant un courrier électronique aux personnes concernées pour leur notifier l'incident de sorte qu'elles puissent prendre les mesures nécessaires, dans le cas où l'alerte par courrier électronique en cas d'échec de la livraison a été configurée ;
- en créant un document d'événements et en envoyant ce document au récepteur.

Pour plus d'informations, voir "Gestion des configurations de destination" dans le *Guide de l'administrateur WebSphere Partner Gateway*

## Messages en double

Tous les messages envoyés à ou reçu de WebSphere Partner Gateway doivent être accompagnés d'un identificateur global unique (GUID). WebSphere Partner Gateway utilise ce GUID pour détecter les messages en double. Lorsque l'empaquetage Backend Integration est utilisé, chaque message contient son GUID dans l'en-tête de niveau transport. Pour le protocole HTTP, par exemple, le GUID se trouve dans la zone `x-aux-system-msg-id` (voir «Contenu de l'en-tête de niveau transport», à la page 27). L'expéditeur du message crée le GUID. Le protocole de système de fichiers ne prend pas en charge la recherche de messages en double.

Si la tentative d'envoi d'un message mène à une erreur, WebSphere Partner Gateway utilise de nouveau le GUID du message à chaque nouvelle tentative. Si WebSphere Partner Gateway reçoit un message qui contient un GUID en double, il renvoie un accusé de réception positif (par exemple, HTTP 200) mais il ne traite pas le message en double.

**Remarque :** WebSphere Partner Gateway contrôle la présence de messages en double au niveau du processus RosettaNet lorsque c'est RosettaNet qui est utilisé. Il recherche également les messages en double si c'est XML qui est utilisé.

---

## Configuration de WebSphere Partner Gateway

«Processus de configuration du concentrateur WebSphere Partner Gateway», à la page 4 fournissait une description générale des étapes suivies par l'administrateur du concentrateur pour configurer le concentrateur. Cette section explique les étapes nécessaires à la configuration de WebSphere Partner Gateway destiné à être utilisé avec une application dorsale. Ces étapes de configuration impliquent que les partenaires externes ont été paramétrés dans la communauté du concentrateur. En particulier, cette section nécessite que les configurations suivantes ont été effectuées :

- Le profil partenaire du partenaire interne a déjà été créé.
- Les partenaires externes de la source (ou de la cible) des documents ont déjà été créés.
- Dans les programmes business-to-business des partenaires externes, les profils partenaire du partenaire interne ont été créés.
- Un récepteur a été défini pour que le récepteur de WebSphere Partner Gateway puisse écouter les documents entrants en provenance du partenaire externe via le protocole de transport approprié.
- Les fonctions business-to-business ont été définies et activées dans le profil du partenaire externe (d'où provient le document reçu) de sorte que WebSphere Partner Gateway s'attende à ce que les documents proviennent de cette source.
- Des connexions de partenaire existent entre le partenaire interne et les partenaires externes de sorte qu'un partenaire externe et le partenaire interne puissent recevoir (ou envoyer) un document.

**Remarque :** Vous devez vous connecter en tant qu'administrateur du concentrateur (hubadmin).

Pour plus d'informations sur la configuration de WebSphere Partner Gateway pour qu'il prenne en charge une communauté de concentrateur, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

Une fois les partenaires externes configurés, vous devez configurer WebSphere Partner Gateway pour qu'il puisse communiquer avec le système dorsal. Cette section fournit les informations suivantes pour décrire comment intégrer une application dorsale dans votre communauté :

- «Envoi de documents au système dorsal»
- «Réception de documents en provenance du système dorsal», à la page 52

## Envoi de documents au système dorsal

Pour envoyer un document au système dorsal, WebSphere Partner Gateway effectue les opérations suivantes :

1. Il reçoit un document provenant d'un partenaire externe.  
Le Récepteur extrait ce document source d'un récepteur défini au niveau du concentrateur pour les messages entrants en provenance d'un partenaire externe ainsi que le protocole de transport qui lui est associé. Lors de l'envoi d'un document au système dorsal, le document source correspond au document reçu d'un partenaire externe : il est par conséquent considéré comme le *document du partenaire*.
2. Il convertit le document du partenaire en document de destination au format requis par le système dorsal.  
Le Gestionnaire de documents WebSphere Partner Gateway est en charge de cette conversion en document cible. Lors de l'envoi d'un document au système dorsal, le document cible correspond au document envoyé au système dorsal ; il est par conséquent considéré comme le *document dorsal* .
3. Il envoie le document du dorsal au système dorsal.  
Le gestionnaire de documents envoie le document du dorsal au système dorsal via une destination définie au niveau du concentrateur pour les messages sortants.

Par conséquent, pour permettre au concentrateur d'envoyer un document au système dorsal, vous devez veiller à ce que la configuration résumée dans le tableau 21 ait été préalablement définie dans WebSphere Partner Gateway.

Tableau 21. Etapes de configuration pour l'envoi de documents au système dorsal

Étape de configuration	Étapes WebSphere Partner Gateway	Pour plus d'informations
1. Définir la destination du document.	1. Créer une destination vers le système dorsal.	«Définition de la cible d'envoi du document du partenaire», à la page 49

Tableau 21. Etapes de configuration pour l'envoi de documents au système dorsal (suite)

Etape de configuration	Etapes WebSphere Partner Gateway	Pour plus d'informations
2. Définir la méthode de traitement du document.	<p>2. Créer des définitions du flux de document pour les formats source et cible.</p> <p>3. Activer les fonctions business-to-business pour la définition du flux du document envoyée au système dorsal.</p> <p>4. Créer une interaction des définitions du flux de document entre les définitions source et cible du flux de document.</p>	«Définition du traitement du document de partenaire», à la page 50
3. Définir la méthode de connexion au système dorsal.	5. Créer une connexion de partenaire qui envoie les documents au système dorsal.	«Définition de la connexion au système dorsal», à la page 52

### Définition de la cible d'envoi du document du partenaire

Pour envoyer des documents au système dorsal, une destination doit être définie pour le concentrateur. Cette destination indique la destination des documents convertis. En d'autres termes, elle précise l'emplacement (sous forme d'URI) où le concentrateur envoie le document de dorsal. Cet emplacement est le même que celui utilisé par le système dorsal pour écouter les messages entrants. La destination identifie le point d'entrée dans la couche application d'entreprise (au sein du système dorsal). Dans WebSphere Partner Gateway, c'est le Gestionnaire de documents qui est chargé de détecter la destination. Après avoir traité le document, le Gestionnaire de documents envoie le document converti au système dorsal à l'emplacement indiqué dans la destination.

Pour définir une destination au sein de WebSphere Partner Gateway

1. Cliquez sur **Administrateur du compte > Profils**.
2. Cliquez sur Rechercher pour afficher la liste des partenaires.
3. Cliquez sur l'icône **Afficher les détails** en regard du partenaire interne.
4. Cliquez sur **Destinations**.
5. Cliquez sur **Créer**.

Lorsque vous définissez la destination, vous devez indiquer le protocole de transport utilisé par le concentrateur et le système dorsal pour transférer le document de dorsal. Comme le montre le tableau 16, à la page 38, le choix du protocole de transport dépend du format du document. Son format est fourni par son type d'empaquetage et son protocole de gestion, tous deux définis dans la définition du flux de document.

**Remarque :** Pour plus d'informations sur la création d'une destination dans WebSphere Partner Gateway, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

Le choix du protocole de transport dépend également des protocoles de transport pris en charge par votre propre application dorsale. Pour plus d'informations, voir le chapitre concernant l'intégration à votre propre système dorsal dans ce manuel.

Une fois qu'un protocole de transport valide a été sélectionné pour votre document, entrez les données requises pour la destination dans la vue Destinations.

### **Définition du traitement du document de partenaire**

Pour permettre au Gestionnaire de documents de traiter le document du partenaire, il doit connaître le format à utiliser pour convertir ce document. En d'autres termes, il doit connaître le format du document du dorsal. Dans le cadre de l'intégration dorsale, vous devez veiller à ce que les entités suivantes soient définies avec WebSphere Partner Gateway :

- Des définitions de document doivent exister pour définir le format du document du partenaire et du document du dorsal.
- Les fonctions business-to-business du partenaire interne doivent permettre l'intégration de la définition de flux de document du document de dorsal en tant que destination (cible).
- Une interaction de la définition du flux de document regroupant le document du partenaire comme source et le document du dorsal comme destination doit exister.

**Définition de la définition de document :** Chaque définition de document définit la manière dont WebSphere Partner Gateway traite un document particulier. Elle inclut le type d'emballage et le protocole de gestion du document. WebSphere Partner Gateway fournit quelques types d'emballage et définitions de protocoles prédéfinis. Si ces formats prédéfinis caractérisent correctement votre document de partenaire et votre document de dorsal, il n'est pas nécessaire de créer de définition de document. Toutefois, si les formats prédéfinis ne caractérisent pas convenablement votre document de partenaire ou votre document de dorsal, vous devez créer une définition de document appropriée pour ce document. Pour créer une définition de flux de document dans WebSphere Partner Gateway, vous devez utiliser la page Gestion des définitions du flux de document (**Administrateur du concentrateur > Configuration du concentrateur WebSphere Partner Gateway > Définition de flux de document > Création d'une définition de flux de document**).

**Remarque :** Pour plus d'informations sur les définitions du flux de documents prédéfinies ainsi que sur la création de définitions de flux de documents dans WebSphere Partner Gateway, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

Pour l'intégration dorsale, le type d'emballage du document de dorsal doit être l'un des suivants :

- Aucun emballage
- Emballage Backend Integration

Vous devez déterminer lequel de ces types d'emballage s'applique, en fonction du protocole de gestion de votre document et de l'application dorsale spécifique utilisée. Pour plus d'informations sur les types d'emballage associés aux applications dorsales, voir «Quel emballage allez-vous utiliser ?», à la page 26. Pour plus d'informations sur les applications dorsales prises en charge, voir «Comment accéder à l'application dorsale?», à la page 45.

**Configuration des fonctions business-to-business pour l'envoi :** Avant de pouvoir convertir le document source, le Gestionnaire de documents doit déterminer s'il peut gérer le format du document de destination souhaité. Pour ce faire, le gestionnaire de documents vérifie les fonctions business-to-business du profil du partenaire interne, qui définit quelles définitions de flux de documents ont été activées pour celui-ci. Chaque type de document des composants des définitions du flux de documents prises en charge (tel que le type d'emballage, le protocole de gestion et le type de document) est activé. Pour activer une définition du flux de document spécifique, vous devez utiliser la page Fonctions business-to-business de WebSphere Partner Gateway. Pour accéder à cette vue, suivez les étapes ci-dessous :

1. Cliquez sur **Administrateur du compte > Profils > Partenaire**.
2. Cliquez sur **Rechercher** pour afficher la liste des partenaires.
3. Cliquez sur l'icône **Afficher les détails** en regard du partenaire interne.
4. Cliquez sur **Fonctions business-to-business**.
5. Pour l'intégration dorsale, assurez-vous que chacun des types de document envoyés au dorsal est activé avec l'emballage, le protocole de gestion et le type de document appropriés pour le document défini comme cible (Set as target). Sous **Définir la cible**, activez chaque composant du type de document dans la définition de document du document de dorsal.
6. Dans la page Capacités business-to-business, si le concentrateur est également supposé recevoir des documents issus de l'application dorsale, vous pouvez activer les fonctions business-to-business correspondants. Dans ce cas, vous activez, sous **Définition de la source**, le composant de type du document dorsal.

**Remarque :** Pour plus d'informations sur le mode de configuration des fonctions business-to-business dans WebSphere Partner Gateway, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

**Définition de l'interaction de documents pour l'envoi :** Pour permettre au Gestionnaire de documents de déterminer comment convertir le document du partenaire, il doit être en mesure de localiser une interaction qui combine les définitions du flux de document pour les documents du partenaire et du dorsal et identifie le partenaire source et destination.

Lorsque le gestionnaire de documents est prêt à envoyer le document converti au système dorsal, il doit être en mesure de rechercher une connexion de partenaires entre le partenaire et le partenaire destination (système dorsal). Toutefois, pour qu'une connexion de partenaire existe, il faut qu'une interaction valide entre les documents source et destination existe. Pour définir une définition de flux de document dans WebSphere Partner Gateway, cliquez sur : **Administrateur du concentrateur > Configuration du concentrateur > Définition du flux de document > Gestion des interactions > Créer une interaction**.

**Remarque :** Pour plus d'informations sur le mode de création d'interactions de la définition du flux de document dans WebSphere Partner Gateway, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

Pour envoyer des documents à l'application dorsale, vous pouvez définir une interaction entre les documents source et de destination (cible), comme indiqué dans le tableau 22.

Tableau 22. Création d'une interaction pour l'envoi d'un document

Section Gestion des interactions	Action
Source	Sélectionnez les types de document des composants dans la définition du flux de document du <i>partenaire</i> .
Cible	Sélectionnez les types de document des composants dans la définition du flux de document du <i>dorsal</i> .

## Définition de la connexion au système dorsal

Pour permettre au gestionnaire de documents d'envoyer le document converti au système dorsal, il doit trouver une connexion de partenaire valide qui identifie les partenaires source et destination et indique l'emplacement utilisé par ces deux partenaires pour communiquer. Pour créer une connexion de partenaire, vous devez utiliser la page Gestion des connexions de WebSphere Partner Gateway. Pour accéder à cette vue, cliquez sur : **Administrateur du compte > Connexions**.

Pour qu'une connexion de partenaire soit définie, une interaction de la définition de flux de document entre les documents source et destination doit préexister. Sur la vue Gestion des connexions, vous devez d'abord vérifier l'existence d'une interaction en indiquant les partenaires source et destination (cible). Le tableau 23 répertorie les partenaires à sélectionner dans la page Gestion des connexions afin de définir une connexion de partenaire pour l'envoi d'un document vers le système dorsal.

Tableau 23. Création d'une connexion de partenaire pour l'envoi de document

Liste déroulante Gestion des connexions	Nom du partenaire externe
Source	Nom du partenaire externe, qui envoie le document au partenaire interne
Cible	Nom du partenaire interne, qui reçoit le document du partenaire externe

Une fois la source et la cible définies, vous pouvez cliquer sur **Rechercher** pour rechercher une interaction existante de la définition du flux de document. Si aucune interaction n'existe, vous devez en créer une avant de pouvoir poursuivre la création de la connexion de partenaire. Si une interaction existe (une dont la source est la définition de flux de document du partenaire et dont la cible est la définition de flux de document du dorsal), vous pouvez configurer la connexion du partenaire pour activer la communication avec le système dorsal.

**Remarque :** Pour plus d'informations sur la façon de créer une connexion de partenaire dans WebSphere Partner Gateway, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

Pour l'intégration dorsale, cette connexion de partenaire doit spécifier comme destination cible la destination définie dans «Définition de la cible d'envoi du document du partenaire», à la page 49.

## Réception de documents en provenance du système dorsal

Pour recevoir un document du système dorsal, le concentrateur effectue les opérations suivantes :

1. Il reçoit un document provenant du système dorsal.

Le Récepteur WebSphere Partner Gateway extrait ce document source d'un récepteur défini par le concentrateur pour les messages entrants en provenance du système dorsal ainsi que le protocole de transport qui lui est associé. Lors de la réception d'un document en provenance du système dorsal, le document source correspond au document reçu du système dorsal ; par conséquent, ce document est appelé *document de dorsal*

2. Il convertit le document de dorsal en document de destination (cible) au format requis par le partenaire externe désigné.

Le Gestionnaire de documents est en charge de cette conversion en document cible. Lors de la réception d'un document du système dorsal, le document cible correspond au document envoyé à un partenaire externe : il est par conséquent appelé *document de partenaire*

3. Il envoie le document de partenaire au partenaire externe approprié.

Le Gestionnaire de documents envoie le document du partenaire au partenaire externe approprié par l'intermédiaire d'une destination définie par le concentrateur pour les messages sortants.

Par conséquent, pour permettre au concentrateur de recevoir un document de l'application dorsale, vous devez veiller à ce que la configuration résumée dans le tableau 24 ait été préalablement définie dans WebSphere Partner Gateway.

Tableau 24. Etapes de configuration pour la réception de documents en provenance du système dorsal

Etape de configuration	Etapes WebSphere Partner Gateway	Pour plus d'informations
1. Définir le lieu d'extraction du document.	1. Créer un récepteur pour la réception des messages entrants en provenance du système dorsal.	«Définition de l'emplacement à partir duquel le document dorsal sera récupéré»
2. Définir la méthode de traitement du document.	2. Créer des définitions du flux de document pour les formats source et cible.  3. Activer les fonctions business-to-business pour la définition du flux du document reçus du système dorsal.  4. Créer une interaction des définitions du flux de document entre les définitions source et cible du flux de document.	«Définition du traitement du document dorsal», à la page 54
3. Définir la méthode de connexion à WebSphere Partner Gateway.	5. Créer une connexion de partenaire qui envoie les documents à WebSphere Partner Gateway.	«Définition de la connexion à WebSphere Partner Gateway», à la page 55

### Définition de l'emplacement à partir duquel le document dorsal sera récupéré

Pour recevoir des documents en provenance du système dorsal, un récepteur doit être définie pour le concentrateur. Ce récepteur indique la source des documents. En d'autres termes, elle précise l'emplacement (sous forme d'URI) où le concentrateur écoute les documents entrants. Cet endroit est le même que celui utilisé par le système dorsal pour envoyer les documents. Le récepteur identifie le point d'entrée dans le Récepteur (au sein de WebSphere Partner Gateway). Au sein de WebSphere Partner Gateway, c'est le Récepteur qui est chargé de recevoir les documents. Après avoir traité le document, le Récepteur enregistre le document converti dans la mémoire partagée permanente pour permettre au Gestionnaire de documents de l'extraire ultérieurement.

Pour définir un récepteur dans WebSphere Partner Gateway, cliquez sur **Administrateur du concentrateur > Configuration du concentrateur> Récepteurs**.

**Remarque :** Pour plus d'informations sur la création d'un récepteur dans WebSphere Partner Gateway, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

Lorsque vous définissez le récepteur, vous devez indiquer le protocole de transport utilisé par le concentrateur et le système dorsal pour transférer le document de dorsal. Comme le montre le tableau 17, à la page 38, le choix du protocole de transport dépend du format du document. Son format est fourni par son type d'emballage et son protocole de gestion, tous deux définis dans la définition du flux de document.

**Remarque :** Le choix du protocole de transport dépend également des protocoles de transport pris en charge par votre propre application dorsale. Pour plus d'informations, voir le chapitre concernant l'intégration à votre propre application dorsale dans ce manuel.

Une fois qu'un protocole de transport valide a été sélectionné pour votre document, entrez les renseignements nécessaires à la définition du récepteur dans la vue Caractéristiques récepteur.

### Définition du traitement du document dorsal

Pour permettre au Gestionnaire de documents de traiter le document de dorsal, il doit connaître le format dans lequel convertir ce document. En d'autres termes, il doit connaître le format du document du partenaire. Dans le cadre de l'intégration dorsale, vous devez veiller à ce que les entités répertoriées dans le tableau 25 soient définies avec WebSphere Partner Gateway.

Tableau 25. Définition de la méthode de conversion du document dorsal

Étape	Pour plus d'informations
1. Des définitions de document doivent exister pour définir à la fois le format du document du partenaire et du dorsal.	«Définition de la définition de document», à la page 50
2. Les fonctions business-to-business du partenaire interne doivent permettre l'intégration de la définition de flux de document de dorsal en tant que source.	«Configuration des fonctions business-to-business pour la réception»
3. Une interaction de la définition de document réunissant le document de dorsal en source et le document du partenaire comme destination doit exister.	«Définition d'une interaction pour la réception», à la page 55

**Configuration des fonctions business-to-business pour la réception :** Pour obtenir un récapitulatif des fonctions business-to-business et de leur application à l'intégration dorsale, voir «Configuration des fonctions business-to-business pour l'envoi», à la page 51.

Si ce n'est pas déjà fait, activez, sous **Définition de la source**, le composant de type du document dorsal.

**Définition d'une interaction pour la réception :** Pour obtenir un récapitulatif des interactions des définitions du flux de document et de leur application à l'intégration dorsale, voir «Définition de l'interaction de documents pour l'envoi», à la page 51. Cette section montre comment définir l'interaction pour la réception d'un document de l'application dorsale.

Pour recevoir des documents en provenance du système dorsale, définir une interaction entre les définitions de flux de document pour les documents source et de destination (cible), comme le récapitule le tableau 26.

Tableau 26. Création d'une interaction pour la réception d'un document

Section Gestion des interactions	Action
Source	Sélectionnez les types de document des composants dans la définition du flux de document du <i>dorsal</i> .
Cible	Sélectionnez les types de document des composants dans la définition du flux de document du <i>partenaire</i> .

### Définition de la connexion à WebSphere Partner Gateway

Pour permettre au gestionnaire de documents de traiter le document en provenance du système dorsale, il doit trouver une connexion de partenaire valide qui identifie les partenaires source et destination et indique l'emplacement utilisé par ces deux partenaires pour communiquer. Pour une synthèse des connexions et de leur application à l'intégration dorsale, voir «Définition de la connexion au système dorsale», à la page 52.

Le tableau 27 répertorie les partenaires à sélectionner sur la page Gestion des connexions de WebSphere Partner Gateway afin de définir une connexion de partenaire pour la réception d'un document en provenance du système dorsale.

Tableau 27. Création d'une connexion de partenaire pour la réception d'un document

Liste déroulante Gestion des connexions	Nom du partenaire externe
Source	Nom du partenaire interne
Cible	Nom du partenaire externe qui reçoit le document du partenaire interne

Une fois la source et la cible définies, vous pouvez cliquer sur **Rechercher** pour rechercher une interaction existante de la définition du flux de document. Si aucune interaction existe, vous devez en créer une *avant* de pouvoir poursuivre la création d'une connexion de partenaire. Si une interaction existe (une dont la source est la définition de flux de document du dorsale et dont la cible est la définition de flux de document du partenaire), vous pouvez configurer la connexion du partenaire pour activer la communication avec le système dorsale.

**Remarque :** Pour plus d'informations sur la façon de créer une connexion de partenaire dans WebSphere Partner Gateway, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.



---

## **Partie 2. Intégration à WebSphere Process Server**

Les chapitres suivants décrivent la procédure permettant d'intégrer WebSphere Partner Gateway à WebSphere Process Server.



---

## Chapitre 3. Présentation de l'intégration à WebSphere Process Server

Le présent chapitre présente l'intégration entre WebSphere Partner Gateway et WebSphere Process Server.

**Remarque :** Pour obtenir une description du processus général utilisé pour intégrer WebSphere Partner Gateway à un système dorsal, voir Chapitre 2, «Planification de l'intégration dorsale», à la page 9.

Ce chapitre comprend les sections suivantes :

- «Présentation»
- «Planification de l'intégration à WebSphere Process Server», à la page 61
- «Présentation des tâches d'intégration de WebSphere Partner Gateway à WebSphere Process Server», à la page 64
- «Gestion des messages avec empaquetage Backend Integration», à la page 66
  - «Classe BCGBackEndIntegrationDataBindingUtil», à la page 78
  - «Classe BCGBackEndIntegrationJMSDataBindingImpl», à la page 84

---

### Présentation

Cette section explique comment WebSphere Partner Gateway peut permettre de rendre des fonctions B2B (business-to-business) disponibles sur WebSphere Process Server.

**Remarque :** Pour obtenir des informations détaillées sur WebSphere Process Server, consultez le centre de documentation de ce produit.

WebSphere Process Server utilise les fonctions business-to-business de WebSphere Partner Gateway pour gérer les interactions avec les partenaires d'échanges (appelés *partenaires externes* dans WebSphere Partner Gateway). Imaginons par exemple qu'un service exécuté sur WebSphere Process Server doit envoyer un document à un partenaire externe. WebSphere Process Server envoie alors le document à WebSphere Partner Gateway, qui détermine (le cas échéant) la mappe de transformation à utiliser pour transformer le document au format attendu par le partenaire externe. WebSphere Partner Gateway gère également les informations de profil de tous les partenaires externes.

Supposons maintenant que WebSphere Process Server doit envoyer un document à plusieurs partenaires externes, et que ces partenaires externes reçoivent ledit document dans des formats différents. WebSphere Process Server n'a à gérer qu'un type de format (celui qu'il envoie à WebSphere Partner Gateway). C'est WebSphere Partner Gateway qui gère l'interaction avec les partenaires externes.

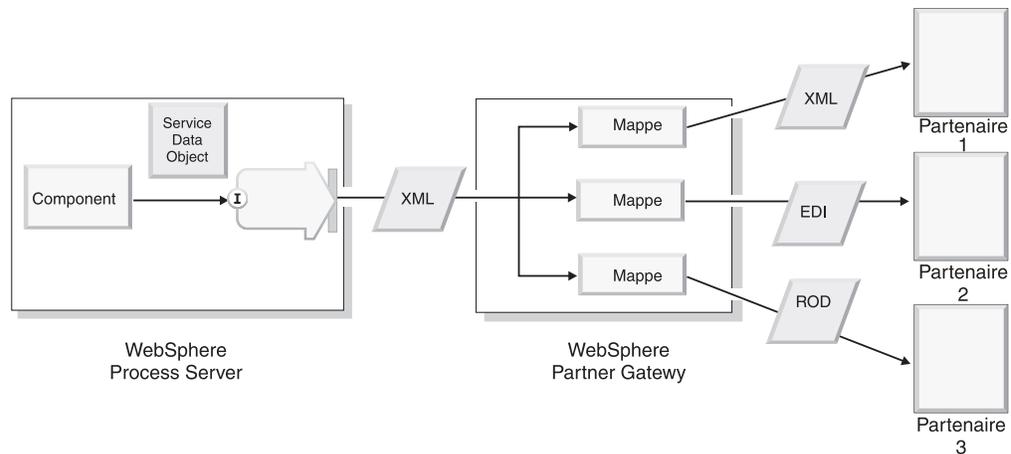


Figure 15. WebSphere Partner Gateway envoie les documents aux partenaires externes dans plusieurs formats

Vous pouvez développer vous-même les mappes de transformation qui convertissent le document au format requis par le partenaire externe, ou vous pouvez en importer depuis le programme client Data Interchange Services.

De même, lorsque WebSphere Process Server reçoit des documents depuis des partenaires externes, ces documents sont traités par WebSphere Partner Gateway. Ils peuvent se présenter dans divers formats. WebSphere Partner Gateway les transforme et les envoie vers la destination définie pour le partenaire interne sur WebSphere Process Server.

## Fonctionnement de la communication entre WebSphere Process Server et WebSphere Partner Gateway

WebSphere Partner Gateway envoie un document depuis un partenaire externe vers WebSphere Process Server, en vue de son traitement par un service résidant sur WebSphere Process Server. Pour pouvoir être utilisé par d'autres applications, un service résidant sur WebSphere Process Server (dont WebSphere Partner Gateway) se sert de sa liaison d'exportation.

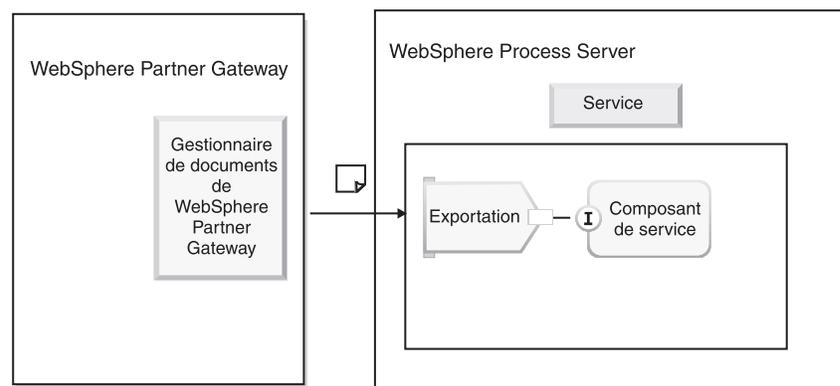


Figure 16. Utilisation de la liaison d'exportation par WebSphere Partner Gateway pour l'envoi de documents à WebSphere Process Server

Le composant de WebSphere Process Server possède une interface qui décrit le service (méthodes disponibles et données d'entrée et de sortie), ainsi qu'une liaison (en l'occurrence, une liaison d'exportation).

La figure 16, à la page 60 présente une vue générique de l'appel d'un service par la liaison d'exportation sur WebSphere Process Server. Le type de transport (par exemple, JMS ou HTTP) que vous utilisez pour envoyer le message a une incidence sur le lieu de destination du message et sur les modalités de sa récupération, comme l'indique la section «Transports de messages pris en charge par WebSphere Process Server», à la page 62.

Symétriquement, lorsqu'un service résidant sur WebSphere Process Server doit envoyer un document de gestion à un partenaire externe, il utilise sa liaison d'importation. Les liaisons d'importation identifient les services externes à un module, afin qu'ils puissent être appelés depuis le module. En l'occurrence, WebSphere Process Server utilise la liaison d'importation pour appeler WebSphere Partner Gateway, qui traite le document et l'envoie au partenaire externe.

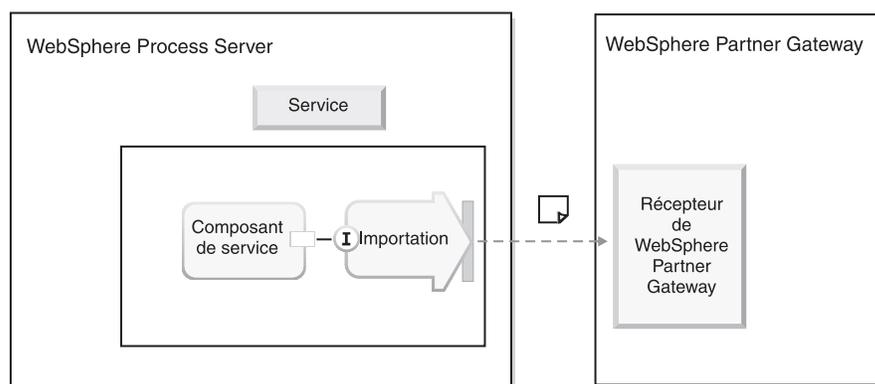


Figure 17. Utilisation de la liaison d'importation par WebSphere Process Server pour appeler WebSphere Partner Gateway

---

## Planification de l'intégration à WebSphere Process Server

Pour planifier l'intégration à WebSphere Process Server, suivez les étapes décrites à la section Chapitre 2, «Planification de l'intégration dorsale», à la page 9.

### Versions de WebSphere Process Server prises en charge par WebSphere Partner Gateway

La version 6.1 de WebSphere Partner Gateway prend en charge l'intégration à WebSphere Process Server Version 6.0.2.

WebSphere Process Server est disponible sur de nombreuses plateformes, y compris Windows 2000 et plusieurs plateformes UNIX. Pour plus d'informations, consultez le guide d'installation de WebSphere Process Server dans le centre de documentation relatif à ce produit.

## Scénarios d'installation pris en charge

Dans le tableau suivant, chaque rangée correspond à une combinaison d'installations prise en charge de WebSphere Partner Gateway et de WebSphere Process Server.

**Remarque :** Au cours de l'installation, WebSphere Partner Gateway crée des profils WebSphere Application Server séparés pour chacun de ses composants. Ces profils sont utilisés uniquement par WebSphere Partner Gateway. Ne déployez pas WebSphere Process Server ou d'autres applications WebSphere Application Server dans ces profils.

Tableau 28. Scénarios d'installation pris en charge

Installation de WebSphere Partner Gateway	Installation de WebSphere Process Server	Caractéristiques
Installé sur WebSphere Application Server 6.1, ou sur une ou plusieurs instances de WebSphere Application Server Network Deployment 6.1.	L'utilitaire d'installation de WebSphere Process Server installe sa version prise en charge de WebSphere Application Server ND parallèlement à WebSphere Process Server.	
Installé sur WebSphere Application Server 6.1, ou sur une ou plusieurs instances de WebSphere Application Server Network Deployment 6.1.	Installé sur l'installation de WebSphere Application Server 6.1 qui contient également WebSphere Partner Gateway, mais pas sur la même instance. Le profil utilisé pour WebSphere Process Server doit être différent du profil utilisé par les composants de WebSphere Partner Gateway.	Combinaison d'installation compatible uniquement avec les plateformes (systèmes d'exploitation et versions) prises en charge à la fois par WebSphere Partner Gateway et par WebSphere Process Server.

## Transports de messages pris en charge par WebSphere Process Server

Lorsque WebSphere Partner Gateway envoie un message à WebSphere Process Server via un protocole de transport de messages particulier, la destination propre à ce transport définie pour le partenaire interne transmet ce message au noeud final de WebSphere Process Server. WebSphere Process Server récupère le message depuis le noeud final, puis le traite. Le type de transport de message détermine le mode de gestion du message après son arrivée sur WebSphere Process Server :

- Dans le cas d'un transport JMS, vous devez définir une exportation JMS pour récupérer le message depuis une file d'attente JMS.
- Dans le cas d'un transport HTTP, vous devez créer un servlet sur WebSphere Process Server qui gèrera la réception du message depuis WebSphere Partner Gateway.
- Dans le cas de documents SOAP (envoyés via le protocole de transport HTTP), vous devez définir une liaison d'exportation de service Web sur WebSphere Process Server pour récupérer la requête SOAP.
- Dans le cas d'un transport via le système de fichiers, vous devez définir un adaptateur de fichier à plat entrant pour router le message vers WebSphere Process Server.

Lorsque vous envoyez des messages depuis WebSphere Process Server vers WebSphere Partner Gateway, vous devez les transmettre au récepteur propre au transport (par exemple, une file d'attente JMS ou une URL) résidant sur WebSphere Partner Gateway. Le type de transport de message détermine le mode d'envoi du message.

- Dans le cas d'un transport JMS, vous devez définir une importation JMS pour envoyer le message à une file d'attente JMS.
- Dans le cas d'un transport HTTP, vous devez créer, sur WebSphere Process Server, un composant qui effectuera une requête HTTP POST sur l'URL spécifiée du récepteur de WebSphere Partner Gateway.
- Dans le cas de documents SOAP (envoyés via le protocole de transport HTTP), vous devez définir une liaison d'importation de service Web sur WebSphere Process Server afin d'envoyer la requête SOAP à une URL de WebSphere Partner Gateway.
- Dans le cas d'un transport via le système de fichiers, vous devez définir un adaptateur de fichier à plat sortant pour router le message vers un répertoire situé sur WebSphere Partner Gateway.

Les sections suivantes contiennent des informations sur l'échange de messages à l'aide des transports pris en charge :

- Chapitre 4, «Intégration de WebSphere Process Server avec HTTP comme protocole de transport», à la page 87
- Chapitre 5, «Intégration de WebSphere Process Server avec le protocole de transport JMS», à la page 95
- Chapitre 6, «Intégration de WebSphere Process Server à SOAP/HTTP», à la page 107
- Chapitre 7, «Intégration de WebSphere Process Server avec le système de fichiers comme protocole de transport», à la page 113

## Prise en charge de l'intégration à WebSphere Process Server

Cette section décrit les exemples, la documentation et les classes utilitaires offerts par WebSphere Partner Gateway afin de faciliter l'intégration à WebSphere Process Server.

### Exemples

WebSphere Partner Gateway propose des exemples d'utilisation du protocole de transport JMS afin de faciliter le processus d'intégration à WebSphere Process Server. Ces exemples se trouvent dans le sous-répertoire suivant du répertoire de produit de WebSphere Partner Gateway :

Integration/WBI/WPG/samples

Le tableau 29 énumère les sous-répertoires du répertoire samples.

Tableau 29. Exemples pour l'intégration à WebSphere Process Server

Type d'exemple	Sous-répertoire d'exemples
Exemples généraux	JMS
Exemples RosettaNet	RosettaNet/JMS

## Documentation

Outre les informations contenues dans ce document, WebSphere Partner Gateway propose le document *PIP Sample for WebSphere Process Server*, qui donne des instructions progressives sur la configuration d'un flux PIP entre un partenaire externe et WebSphere Process Server.

## Classes utilitaires

WebSphere Partner Gateway propose deux classes utilitaires qui peuvent permettre de transformer un message avec empaquetage Backend Integration en objet métier ou un objet métier en message avec empaquetage Backend Integration :

- Classe `BCGBackEndIntegrationDataBindingUtil`  
Cette classe implémente l'Interface `DataBinding` et offre deux méthodes utilitaires pour lire et écrire des chaînes, des flux et des tableaux d'octets. Vous pouvez utiliser cette classe en l'état ou développer une nouvelle liaison de données par son intermédiaire.
- Classe `BCGBackEndIntegrationJMSDataBindingImpl`  
Cette classe implémente l'Interface `JMSDataBinding`. Elle peut être spécifiée dans les liaisons d'exportation et d'importation SCA JMS. Elle crée un objet données à partir d'un message JMS contenant des données utiles ou écrit l'objet données dans un message JMS.

---

## Présentation des tâches d'intégration de WebSphere Partner Gateway à WebSphere Process Server

Cette section répertorie les tâches à effectuer pour que WebSphere Partner Gateway puisse échanger des documents avec WebSphere Process Server. Elle offre une base aux chapitres consacrés au transport qui décrivent l'intégration à WebSphere Process Server.

### Sur le système WebSphere Partner Gateway

Cette section présente très brièvement les tâches à accomplir pour configurer le concentrateur de manière à ce que vous puissiez échanger des documents avec WebSphere Partner Gateway. Ces tâches, décrites en détail dans le *Guide de configuration du concentrateur WebSphere Partner Gateway*, sont effectuées sur la console de communauté de WebSphere Partner Gateway.

- Créez un récepteur propre au transport sur le concentrateur afin qu'il reçoive les documents envoyés à partir de WebSphere Process Server ou de partenaires externes.
- Créez un profil de partenaire interne (s'il n'en existe aucun), incluant une destination spécifique au transport que WebSphere Partner Gateway utilisera pour envoyer des documents à WebSphere Process Server.
- Créez des profils de partenaire externe, incluant des destinations spécifiques au protocole de transport que WebSphere Partner Gateway utilisera pour envoyer des documents aux partenaires.
- Importez tous les fichiers WSDL, mappes de transformation, packages RosettaNet ou autres mécanismes de définition de document, afin d'afficher, sur la page Définition de flux de document de la console de communauté de WebSphere Partner Gateway, une définition de document pour le type de document que vous échangez.
- Créez des interactions entre les types de document que le concentrateur recevra (depuis WebSphere Process Server ou un partenaire externe) et ceux qu'il enverra (à WebSphere Process Server ou aux partenaires externes).

- Créez des fonctions business-to-business dans les profils du partenaire interne et des partenaires externes pour indiquer les types de documents qu'ils peuvent envoyer ou recevoir.
- Créez des connexions de partenaire entre le partenaire interne et les partenaires externes pour indiquer le partenaire source (expéditeur du document), le partenaire cible (son destinataire) et l'action que le concentrateur doit accomplir (le cas échéant) pour transformer le document.

## Sur le système WebSphere Process Server

Un module est un artefact de WebSphere Process Server permettant d'assembler et de déployer un service. La première étape consiste donc à créer un module à l'aide de WebSphere Integration Developer.

Après la création du module, vous devez créer ses composants et leurs interfaces, puis spécifier la liaison utilisée.

1. Spécifiez une interface pour le composant. Vous pouvez importer une interface (par exemple, un fichier WSDL existant) ou en créer une.  
Lorsque vous créez une interface, vous devez définir une ou plusieurs opérations effectuées par le composant, ainsi que les entrées et sorties attendues par le composant.  
L'interface d'un composant peut consister en un fichier WSDL ou en une implémentation Java. Reportez-vous à la documentation de WebSphere Process Server pour déterminer si vous devez créer une interface WSDL ou si vous devez créer une interface Java .
2. Spécifiez une implémentation pour le composant. Vous pouvez importer une implémentation (par exemple, un programme Java existant) ou en créer une.
3. Composez l'application à l'aide de l'éditeur d'assemblage de WebSphere Integration Developer. Créez un composant de service, puis spécifiez l'interface du composant (que vous avez créée ou importée à l'étape 1). Spécifiez également l'implémentation (que vous avez créée ou importée à l'étape 2).
4. Créez une liaison d'exportation pour permettre à WebSphere Partner Gateway d'envoyer un document au service, ou une liaison d'importation pour permettre à WebSphere Process Server d'envoyer un document à WebSphere Partner Gateway.  
Lors de la création de la liaison, indiquez les informations nécessaires à l'envoi et à la réception de documents. Par exemple, dans la définition de la liaison JMS, indiquez la file d'attente et le bus JMS, ainsi que la liaison de données à utiliser pour transformer un document de gestion en objet métier ou vice-versa. (Voir «Gestion des messages avec empaquetage Backend Integration», à la page 66 pour plus d'informations sur l'utilisation et les exigences de la liaison de données.) Les exigences des liaisons de service propres à chaque transport sont décrites dans les chapitres suivants.
5. Au terme de l'assemblage des composants formant le module, déployez le module.

---

## Gestion des messages avec empaquetage Backend Integration

Pour certains protocoles, tels que RosettaNet, WebSphere Partner Gateway s'attend à ce que les applications dorsales utilisent l'empaquetage Backend Integration. WebSphere Partner Gateway prend en charge cet empaquetage via les protocoles HTTP et JMS. Les services de WebSphere Process Server doivent donc pouvoir gérer les messages avec empaquetage Backend Integration. Ces services utilisent des objets métier. Par conséquent, pour envoyer des messages avec empaquetage Backend Integration vers WebSphere Partner Gateway, les services de WebSphere Process Server doivent sérialiser les objets métier sous forme de messages avec empaquetage Backend Integration. Symétriquement, pour recevoir les messages avec empaquetage Backend Integration provenant de WebSphere Partner Gateway, les services de WebSphere Process Server doivent les désérialiser sous forme d'objets métier.

Cette section décrit l'API utilitaire de liaison de données et la liaison de données JMS proposées par WebSphere Partner Gateway qui permettent aux services WebSphere Process Server de gérer les messages avec empaquetage Backend Integration.

La liaison de données correspond au mécanisme utilisé pour :

- transformer un objet métier reçu depuis WebSphere Process Server en un document de gestion que WebSphere Partner Gateway peut traiter ;
- transformer un document de gestion reçu depuis WebSphere Partner Gateway en un objet métier que WebSphere Process Server peut traiter.

L'API utilitaire de liaison de données pour l'empaquetage Backend Integration et la liaison de données JMS proposées par WebSphere Partner Gateway permettent de créer un message avec empaquetage Backend Integration à partir d'un objet métier ou de convertir un message avec empaquetage Backend Integration en objet métier. La liaison de données fournie par WebSphere Partner Gateway peut être utilisée en l'état lorsque vous envoyez ou recevez des documents avec empaquetage Backend Integration. En revanche, si vous souhaitez personnaliser le mode de traitement, vous pouvez créer votre propre liaison de données à l'aide de l'API utilitaire de liaison de données proposée par WebSphere Partner Gateway.

## Transport JMS

Si vous utilisez le protocole de transport JMS et que vous envoyez ou recevez un document XML sans emballage BackEnd Integration (l'emballage spécifié étant "None"), vous pouvez utiliser la liaison JMS par défaut proposée par WebSphere Process Server ou créer votre propre liaison de données à l'aide des API d'objet métier offertes par WebSphere Process Server. Si vous envoyez ou recevez un document non XML sans emballage BackEnd Integration (l'emballage spécifié étant "None"), vous devez créer votre propre liaison de données. Sélectionnez la liaison JMS lors de la configuration de la liaison d'importation SCA (Service Component Architecture) JMS. Consultez le centre de documentation de WebSphere Process Server pour plus d'informations sur l'utilisation de la liaison JMS par défaut.

Si vous utilisez le protocole de transport JMS et que vous envoyez ou recevez un document avec emballage BackEnd Integration, vous pouvez utiliser la liaison de données `BCGBackendIntegrationJMSDataBindingImpl` en l'état, ou, si vous souhaitez personnaliser le mode de traitement, vous pouvez créer votre propre liaison de données à l'aide de `BCGBackendIntegrationJMSDataBindingImpl`. Pour utiliser la liaison `BCGBackendIntegrationJMSDataBindingImpl`, configurez-la comme liaison de données dans les liaisons d'exportation et d'importation SCA JMS.

- Pour le traitement des demandes, l'exportation appelle la méthode `read` de l'interface `JMSDataBinding` afin de convertir en objet métier le message JMS avec emballage Backend Integration provenant de WebSphere Partner Gateway. Pour le traitement des flux demande-réponse, l'exportation peut également appeler la méthode `write`.
- Pour le traitement des demandes, l'importation appelle la méthode `write` de l'interface `JMSDataBinding` afin de construire un message JMS avec emballage Backend Integration à partir d'un objet métier, avant de l'envoyer à WebSphere Partner Gateway. Pour le traitement des flux demande-réponse, l'importation peut également appeler la méthode `read` pour lire la réponse du service.

Consultez le centre de documentation de WebSphere Process Server pour plus d'informations sur l'appel des méthodes de liaison de données.

## Transport HTTP

Si vous utilisez le protocole de transport HTTP, vous pouvez écrire une classe de liaison de données HTTP qui étend et remplace la Classe `BCGBackendIntegrationDataBindingUtil` générique.

- Un servlet HTTP peut appeler des méthodes de `BCGBackendIntegrationDataBindingUtil` pour construire un objet métier à partir du flux HTTP provenant de WebSphere Partner Gateway. Le servlet peut alors utiliser cet objet métier pour appeler un service.
- Vous pouvez écrire un composant pour effectuer une requête HTTP POST sur WebSphere Partner Gateway. Un service peut alors appeler ce composant avec l'objet métier. Si le composant nécessite un message avec emballage Backend Integration, il peut utiliser la classe `BCGBackendIntegrationDataBindingUtil` pour construire le message à partir de l'objet métier.

**Remarque :** WebSphere Partner Gateway prend en charge les appels de type demande seule pour l’empaquetage Backend Integration. Cependant, vous pouvez développer des flux demande-réponse avec la liaison de données de l’empaquetage Backend Integration (par le biais des exits utilisateur de WebSphere Partner Gateway). La classe utilitaire de cette liaison peut servir de base à la conversion d’un objet métier de type réponse en flux de réponse ou d’un flux de réponse en objet métier de type réponse.

## Objets métier enfant et de niveau supérieur

Cette section décrit l’objet métier de niveau supérieur et les objets métier enfant utilisés par la liaison de données Backend Integration.

### Objet métier de niveau supérieur

Pour utiliser la liaison de l’empaquetage Backend Integration, créez un objet métier de niveau supérieur possédant trois attributs enfant :

*Tableau 30. objet métier de niveau supérieur*

Attribut	Type
payload	Cet attribut est de type objet métier conteneur de données utiles. Il peut être de tout type, mais ses propriétés doivent correspondre à celles qui sont indiquées dans le tableau 31, à la page 69 ou le tableau 32, à la page 70
attachment	Cet attribut est de type objet métier conteneur de pièces jointes. Il peut être de tout type, mais ses propriétés doivent correspondre à celles qui sont indiquées dans le tableau 33, à la page 71
packagingHeaders	Cet attribut est de type objet métier en-têtes d’empaquetage. Il peut être de tout type, mais ses propriétés doivent correspondre à celles qui sont indiquées dans le tableau 34, à la page 71

Lorsqu’un message avec empaquetage Backend Integration est converti en objet de niveau supérieur, ce dernier est alimenté par les données du message. Lorsqu’un objet de niveau supérieur est converti en message avec empaquetage Backend Integration, ce dernier constitue l’entrée de la liaison de données.

Les figures suivantes illustrent l'objet métier de niveau supérieur et ses objets enfant :

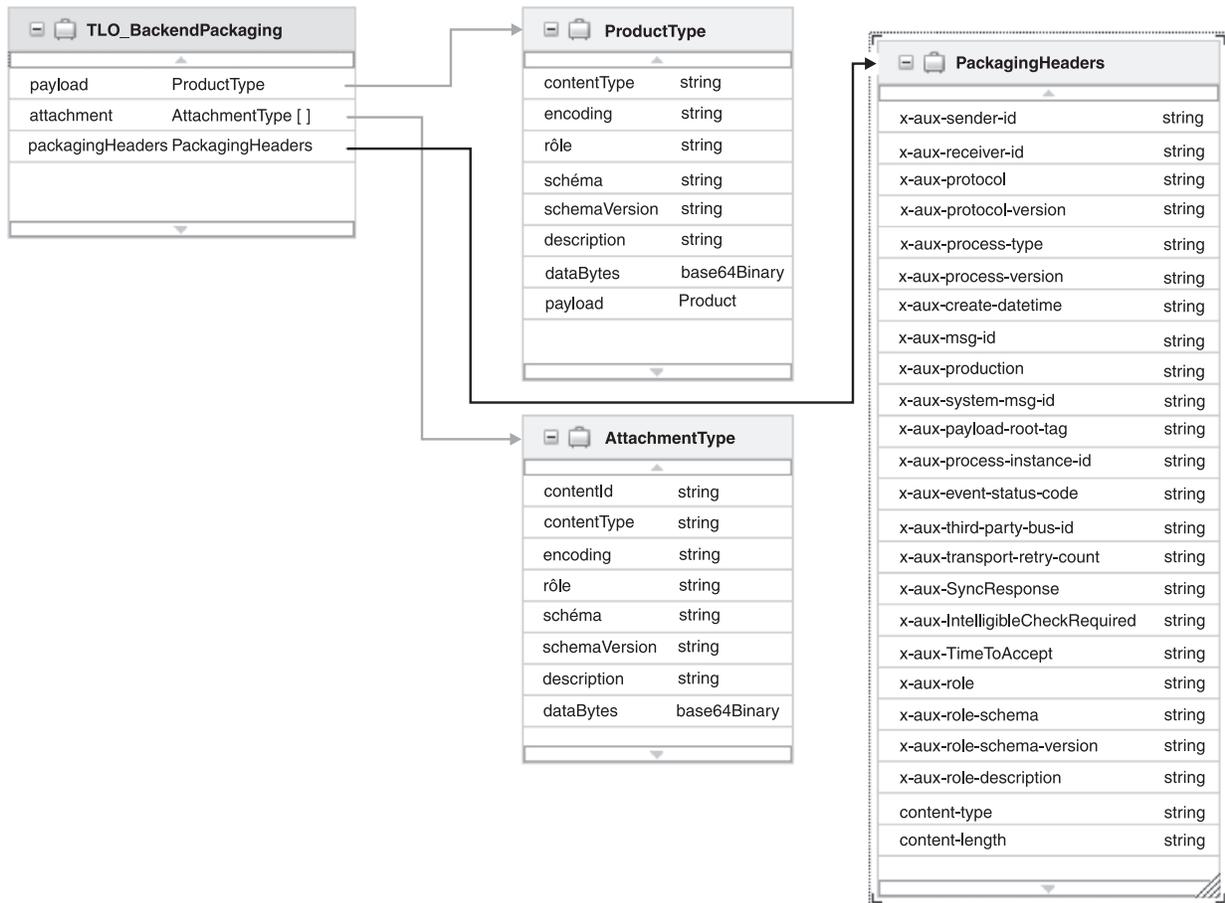


Figure 18. objet métier de niveau supérieur

### Objet métier conteneur de données utiles

La structure de l'objet métier conteneur de données utiles dépend du format des données utiles : XML ou non XML.

**Données utiles XML :** Si les données utiles sont au format XML, le conteneur possède les propriétés suivantes :

Tableau 31. Objet conteneur de données utiles pour les données utiles XML

Attribut	Type
contentType	xsd:string
encoding	xsd:string
rôle	xsd:string
schéma	xsd:string
versionSchéma	xsd:string
description	xsd:string
dataBytes	xsd:base64Binary

Tableau 31. Objet conteneur de données utiles pour les données utiles XML (suite)

Attribut	Type
payload	Référence à l'objet métier données utiles. Par exemple, si les données utiles sont contenues dans un objet appelé Pip3A4PurchaseOrderRequest, l'attribut payload est de type PIP3A4PurchaseOrderRequest.

La section «Création d'objets métier pour le contenu de service RosettaNet», à la page 72 décrit la création d'un objet métier pour un PIP.

**Données utiles non XML :** Si les données utiles ne sont pas au format XML, le conteneur possède les propriétés suivantes :

Tableau 32. Objet conteneur de données utiles pour les données utiles non XML

Attribut	Type
contentType	xsd:string
encoding	xsd:string
rôle	xsd:string
schéma	xsd:string
versionSchéma	xsd:string
description	xsd:string
dataBytes	xsd:base64Binary
dataString	xsd:string

Les données utiles se retrouvent dans la valeur de l'attribut dataBytes ou dataString.

- Si vos données utiles ne sont pas au format XML, lors de la conversion du message avec empaquetage Backend Integration en objet de niveau supérieur, la liaison de données de l'empaquetage ne convertira pas la valeur de l'élément <payload> du message en objet métier données utiles. En revanche, l'attribut dataBytes (ou dataString) de l'objet métier conteneur de données utiles aura pour valeur les octets (ou la chaîne) des données utiles.
- Symétriquement, lors de la construction du message avec empaquetage Backend Integration à partir d'un objet métier de niveau supérieur, la liaison de données de l'empaquetage définira le contenu de l'attribut dataBytes (ou dataString) de l'objet métier conteneur de données utiles comme valeur de l'élément <payload> du message avec empaquetage Backend Integration.

Les attributs Role, Schema, SchemaVersion et Description sont ajoutés dans wbipackaging 1.2. Ils ne sont utilisés que dans les flux ebXML. La valeur de ces attributs pour les données utiles sera définie en tant que valeur des attributs correspondants dans le message ebXML Soap pour ces données utiles.

### Objet métier conteneur de pièces jointes

Cet objet métier représente une pièce jointe. Il peut être de tout type complexe, mais doit posséder les attributs suivants :

Tableau 33. Attributs de l'objet métier conteneur de pièces jointes

Attribut	Type
contentID	xsd:string
contentType	xsd:string
encoding	xsd:string
rôle	xsd:string
schéma	xsd:string
versionSchéma	xsd:string
description	xsd:string
dataBytes	xsd:base64Binary
dataString	xsd:string

La pièce jointe se retrouve dans la valeur de l'attribut dataBytes ou dataString.

### Objet métier en-têtes d'empaquetage

Cet objet métier contient les en-têtes de niveau transport. Les en-têtes de niveau transport x-aux sont décrits dans la section «Contenu de l'en-tête de niveau transport», à la page 27. Cet objet métier doit posséder les attributs suivants.

Tableau 34. Attributs de l'objet métier en-têtes d'empaquetage

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

## Création d'objets métier pour le contenu de service RosettaNet

La liaison de données de l'empaquetage Backend Integration nécessite un objet de niveau supérieur. Vous pouvez créer l'objet de niveau supérieur selon les instructions de la section «Objet métier de niveau supérieur», à la page 68.

WebSphere Partner Gateway attend le contenu de service RosettaNet XML comme données utiles du message avec empaquetage Backend Integration. Pour créer l'objet conteneur de données utiles vous devez créer un objet métier données utiles qui représente le contenu de service RosettaNet XML. Vous pouvez créer l'objet métier contenu de service RosettaNet par l'un ou plusieurs de ces moyens :

- Si la structure du contenu de service PIP proposée par RosettaNet est au format de schéma XML, ce schéma peut être utilisé comme objet métier données utiles.
- Si la structure du contenu de service PIP proposée par RosettaNet est au format DTD, vous devez le convertir en schéma XML. Pour convertir une DTD de contenu de service PIP en schéma XML, suivez la procédure décrite dans le *Guide de configuration du concentrateur WebSphere Partner Gateway*. Ce schéma XML peut alors être utilisé comme objet métier données utiles.
- Si le PIP est au format DTD et que WebSphere Partner Gateway propose un package pour ce PIP, suivez la procédure ci-après pour créer l'objet métier contenu de service PIP.
  1. Accédez au package PIP de contenu de service RosettaNet correspondant à la version RNIF qui vous intéresse. Par exemple, si vous souhaitez créer un objet métier PIP3A4PurchaseOrderRequest, vous pouvez utiliser le fichier BCG\_Package\_RNSC1.0\_RNIFV02.02.zip figurant dans le répertoire du produit WebSphere Partner Gateway.
  2. À l'aide de WebSphere Integration Developer, importez le fichier ZIP dans les modules de WebSphere Process Server que vous avez configurés pour le PIP.
  3. Développez le dossier des types de données et cliquez avec le bouton droit de la souris sur l'objet métier qui correspond à l'élément racine de votre contenu de service. Par exemple, dans le cas d'une demande PIP 3A4, cliquez avec le bouton droit sur Pip3A4PurchaseOrder, dans le dossier des types de données, puis sélectionnez **Open with the Text Editor** (Ouvrir avec l'éditeur de texte).
  4. Dans l'éditeur de texte, modifiez comme suit l'élément suivant :

```
../../../../common/  
  
en  
  
./
```
  5. Sauvegardez vos modifications, puis fermez l'éditeur de texte.

## Fonctionnement de la liaison de données de l'empaquetage Backend Integration

Cette section explique comment la liaison de données de l'empaquetage Backend Integration crée un message avec empaquetage Backend Integration à partir d'un objet métier ou un objet métier à partir d'un message entrant.

### Conversion d'un message avec empaquetage Backend Integration en objet de niveau supérieur

Cette section explique comment un message avec empaquetage Backend Integration est converti en un objet métier utilisable par les services de WebSphere Process Server.

Comme l'indique la section «Objets métier enfant et de niveau supérieur», à la page 68, la liaison de données de l'empaquetage Backend Integration utilise un type spécifique d'objet métier de niveau supérieur. Vous devez créer cet objet métier avant d'utiliser la liaison de données.

L'objet métier de niveau supérieur est décrit dans la section «Objet métier de niveau supérieur», à la page 68.

**Types de message :** Les modalités de conversion du message de WebSphere Partner Gateway avec empaquetage Backend Integration dépendent du type de message. Ce message peut être de trois types :

- Un message XML comportant la balise racine <transport envelope> et l'URI "http://www.ibm.com/websphere/bcg/2003/v1.1/wbipackaging" ou "http://www.ibm.com/websphere/bcg/2003/v1.0/wbipackaging" .

Exemple de message contenu dans une enveloppe de transport :

```
<transport-envelope
  xmlns="http://www.ibm.com/websphere/bcg/2003/v1.1/wbipackaging">
  <payload encoding="base64" contentType="application/xml"
  contentId="1111111111">
    ...base64 encoded XML message...
  </payload>
  <attachment encoding="base64" contentType="text/xml"
  contentId="2222222222">
    ...base64 encoded XML attachment...
  </attachment>
  <attachment encoding="base64" contentType="application/pdf"
  contentId="3333333333">
    ...base64 encoded PDF attachment...
  </attachment>
</transport-envelope>
```

- Tout autre message XML
- Un message non XML

**Méthodes utilisées pour convertir le message :** Pour convertir un message JMS avec empaquetage Backend Integration en objet de niveau supérieur, utilisez la méthode read de BCGBackEndIntegrationJMSDataBindingImpl.

Pour convertir un message avec empaquetage Backend Integration en objet de niveau supérieur, vous pouvez utiliser les méthodes suivantes de la classe BCGBackEndIntegrationDataBindingUtil :

- read()  
Vous pouvez l'utiliser si le message avec empaquetage Backend Integration est contenu dans un flux d'entrée.
- setFromByteArray  
Vous pouvez l'utiliser si le message avec empaquetage Backend Integration est contenu dans un tableau d'octets.
- setFromString  
Vous pouvez l'utiliser si l'empaquetage Backend Integration est contenu dans une chaîne.

Avant d'appeler l'une de ces méthodes, vous pouvez personnaliser le mode de traitement de la liaison de données en appelant les méthodes suivantes :

- setTLOTypeName()  
Vous pouvez utiliser cette méthode sur l'objet pour le nommer et spécifier son URI.

- setBOPrefix()

Vous pouvez utiliser cette méthode pour spécifier le préfixe de l'objet métier. Si vous ne le faites pas, c'est le préfixe par défaut TLO\_ qui est utilisé.

- Si le type d'objet métier données utiles peut être déterminé, le nom de l'objet de niveau supérieur prend la forme suivante : <PréfixeObjetMétier><NomTypeObjetMétierDonnéesUtiles>.
- Si l'objet métier données utiles n'est pas déterminé (pour les données utiles non XML), le nom de l'objet de niveau supérieur est BCG\_TLO\_BackendPackaging (nom par défaut).

Si vous utilisez à la fois setTLOTypeName et setBOPrefix, setBOPrefix est ignoré.

Pour les données utiles non XML, si vous ne spécifiez pas le nom de l'objet de niveau supérieur à l'aide de setTLOTypeName avant d'appeler la méthode read, c'est l'objet de niveau supérieur par défaut qui est utilisé. Les données utiles ne sont pas converties en objet métier données utiles. A la place, les octets de données sont définis en l'état dans l'objet de niveau supérieur par défaut. L'objet de niveau supérieur par défaut est présenté dans la figure 19.

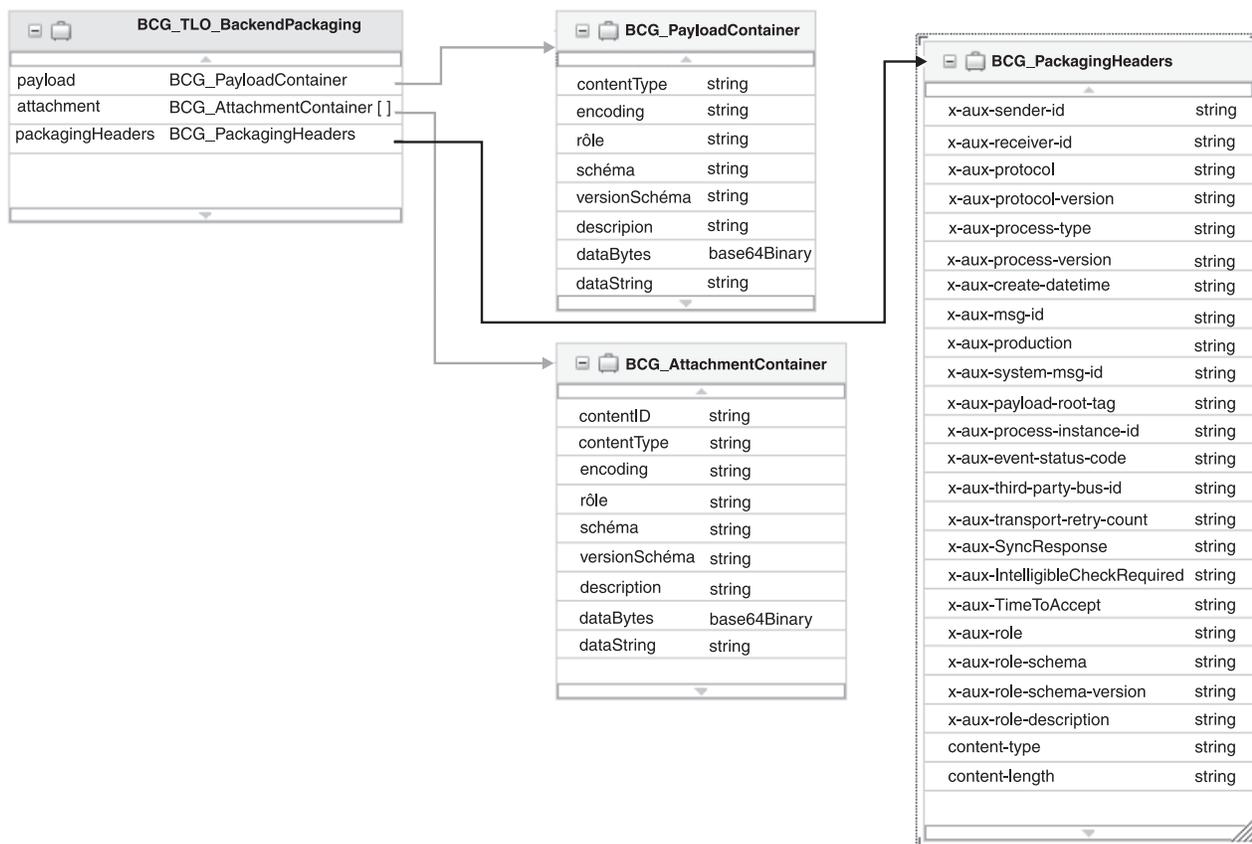


Figure 19. Objet métier de niveau supérieur par défaut

Si le message comporte des en-têtes de transport, vous devez lire ces en-têtes et les définir à l'aide de la méthode setxAuxHeaders (). La méthode setxAuxHeaders doit être appelée avant la méthode read(). Elle ne s'applique pas à la liaison de données de la classe BCGBackEndIntegrationJMSDataBindingImpl, car les en-têtes de transport JMS seront lus à partir du message JMS donné dans la méthode read.

**Fonctionnement de la conversion des messages :** La section suivante explique le fonctionnement des méthodes `read`, `setFromArray` et `setFromString` de la classe `BCGBackendIntegrationDataBindingUtil` et de la méthode `read` de la classe `BCGBackendIntegrationJMSDataBindingImpl`. La méthode :

1. Détermine le type de contenu XML à partir de l'en-tête `Content-type` (s'il existe dans le message). Exemple :

`Content-Type: application/xml`

Si l'en-tête `contentType` n'existe pas, la méthode inspecte les premiers octets du message pour déterminer son type.

La méthode obtient les données utiles en fonction du type de contenu, comme suit :

- Dans le cas d'un message XML, elle déserialise le message pour obtenir l'objet métier données utiles.
- Dans le cas d'un message dont la balise racine est `<transport envelope>`, elle décode les données utiles Base64 pour obtenir les octets de ces données, puis analyse le message pour déterminer s'il s'agit ou non d'un message XML.
  - S'il s'agit d'un message XML, elle déserialise les octets de données utiles pour obtenir l'objet métier données utiles.
  - S'il ne s'agit pas d'un message XML, elle ne déserialise pas les octets de données utiles. Ces derniers sont utilisés en l'état.

Elle décode également chaque pièce jointe Base64 contenue dans l'enveloppe de transport pour obtenir les octets de pièces jointes. Les octets de pièces jointes sont utilisés en l'état ; ils ne sont pas déserialisés.

- Dans le cas d'un message XML dont la balise racine est `<EventNotification>`, elle déserialise le message pour obtenir l'objet métier notification d'événements.
  - Dans le cas d'un message non XML, elle ne déserialise pas le message. Les octets sont utilisés en l'état.
2. Détermine le nom de l'objet métier de niveau supérieur.
    - Si les données utiles sont au format XML :
      - Si `setTLOTypeName` a été appelé avant `read`, `setFromArray` ou `setFromString`, le nom spécifié dans la méthode est directement utilisé pour créer le nom de l'objet de niveau supérieur.
      - Si vous avez utilisé la méthode `setBOPrefix`, le nom de l'objet de niveau supérieur est composé du préfixe spécifié et de l'élément racine XML des données utiles. L'URI est également obtenue à partir des données utiles.
      - Si vous n'avez pas utilisé la méthode `TLOTypeName` ou `setBOPrefix`, le nom de l'objet de niveau supérieur est composé du préfixe par défaut `TLO_` et de l'élément racine XML des données utiles.
    - Si les données utiles ne sont pas au format XML :
      - L'objet de niveau supérieur par défaut peut être utilisé.
      - La méthode `setTLOTypeName` peut être utilisée, mais elle doit se conformer à l'objet de niveau supérieur que vous avez créé ou à l'objet de niveau supérieur par défaut (`BCG_TLO_BackendPackaging`).
  3. Instancie l'objet de niveau supérieur.
  4. Instancie l'objet métier conteneur de données utiles et définit ses valeurs. La définition des valeurs dépend du type de message.
    - Dans le cas d'un message XML dont la balise racine est `<transport envelope>`, les attributs sont définis comme suit :
      - `contentType`

- Valeur de l'attribut contentType de la balise <payload>.
    - encoding
    - Valeur de l'attribut encoding de la balise <payload>.
    - payload
    - Nom de l'objet métier données utiles.
  - Dans le cas d'un message XML dont la balise racine est <event notification> ou de tout autre message XML, les attributs sont définis comme suit :
    - contentType n'est pas défini
    - encoding n'est pas défini
    - payload
    - Nom de l'objet métier données utiles.
  - Dans le cas d'un message non XML, les attributs sont définis comme suit :
    - contentType n'est pas défini
    - encoding n'est pas défini
    - dataBytes
    - Si le message entrant est reçu sous forme d'octets, tout le jeu d'octets est défini en tant que valeur de cet attribut.
    - dataString
    - Si le message entrant est reçu sous forme de chaîne, toute la chaîne est définie en tant que valeur de cet attribut.
    - payload n'est pas défini.
5. Instancie les objets métier pièce jointe et définit leurs valeurs comme suit :
- contentType
  - Valeur de l'attribut contentType de la balise <attachment>.
  - encoding
  - Valeur de l'attribut encoding de la balise <attachment>.
  - contentId
  - Valeur de l'attribut contentId de la balise <attachment>.
  - dataBytes
  - Octets Base64 décodés de la pièce jointe.
6. Instancie l'objet métier en-têtes de transport et définit les en-têtes de transport lus à partir du message. Le nom de cet objet métier est défini en tant que valeur de l'attribut packagingHeaders de l'objet de niveau supérieur.

L'Exemple de code illustre l'utilisation des méthodes.

### **Conversion de l'objet de niveau supérieur en message avec empaquetage Backend Integration**

Cette section explique comment un objet métier provenant de WebSphere Process Server est sérialisé sous forme de message avec empaquetage Backend Integration. La liaison de données place le contenu et les pièces jointes dans une enveloppe de transport au format Base64. Elle place également les en-têtes de transport de l'objet de niveau supérieur dans une enveloppe sous forme de chaîne.

Avant d'appeler la liaison de données de l'empaquetage Backend Integration, un service doit créer un objet de niveau supérieur, comme indiqué dans la section «Objet métier de niveau supérieur», à la page 68.

**Méthodes utilisées pour convertir l'objet :** Un service WebSphere Process Server appelle la liaison de données de l'empaquetage Backend Integration en lui envoyant l'objet de niveau supérieur. Pour convertir cet objet en message JMS avec empaquetage Backend Integration, utilisez la méthode `write` de la classe `BCGBackEndIntegrationJMSDataBindingImpl`. Pour convertir cet objet en message avec empaquetage Backend Integration, utilisez la méthode `write`, `getAsByteArray` ou `getAsString` de la classe `BCGBackEndIntegrationDataBindingUtil`.

Avant d'appeler l'une de ces méthodes, vous pouvez appeler :

- La Méthode `setPackagingSchema`  
Cette méthode indique quel schéma d'empaquetage utiliser pour la construction du message avec empaquetage Backend Integration.
- La Méthode `setOptions`  
Cette méthode donne des informations (telles que la balise racine des données utiles) pouvant être utilisées pour la sérialisation du message. Concernant les options que vous pouvez définir, voir «Méthode `setOptions`», à la page 82.

**Fonctionnement de la conversion des objets :** La section suivante explique le fonctionnement des méthodes `write`, `getAsByteArray` et `getAsString` de la classe `BCGBackEndIntegrationDataBindingUtil` et de la méthode `write` de la classe `BCGBackEndIntegrationJMSDataBindingImpl`. La méthode :

1. Détermine le type de contenu des données utiles. Cette information est obtenue à partir de l'objet métier conteneur de données utiles. Le traitement des données utiles dépend de son type de contenu :
  - Données utiles XML  
L'attribut `dataBytes` ou `dataString` du conteneur de données utiles (s'il existe) doit être vide et l'objet métier conteneur de données utiles doit posséder au moins un attribut de type complexe. Il s'agit d'une référence à l'objet métier données utiles. L'objet métier est sérialisé et utilisé en tant que données utiles XML. Si plusieurs attributs de type complexe ont été définis, le premier d'entre eux est considéré comme données utiles XML.
  - EventNotification  
Le premier attribut à valeur définie doit être de type `EventNotification`. Il s'agit d'une référence à l'objet métier notification d'événements. L'objet métier est sérialisé et utilisé en tant que données utiles XML.
  - Non XML  
Si l'attribut `dataBytes` n'est pas vide, sa valeur est utilisée en tant que données utiles.  
Si l'attribut `dataString` n'est pas vide, sa valeur est utilisée en tant que données utiles.
2. Sérialise l'objet métier données utiles s'il est de type XML ou `EventNotification`.
3. Encode les données utiles en Base64, en fonction de leur type :
  - Si l'objet métier conteneur de données utiles comporte une valeur définie pour l'attribut `dataByte`, cette valeur (octets de données utiles) est codée en Base64.
  - Si l'objet métier conteneur de données utiles comporte une valeur définie pour l'attribut `dataString`, les octets sont extraits en fonction de l'attribut `contentType` de l'objet métier, puis codés en Base64.
  - Si les données utiles ont été sérialisées en objet métier, les octets sont extraits en fonction de l'attribut `contentType` de l'objet métier, puis codés en Base64.
4. Construit un document XML en fonction du schéma d'empaquetage spécifié.
  - a. La balise `<transport envelope>` est ajoutée.

- b. La balise <payload>, un élément enfant de la balise <transport envelope>, est définie à l'aide de la chaîne codée en base64 à l'étape 3, à la page 77.
5. Traite les pièces jointes.
- Si l'objet conteneur de pièces jointes comporte une valeur définie pour l'attribut dataByte, les octets sont codés en base64 et la chaîne est définie en tant que valeur de la balise <attachment>.
- Si l'objet conteneur de pièces jointes comporte une valeur définie pour l'attribut dataString, les octets sont extraits en fonction de l'attribut contentType de l'objet métier. Ces octets sont alors codés en base64 et la chaîne est définie en tant que valeur de la balise <attachment>.

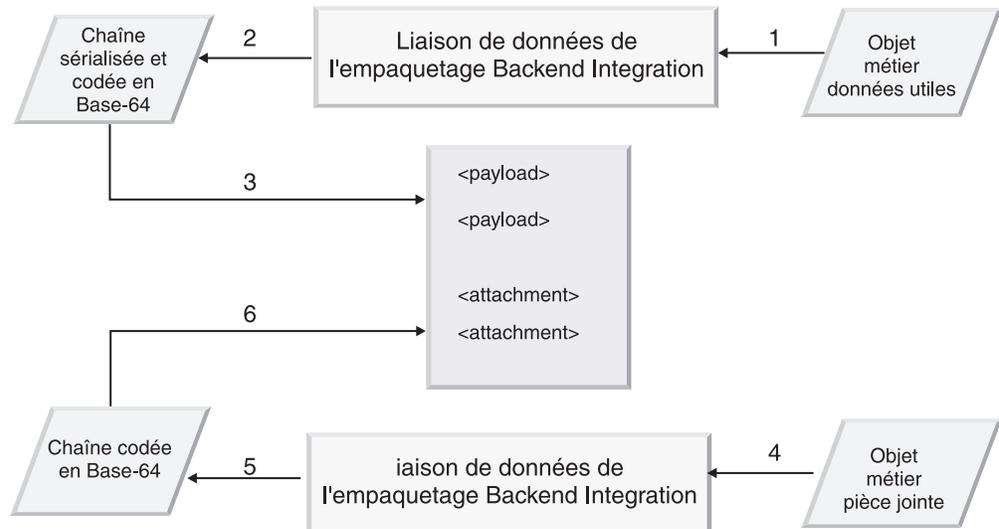


Figure 20. Ajout, par la liaison de données, des objets métier convertis au message avec empaquetage Backend Integration

L'Exemple de code illustre l'utilisation des méthodes.

## Classe BCGBackendIntegrationDataBindingUtil

Cette section décrit la classe BCGBackendIntegrationDataBindingUtil et ses méthodes. Vous pouvez utiliser la classe en l'état, ou, si vous souhaitez personnaliser le mode de traitement de la liaison de données, vous pouvez créer une classe de liaison.

### Interface DataBinding

BCGBackendIntegrationDataBindingUtil implémente l'interface DataBinding suivante :

```

DataBinding
public interface commonj.connector.runtime.DataBinding extends Serializable {
    public DataObject getDataObject()throws
    commonj.connector.runtime.DataBindingException;
    public void setDataObject(DataObject dataObject) throws
    commonj.connector.runtime.DataBindingException;
}
  
```

### Méthodes

Les méthodes de la classe BCGBackendIntegrationDataBindingUtil sont décrites dans la présente section.

La classe `BCGBackEndIntegrationDataBindingUtil` comporte deux types de méthode :

- Celles qui créent un objet données à partir d'un tableau d'octets, d'une chaîne ou d'un flux de données :
  - `read(InputStream)`
  - `setFromArray(byte[ ])`
  - `setFromString(String)`
  - `setxAuxHeaders(HashMap)`
  - `setBOPrefix(String)`
  - `setTLOTypeName(String, String)`
  - `getDataObject()`

**Remarques :**

1. Pour créer l'objet données, vous pouvez utiliser l'une des méthodes suivantes :

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

Avant d'appeler la méthode permettant de créer un objet données, le programme peut appeler l'une des méthodes suivantes :

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

2. Si la méthode `setTLOTypeName` est appelée, la méthode `setBOPrefix` n'a plus aucune importance et est ignorée.
3. La méthode `getDataObject` permet d'obtenir l'objet métier de niveau supérieur.

- Celles qui convertissent un objet données en octet, chaîne ou flux de données :
  - `setDataObject(DataObject)`
  - `setPackagingSchema(String)`
  - `setOptions(HashMap)`
  - `getAsString()`
  - `getAsByteArray()`
  - `write(OutputStream)`
  - `getxAuxHeaders()`

**Méthode `getAsByteArray`**

Cette méthode renvoie la valeur de `byte[ ]` obtenue à partir de l'objet de niveau supérieur.

**Syntaxe :**

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

**Méthode `getAsString`**

Cette méthode renvoie l'objet de niveau supérieur sous forme de chaîne.

**Syntaxe :**

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

## Méthode getDataObject

Cette méthode renvoie l'objet données, s'il a déjà été instancié.

### Syntaxe :

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

## Méthode getXAuxHeader

Cette méthode renvoie les en-têtes x-aux de l'enveloppe de transport. Voir «Contenu de l'en-tête de niveau transport», à la page 27 pour plus d'informations sur les en-têtes x-aux.

### Syntaxe :

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

## Méthode read

Cette méthode récupère un flux d'entrée, lit ce flux, puis le convertit en objet métier de niveau supérieur. Les actions accomplies par la méthode read sont détaillées dans la section «Conversion d'un message avec empaquetage Backend Integration en objet de niveau supérieur», à la page 72.

### Syntaxe :

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

### Paramètre :

#### inputStream

Flux d'entrée dans lequel la méthode lit les données.

## Méthode setBOPrefix

Si l'objet de niveau supérieur n'a pas été spécifié, vous pouvez définir un préfixe à l'aide de la méthode setBOPrefix(). Si vous ne définissez pas de préfixe, c'est le préfixe par défaut TLO\_ qui est utilisé.

### Remarques :

1. Dans le cas de données utiles XML, l'espace de nom des données utiles est utilisé comme espace de nom de l'objet de niveau supérieur.
2. Si les données utiles ne sont pas au format XML, l'espace de nom par défaut "http://ibm.com/websphere/bcg/2005/wbi/bo" est utilisé comme espace de nom de l'objet de niveau supérieur.

### Syntaxe :

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

## Méthode setDataObject

Cette méthode définit l'objet données.

### Syntaxe :

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

### Paramètre :

#### topLevelbo

Objet métier de niveau supérieur. Voir «Objet métier de niveau supérieur», à la page 68 pour plus d'informations.

## Méthode setDebugLevel

Par défaut, le niveau de débogage est défini sur "error". Cette méthode vous permet de modifier ce niveau.

### Syntaxe :

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

### Paramètre :

#### debugLevel

Les valeurs de niveau de débogage possibles sont :

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

## Méthode setFromArray

Cette méthode récupère les octets de données utiles ou de données de l'enveloppe de transport et crée un objet de niveau supérieur. Avant d'appeler cette méthode, vous pouvez appeler l'une ou plusieurs des méthodes suivantes :

- Méthode setTLOTypeName
- Méthode setBOPrefix
- Méthode setxAuxHeader

### Syntaxe :

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

### Paramètre :

#### dataBytes

Données utiles sous forme d'octets.

## Méthode setFromString

Cette méthode convertit un paramètre sous forme de chaîne en objet de niveau supérieur. Avant d'appeler cette méthode, vous pouvez appeler l'une ou plusieurs des méthodes suivantes :

- Méthode setTLOTypeName
- Méthode setBOPrefix
- Méthode setOptions

### Syntaxe :

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

### Paramètre :

#### fromString

Données utiles ou objet données de niveau supérieur sous forme de chaîne.

## Méthode setOptions

Cette méthode définit les options obligatoires.

### Syntaxe :

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

### Paramètre :

#### Hashmap

Le paramètre Hashmap peut comporter les valeurs de clé suivantes :

- BCG\_ROOT\_NODE\_NAME  
Nom racine à utiliser pour générer les données utiles.
- BCG\_APPEND\_DTD  
Attribut indiquant s'il faut ajouter une balise DTD aux données utiles. Sa valeur peut être BCG\_APPEND\_DTD\_TRUE ou BCG\_APPEND\_DTD\_FALSE. La valeur par défaut est BCG\_APPEND\_DTD\_FALSE.
- BCG\_SYSTEM\_ID  
ID système de la DTD. Si la valeur de BCG\_APPEND\_DTD est BCG\_APPEND\_DTD\_TRUE, cette valeur peut être définie. Si la valeur de BCG\_APPEND\_DTD est BCG\_APPEND\_DTD\_FALSE, cette valeur est ignorée.
- BCG\_PUBLIC\_ID  
ID public de la DTD. Si la valeur de BCG\_APPEND\_DTD est BCG\_APPEND\_DTD\_TRUE, cette valeur est facultative. Si la valeur de BCG\_APPEND\_DTD est BCG\_APPEND\_DTD\_FALSE, cette option est ignorée.

## Méthode setPackagingSchema

Cette méthode définit le schéma d'emballage à utiliser lors de la génération de l'emballage Backend Integration. Vous pouvez spécifier l'un des schémas d'emballage suivants :

- <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>

Si le schéma d'emballage n'est pas spécifié, c'est <http://www.ibm.com/websphere/bcg/2003/v1.1/wbipackaging> qui est utilisé.

Le schéma d'emballage est utilisé lorsque vous appelez la méthode write.

### Syntaxe :

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

### Paramètre :

#### packagingSchema

Nom du schéma de l'objet métier de niveau supérieur.

## Méthode setTLOTypeName

Cette méthode définit le nom de l'objet de niveau supérieur ainsi que son espace de nom.

- Si les données utiles ne sont pas au format XML et que la méthode setTLOTypeName n'a pas été appelée, c'est le nom d'objet de niveau supérieur par défaut qui est utilisé.
- Si les données utiles figurent dans un document XML, le nom de l'objet de niveau supérieur est dérivé comme suit :

boPrefix + < *Balise racine des données utiles XML* >.

Par exemple, si la méthode setBOPrefix("IBM") est appelée et que l'élément racine des données utiles est "ABC", le nom de l'objet de niveau supérieur sera "IBM\_ABC".

Le préfixe peut être défini à l'aide de la Méthode setBOPrefix. Si le nom de l'objet de niveau supérieur n'est pas défini et que les données utiles ne sont pas contenues dans un document XML, vous devez créer un type de données boPrefix + < *Balise racine* >.

### Syntaxe :

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

### Paramètres :

#### tns

Espace de nom de l'objet métier de niveau supérieur.

#### typeName

Type de l'objet de niveau supérieur.

## Méthode setxAuxHeader

Cette méthode définit les en-têtes de transport. Si aucun en-tête de transport n'est défini, aucun ne peut être lu. Voir «Contenu de l'en-tête de niveau transport», à la page 27 pour plus d'informations sur les en-têtes x-aux.

### Syntaxe :

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

## Méthode write

Cette méthode écrit l'objet de niveau supérieur dans le flux de sortie spécifié. Les actions accomplies par la méthode write sont détaillées dans la section «Conversion de l'objet de niveau supérieur en message avec empaquetage Backend Integration», à la page 76.

### Syntaxe :

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

### Paramètre :

#### outputStream

Flux de sortie dans lequel la méthode écrit l'objet données.

## Exemple de code

Le code suivant montre comment créer un objet de niveau supérieur à partir d'un flux d'entrée :

```
BCGBackEndIntegrationDataBindingUtil util = BCGBackEndIntegrationDataBindingUtil ();
util.setTLOTypeName ("TLO_URIName","TLOName");
//util.setBOPrefix ("BO_prefix");
//setBoPrefix est commenté car la méthode setTLOTypeName () est utilisée
util.read (inputStream);
DataObject tlo = util.getDataObject ();
```

Le code suivant montre comment obtenir un flux à partir d'un objet données :

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

---

## Classe BCGBackEndIntegrationJMSDataBindingImpl

Cette classe crée un objet métier à partir d'un message JMS contenant des données utiles ou écrit l'objet métier dans un message JMS. Lorsqu'elle lit l'objet métier à partir d'un message JMS, elle lit les en-têtes de transport JMS avant de créer l'objet métier. Lorsqu'elle écrit l'objet métier dans un message JMS, elle écrit les en-têtes de transport JMS si cet objet possède un objet métier enfant pour les en-têtes d'empaquetage.

BCGBackEndIntegrationJMSDataBindingImpl étend la classe BCGBackEndIntegrationDataBindingUtil décrite dans la section «Classe BCGBackEndIntegrationDataBindingUtil», à la page 78.

## Interface JMSDataBinding

BCGBackEndIntegrationJMSDataBindingImpl implémente l'interface JMSDataBinding :

```
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;

}
```

## Méthodes

Les méthodes de la classe BCGBackEndIntegrationJMSDataBindingImpl sont décrites dans la présente section.

### getMessageType

Cette méthode renvoie le type de message.

**Syntaxe :**

```
public int getMessageType()
```

## **isBusinessException**

Cette méthode indique s'il existe des exceptions métier.

### **Syntaxe :**

```
public boolean isBusinessException()
```

### **read**

Cette méthode lit les en-têtes de transport et, en fonction du type de message, crée l'objet métier de niveau supérieur. Les actions accomplies par la méthode read sont détaillées dans la section «Conversion d'un message avec empaquetage Backend Integration en objet de niveau supérieur», à la page 72.

Dans cette méthode, il n'est pas nécessaire de définir AuxHeaders et de lire le message.

### **Syntaxe :**

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

### **Paramètre :**

#### **jmsMessage**

Message contenant les données utiles et les en-têtes de transport.

## **setBusinessException**

Cette méthode définit un indicateur d'occurrence d'exception métier.

### **Syntaxe :**

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

### **write**

Cette méthode écrit l'objet données dans le message et, en fonction du type de message, définit les en-têtes du message. Les actions accomplies par la méthode write sont détaillées dans la section «Conversion de l'objet de niveau supérieur en message avec empaquetage Backend Integration», à la page 76.

### **Syntaxe :**

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

---

## **Insertion de classes de liaison de données dans l'implémentation de votre composant**

Lors du développement de composants WebSphere Process Server à l'aide de WebSphere Integration Developer, vous pouvez utiliser les classes BCGBackEndIntegrationJMSDataBindingImpl et BCGBackEndIntegrationDataBindingUtil proposées par WebSphere Partner Gateway. Ces classes sont disponibles dans le fichier databinding.jar, qui figure dans le répertoire Integration\WBI\WebSphereProcessServer\DataBinding de l'image du produit.

Afin d'utiliser ces classes dans le projet Business Integration, vous devez vous assurer que databinding.jar figure bien dans le fichier EAR du projet Business Integration que vous allez déployer sur WebSphere Process Server. Pour inclure le fichier databinding.jar dans votre fichier EAR généré, vous pouvez consulter les centres de documentation de WebSphere Process Server et de WebSphere

Integration Developer, ou suivre la procédure indiquée dans cette section. Avant de suivre cette procédure, vérifiez que vous vous trouvez bien dans la perspective Business Integration et que vous avez créé le projet Business Integration dans lequel vous allez importer le fichier .jar.

1. A partir de WebSphere Integration Developer, importez le fichier databinding.jar dans votre projet. Vous devez l'importer en tant que "J2EE Utility jars" (fichiers jar utilitaires J2EE). Une fois l'importation correctement effectuée, WebSphere Integration Developer crée implicitement un projet pour le fichier databinding.jar
2. Ajoutez ce projet en tant que dépendance dans votre projet Business Integration. Pour ajouter le projet en tant que dépendance :

**Remarque :** La procédure ci-dessous est un exemple parmi d'autres permettant d'ajouter le projet en tant que dépendance. Reportez-vous à la documentation de WebSphere Process Server pour en savoir plus sur les autres procédures possibles.

- a. Cliquez deux fois sur votre projet Business Integration.  
L'éditeur de dépendance apparaît.
  - b. Développez le dossier **Java**.
  - c. Cliquez sur le bouton **Add** (Ajouter).  
La liste des projets s'affiche.
  - d. Sélectionnez le projet implicitement créé par WebSphere Integration Developer pour le fichier databinding.jar (à l'étape 1).  
Ce projet s'affiche en tant que dépendance sous Java.
  - e. Sélectionnez-le.
  - f. Cochez la case **Deploy with Module** (Déployer avec le module) .
3. Fermez l'éditeur de dépendance et sauvegardez l'espace de travail.

Au terme de la procédure, le fichier databindng.jar figure dans votre fichier EAR généré.

---

## Chapitre 4. Intégration de WebSphere Process Server avec HTTP comme protocole de transport

Ce chapitre décrit le processus d'intégration de WebSphere Partner Gateway à WebSphere Process Server via le protocole de transport HTTP. Ce chapitre contient les informations suivantes :

- «Fonctionnement de l'envoi de messages vers WebSphere Process Server»
- «Fonctionnement de l'envoi de messages depuis WebSphere Process Server», à la page 89
- «Envoi de documents vers WebSphere Process Server», à la page 91
- «Envoi de documents depuis WebSphere Process Server», à la page 93

Ce chapitre ne décrit pas le traitement des requêtes SOAP envoyées par HTTP. Pour plus d'informations sur l'envoi et la réception de requêtes SOAP via HTTP, voir Chapitre 6, «Intégration de WebSphere Process Server à SOAP/HTTP», à la page 107.

---

### Fonctionnement de l'envoi de messages vers WebSphere Process Server

Cette section explique comment WebSphere Partner Gateway reçoit un message provenant d'un partenaire externe, puis l'envoie à WebSphere Process Server par le biais du protocole HTTP.

#### Remarques :

1. Tous les types de document, à l'exception de RosettaNet, peuvent ne comporter aucun emballage lorsqu'ils sont envoyés de WebSphere Partner Gateway vers WebSphere Process Server. Les documents RosettaNet doivent obligatoirement être associés à l'emballage Backend Integration.
2. Tous les types de document, à l'exception de SOAP (demandes de services Web), peuvent être associés à l'emballage Backend Integration lorsqu'ils sont envoyés de WebSphere Partner Gateway vers WebSphere Process Server. Pour les requêtes SOAP, vous devez obligatoirement spécifier None en tant qu'emballage.

Pour plus d'informations sur l'envoi et la réception de documents SOAP via HTTP, voir Chapitre 6, «Intégration de WebSphere Process Server à SOAP/HTTP», à la page 107.

Pour que WebSphere Process Server puisse recevoir un message provenant de WebSphere Partner Gateway, vous devez écrire un servlet qui permettra de récupérer le message et de le convertir en objet métier.

WebSphere Partner Gateway envoie des messages à l'URL configuré dans la zone Destination HTTP "cible" du partenaire interne, dans la connexion de partenaire. Le servlet effectue ses interrogations sur cette URL et reçoit le message. Le service qui recevra le document de gestion converti possède une liaison d'exportation SCA. La figure 21, à la page 88 montre comment WebSphere Partner Gateway traite un message envoyé par un partenaire externe, puis l'envoie au servlet HTTP où l'objet métier appelle le service, par le biais de la liaison d'exportation.

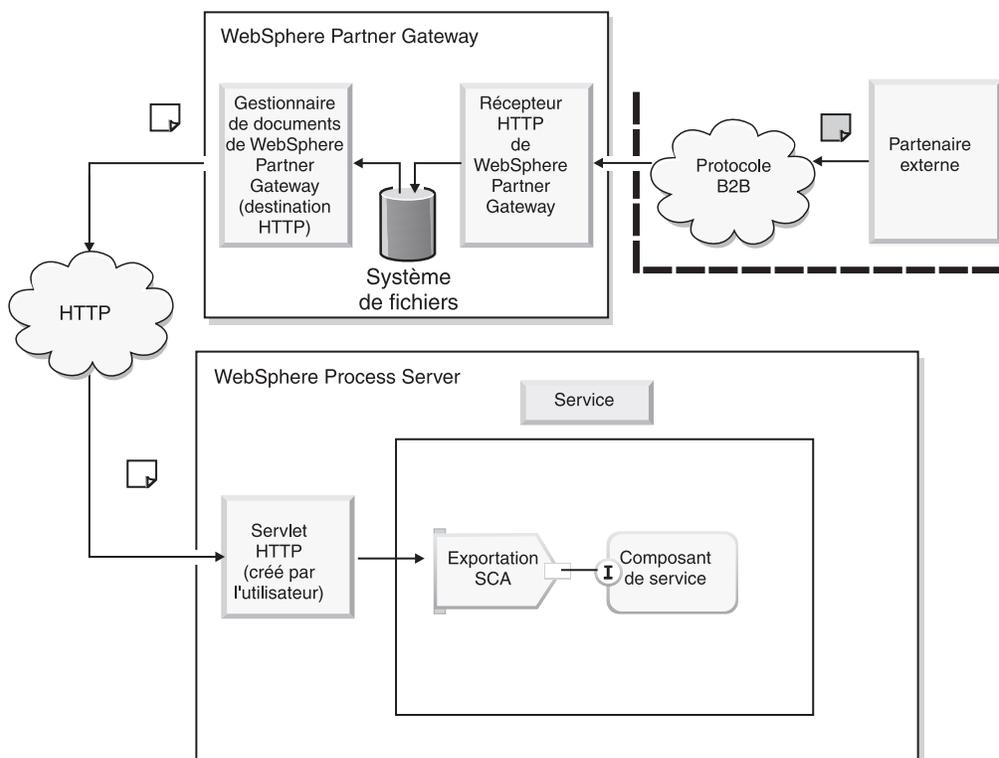


Figure 21. Fonctionnement de l'envoi d'un message de WebSphere Partner Gateway au servlet HTTP

Le servlet HTTP effectue les tâches suivantes :

1. En fonction du message de requête qu'il reçoit, le servlet détermine le service et la méthode de ce service à appeler.
2. Il construit un objet métier à partir du message entrant, comme décrit dans la section «Création d'un servlet», à la page 92.
3. Il appelle le service à l'aide de l'API cliente SCA appropriée :
  - a. Si la méthode du service SCA n'autorise que les demandes, aucun objet métier de type réponse n'est attendue. Si le servlet HTTP est en mesure d'appeler le service SCA, il renvoie un code d'état HTTP 200 à WebSphere Partner Gateway. S'il n'est pas en mesure de l'appeler, il renvoie le code d'erreur d'état HTTP approprié.
  - b. Si la méthode du service SCA combine la demande et la réponse, ce service répond en renvoyant un objet métier. Le servlet HTTP sérialise cet objet métier sous forme de message, puis il renvoie ce message à WebSphere Partner Gateway dans une réponse HTTP. Si, pour une raison ou pour une autre, le servlet HTTP n'est pas en mesure d'appeler le service SCA avec succès, il renvoie le code d'erreur d'état HTTP approprié.

Par exemple, lorsque WebSphere Partner Gateway reçoit un message cXML synchrone provenant de partenaires, il envoie ce message à WebSphere Process Server via le protocole HTTP. WebSphere Partner Gateway attend de WebSphere Process Server qu'il renvoie une réponse cXML synchrone sur la même connexion HTTP.

**Remarque :** WebSphere Partner Gateway prend en charge les appels de type demande seule pour l’empaquetage Backend Integration. Cependant, vous pouvez développer des flux combinant demande et réponse avec la liaison de données de l’empaquetage Backend Integration (par le biais des exits utilisateur de WebSphere Partner Gateway). La Classe `BCGBackEndIntegrationDataBindingUtil` peut servir de base à la conversion d’un objet métier de type réponse en un flux de réponse.

---

## Fonctionnement de l’envoi de messages depuis WebSphere Process Server

Cette section explique comment WebSphere Partner Gateway reçoit un message provenant de WebSphere Process Server par le biais du protocole HTTP, puis l’envoi à un partenaire externe.

**Remarques :**

1. Tous les types de document, à l’exception de RosettaNet et des documents binaires, peuvent ne comporter aucun empaquetage lorsqu’ils sont envoyés de WebSphere Partner Server vers WebSphere Process Gateway. Les documents RosettaNet et binaires doivent obligatoirement être associés à l’empaquetage Backend Integration.
2. Tous les types de document, à l’exception de SOAP (demandes de services Web), peuvent être associés à l’empaquetage Backend Integration lorsqu’ils sont envoyés de WebSphere Partner Server vers WebSphere Process Gateway. Pour les requêtes SOAP, vous devez obligatoirement spécifier None en tant qu’empaquetage.

Pour plus d’informations sur l’envoi et la réception de documents SOAP via HTTP, voir Chapitre 6, «Intégration de WebSphere Process Server à SOAP/HTTP», à la page 107.

Lorsque WebSphere Process Server envoie un message à WebSphere Partner Gateway, il utilise un composant qui envoie une requête HTTP POST à un récepteur HTTP de WebSphere Partner Gateway. Pour envoyer une requête HTTP POST à un récepteur HTTP de WebSphere Partner Gateway, vous devez développer ce composant et l’exposer en tant que service.

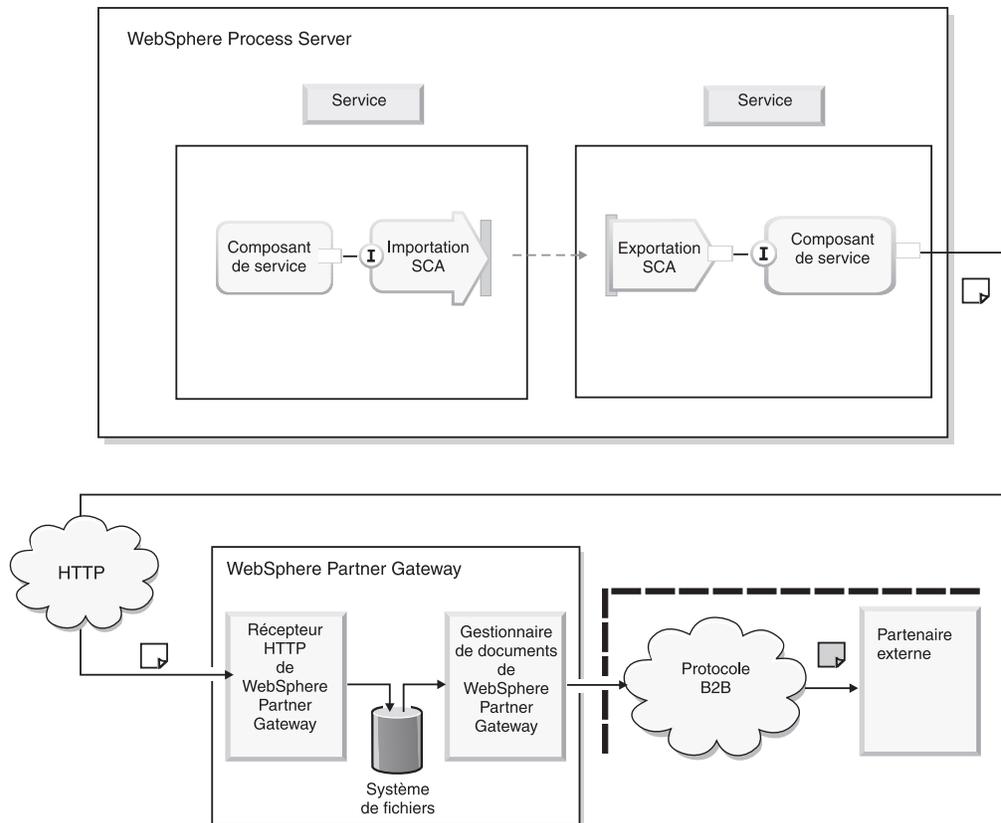


Figure 22. Fonctionnement de l'envoi d'un message de WebSphere Process Server au récepteur HTTP de WebSphere Partner Gateway

Les services WebSphere Process Server procèdent comme suit pour envoyer des messages à WebSphere Partner Gateway :

1. Un service SCA WebSphere Process Server qui doit envoyer des documents de gestion à WebSphere Partner Gateway via HTTP utilise sa liaison d'importation SCA pour appeler le composant qui effectuera la requête HTTP POST. La liaison d'importation SCA du service appelle le composant avec un objet métier.
2. Le composant reçoit l'objet métier sur son exportation. Il sérialise alors l'objet métier sous forme de document de gestion. Voir «Création du composant nécessaire à la conversion et à l'envoi du message», à la page 94.
3. Le composant transmet le message en envoyant une requête HTTP POST à l'URL du récepteur HTTP de WebSphere Partner Gateway.
4. WebSphere Partner Gateway reçoit ce message sur son récepteur HTTP.

La réponse de WebSphere Partner Gateway sera différente en fonction des nécessités du message : s'il nécessite une réponse synchrone ou s'il nécessite une réponse au niveau du transport HTTP uniquement. Le récepteur HTTP possède un paramètre de configuration SyncCheck qui lui permet de déterminer la réponse à apporter. Vous pouvez configurer SyncCheck lors de la création du récepteur ou modifier le récepteur pour définir ce paramètre. Reportez-vous au *Guide de configuration du concentrateur WebSphere Partner Gateway* pour plus d'informations sur le paramètre de configuration SyncCheck.

- a. Si WebSphere Partner Gateway détecte que le message provenant de WebSphere Process Server nécessite une réponse au niveau du transport

- uniquement, le récepteur de WebSphere Partner Gateway répond en renvoyant le code d'état HTTP approprié. Ensuite, il route le document de gestion vers le partenaire externe.
- b. S'il détecte que le message provenant de WebSphere Process Server nécessite une réponse sous forme de document de gestion, il garde la connexion HTTP ouverte. WebSphere Partner Gateway route le document vers un partenaire externe. Le partenaire externe reçoit le document demandé et envoie un document en réponse à WebSphere Partner Gateway. WebSphere Partner Gateway renvoie le document de gestion en réponse à WebSphere Process Server, en tant que réponse HTTP.
5. Le composant reçoit la réponse HTTP. Il détermine si la réponse est uniquement un code d'état HTTP ou également un document de gestion :
- a. Si la réponse est un code d'état HTTP indiquant la réussite de l'opération, il renvoie l'appel. Le service SCA de WebSphere Process Server (qui a appelé le composant) poursuit son traitement. En revanche, si le code d'état HTTP indique un échec, le composant renvoie l'erreur appropriée. Le service SCA de WebSphere Process Server traite l'erreur de façon appropriée.
  - b. Si la réponse est un document de gestion, le composant convertit ce document en objet métier. Ensuite, il renvoie l'objet métier au service SCA de WebSphere Process Server, qui traite l'objet métier de type réponse. Par exemple, dans le cas des messages cXML synchrones provenant de WebSphere Process Server et destinés à un partenaire externe, WebSphere Partner Gateway envoie le message au partenaire et reçoit la réponse de manière synchrone. WebSphere Partner Gateway renvoie ensuite la réponse à WebSphere Process Server, en tant que réponse à la requête HTTP initialement envoyée par WebSphere Process server.

**Remarque :** WebSphere Partner Gateway prend en charge les appels de type demande seule pour l'empaquetage Backend Integration. Cependant, vous pouvez développer des flux combinant demande et réponse avec la liaison de données de l'empaquetage Backend Integration (par le biais des exits utilisateur de WebSphere Partner Gateway). La Classe `BCGBackEndIntegrationDataBindingUtil` peut servir de base à la conversion d'un objet métier de type réponse en un flux de réponse.

---

## Envoi de documents vers WebSphere Process Server

Cette section décrit les étapes à accomplir pour permettre à WebSphere Process Server d'accepter les documents envoyés par WebSphere Partner Gateway.

### Configuration de WebSphere Partner Gateway

Vous devez configurer WebSphere Partner Gateway selon la procédure décrite dans le *Guide de configuration du concentrateur WebSphere Partner Gateway*. Lors de la configuration de WebSphere Partner Gateway, prenez note des indications suivantes :

1. Ajoutez l'indicateur `com.ibm.bcg.server.sync.SoapSyncHdlr` au paramètre de configuration `syncCheck` du récepteur HTTP (si vous envisagez de traiter des demandes bidirectionnelles).
2. Assurez-vous que la destination HTTP du partenaire interne indique l'URL qu'utilisera WebSphere Partner Gateway pour envoyer des messages à WebSphere Process Server.

La destination doit pointer vers le servlet HTTP déployé sur WebSphere Process Server.

## Configuration de WebSphere Process Server

Cette section indique comment configurer WebSphere Process Server pour qu'il puisse recevoir les messages provenant de WebSphere Partner Gateway.

### Création d'un objet métier

Si vous utilisez la classe `BCGBackendIntegrationDataBindingUtil` pour convertir le message en objet métier, vous devez utiliser l'objet métier de niveau supérieur décrit dans la section «Objets métier enfant et de niveau supérieur», à la page 68. Vous pouvez créer l'objet métier à l'aide de l'éditeur d'objet métier ou importer le fichier `.xsd(BCG_TLO_BackendPackaging.xsd)` disponible sur le support d'installation.

Si vous n'utilisez pas l'empaquetage Backend Integration (en d'autres termes, si l'empaquetage spécifié pour le document est `None - Aucun`), vous devez créer l'objet métier en fonction des besoins pour l'échange de documents.

### Création d'un servlet

Cette section décrit le servlet que vous devez créer pour que WebSphere Partner Gateway puisse envoyer des messages à WebSphere Process Server via le protocole de transport HTTP. Concevez le servlet pour les opérations suivantes :

- Réception des messages envoyés par WebSphere Partner Gateway
- Conversion du document de gestion en objet métier
- Appel du service approprié sur WebSphere Process Server

Si vous utilisez l'empaquetage Backend Integration, vous pouvez utiliser la Classe `BCGBackEndIntegrationDataBindingUtil` pour convertir un message de requête HTTP en objet métier attendu par votre service. Vous pouvez également utiliser cette classe pour créer un message de réponse HTTP à partir de l'objet métier de type réponse renvoyé par votre service. L'extrait de code ci-après offre un exemple d'utilisation de la classe `BCGBackEndIntegrationDataBindingUtil` pour convertir un message de requête HTTP en objet métier de type demande. Cet exemple utilise la Méthode `read` et la Méthode `getDataObject`. Si vous connaissez déjà le nom de l'objet métier de niveau supérieur, vous pouvez également appeler la Méthode `setTLOTypeName`.

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

    // request.getInputStream() indique le flux d'entrée du
// empaquetage Backend Integration reçu par le servlet
dataBinding.read(request.getInputStream());
    rootBO = dataBinding.getDataObject();

}
catch(Exception exp){
    System.out.println("Une erreur est survenue lors de la création de l'objet
métier demandé : " + exp);
}
```

Après avoir créé le servlet, déployez-le. L'adresse URL à laquelle le servlet reçoit le message doit être spécifiée en tant qu'URL dans la destination cible configurée pour la connexion de partenaire dans WebSphere Partner Gateway.

---

## Envoi de documents depuis WebSphere Process Server

Cette section décrit les étapes à accomplir pour permettre à WebSphere Process Server d'envoyer des documents à WebSphere Partner Gateway.

### Configuration de WebSphere Partner Gateway

Vous devez configurer WebSphere Partner Gateway selon la procédure décrite dans le *Guide de configuration du concentrateur WebSphere Partner Gateway*. Lors de la configuration de WebSphere Partner Gateway, prenez note des indications suivantes :

1. Créez un récepteur HTTP/S sur le concentrateur afin qu'il reçoive les documents envoyés par WebSphere Process Server (s'il n'en existe pas déjà). Ce récepteur identifie l'adresse URL à laquelle le composant récepteur de WebSphere Partner Gateway récupère les documents envoyés par WebSphere Process Server.
2. Ajoutez l'indicateur `com.ibm.bcg.server.sync.SoapSyncHdlr` au paramètre de configuration `syncCheck` du récepteur HTTP (si vous envisagez de traiter des demandes bidirectionnelles).

### Configuration de WebSphere Process Server

Cette section indique comment configurer WebSphere Process Server pour qu'il puisse envoyer des messages à WebSphere Partner Gateway. Pour qu'un service de WebSphere Process Server envoie un message à WebSphere Partner Gateway via le protocole HTTP, vous devez utiliser un composant qui envoie une requête HTTP POST à l'URL à laquelle le récepteur HTTP de WebSphere Partner Gateway reçoit les documents. Chaque fois qu'un service de WebSphere Process Server envoie un document de gestion à un partenaire externe, il appelle ce composant.

#### Création d'un objet métier

Si vous utilisez la classe `BCGBackendIntegrationDataBindingUtil` pour convertir un document de gestion en objet métier, vous devez utiliser l'objet métier de niveau supérieur décrit dans la section «Objets métier enfant et de niveau supérieur», à la page 68. Vous pouvez créer l'objet métier à l'aide de l'éditeur d'objet métier ou importer le fichier `.xsd(BCG_TLO_BackendPackaging.xsd)` disponible sur le support d'installation.

Si vous n'utilisez pas l'emballage Backend Integration (en d'autres termes, si l'emballage spécifié est `None - Aucun`), vous devez créer l'objet métier en fonction des besoins pour l'échange de documents.

## Création du composant nécessaire à la conversion et à l'envoi du message

Créez un composant qui convertit l'objet métier en un flux d'entrée et envoie le message HTTP à WebSphere Partner Gateway. Ce composant sérialise l'objet métier provenant du service en un document de gestion. Il peut ensuite utiliser l'API Java `java.net.HttpURLConnection` pour effectuer une requête HTTP POST sur l'URL à laquelle le récepteur HTTP de WebSphere Partner Gateway reçoit les documents.

L'exemple de code suivant illustre la manière dont le composant peut transmettre un message à WebSphere Partner Gateway lorsque l'empaquetage NONE est spécifié.

```
String wpgHTTPTargetURL;    // URL de la cible HTTP de WebSphere Partner Gateway
byte payload[];           // message devant être envoyé à WebSphere Partner Gateway

// Définit wpgHTTPTargetURL
// Définit les octets de données utiles. Convertit en octets l'objet métier
// envoyé par le service.
...

try{

    java.net.HttpURLConnection uc = (java.net.HttpURLConnection)
        new URL(wpgHTTPTargetURL).openConnection();

    uc.setDoInput(true);
    uc.setDoOutput(true);
    uc.setRequestMethod("POST");

    // Définit le type de contenu à envoyer
    uc.setRequestProperty("Content-Type", "text/xml");

    uc.connect();
    BufferedOutputStream os = new BufferedOutputStream (uc.getOutputStream());
    os.write( payload);
    os.close();

    // Si vous attendez un document de gestion en réponse de WebSphere Partner
    // Gateway, vous devez lire la réponse
    InputStream is = uc.getInputStream();
    if(is != null){
        BufferedInputStream bis = new BufferedInputStream
(uc.getInputStream());
        // Lit les octets de BufferedInputStream obtenus ci-dessus
        // Convertit en objet métier le document de gestion en réponse envoyé
        // par WebSphere Partner Gateway
        ...
        bis.close();
    }
}
catch(Throwable ex){
    System.out.println("Impossible d'envoyer : " + ex.getMessage());
    ex.printStackTrace();
}
}
```

**Important :** Si vous utilisez l'empaquetage Backend Integration pour envoyer des documents à WebSphere Process Server, vous pouvez utiliser la Classe `BCGBackEndIntegrationDataBindingUtil` pour convertir des objets métier envoyés par le service (ou le composant) en documents de gestion attendus par WebSphere Partner Gateway. Si vous attendez de WebSphere Partner Gateway un document de gestion en réponse, vous pouvez également utiliser cette classe pour créer le message de réponse HTTP à partir de l'objet métier de type réponse renvoyé par votre service.

---

## Chapitre 5. Intégration de WebSphere Process Server avec le protocole de transport JMS

Ce chapitre décrit le processus d'intégration de WebSphere Partner Gateway à WebSphere Process Server via le protocole de transport JMS. Il présente les procédures de configuration de WebSphere Partner Gateway for JMS, lorsque WebSphere Platform Messaging (fournisseur de messagerie par défaut) et le bus d'intégration de service sont utilisés au titre de fournisseur JMS et que WebSphere Partner Gateway est installé sur le WebSphere Application Server.

Ce chapitre donne les informations suivantes quant à l'envoi et à la réception de documents entre WebSphere Partner Gateway et WebSphere Process Server via le protocole de transport JMS :

- «Envoi de documents via le protocole de transport JMS»
- «Réception de documents via le protocole de transport JMS», à la page 97
- «Configuration de l'environnement WebSphere Process Server sur WebSphere Application Server», à la page 98
- «Configuration de JMS avec WebSphere Partner Gateway installé sur WebSphere Application Server», à la page 101

---

### Envoi de documents via le protocole de transport JMS

Pour que WebSphere Partner Gateway envoie un document à WebSphere Process Server par le biais du protocole JMS, vous devez utiliser la liaison d'exportation JMS pour appeler le composant WebSphere Process Server via JMS. Les composants SCA (Service Component Architecture) peuvent recevoir des messages JMS provenant de la file d'attente JMS configurée dans leur liaison d'exportation JMS. figure 23, à la page 96 fournit une présentation générale de la manière dont WebSphere Partner Gateway envoie des documents à WebSphere Process Server via le protocole JMS.

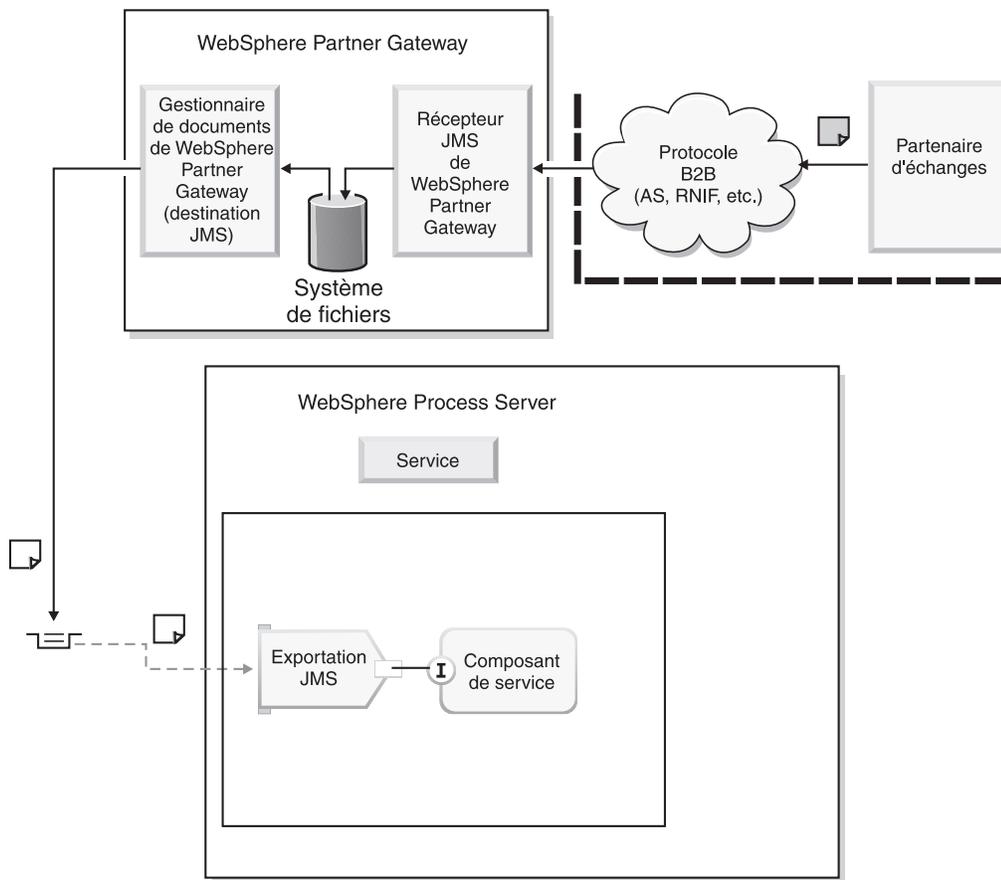


Figure 23. Exportation JMS d'appel des services SCA via JMS

Les étapes suivantes décrivent de quelle manière les liaisons d'exportation JMS SCA sont utilisées pour appeler les services SCA via JMS :

1. Le partenaire d'échange envoie un document de gestion à WebSphere Partner Gateway via un protocole business-to-business.
2. WebSphere Partner Gateway reçoit le document de gestion du partenaire d'échange.
3. Grâce à la connexion de partenaire configurée pour ce document de gestion particulier, WebSphere Partner Gateway route le document vers WebSphere Process Server.

Pour l'intégration basée sur JMS, la destination "Vers" destination de la connexion de partenaire est censée être une destination JMS.

Cette destination JMS est configurée pour l'envoi de messages JMS à une destination de bus SI (d'intégration de service).

La destination de bus SI est la file d'attente JMS configurée dans la liaison d'exportation JMS SCA du composant SCA.

## Réception de documents via le protocole de transport JMS

Pour que WebSphere Partner Gateway reçoive un document de WebSphere Process Server par le biais du protocole JMS, vous devez utiliser la liaison d'importation JMS pour appeler les services externes via JMS. Les composants SCA (Service Component Architecture) peuvent envoyer des messages JMS provenant dans la file d'attente JMS configurée dans leur liaison d'importation JMS. figure 24 fournit une présentation générale de la manière dont WebSphere Partner Gateway reçoit des documents de WebSphere Process Server via le protocole JMS.

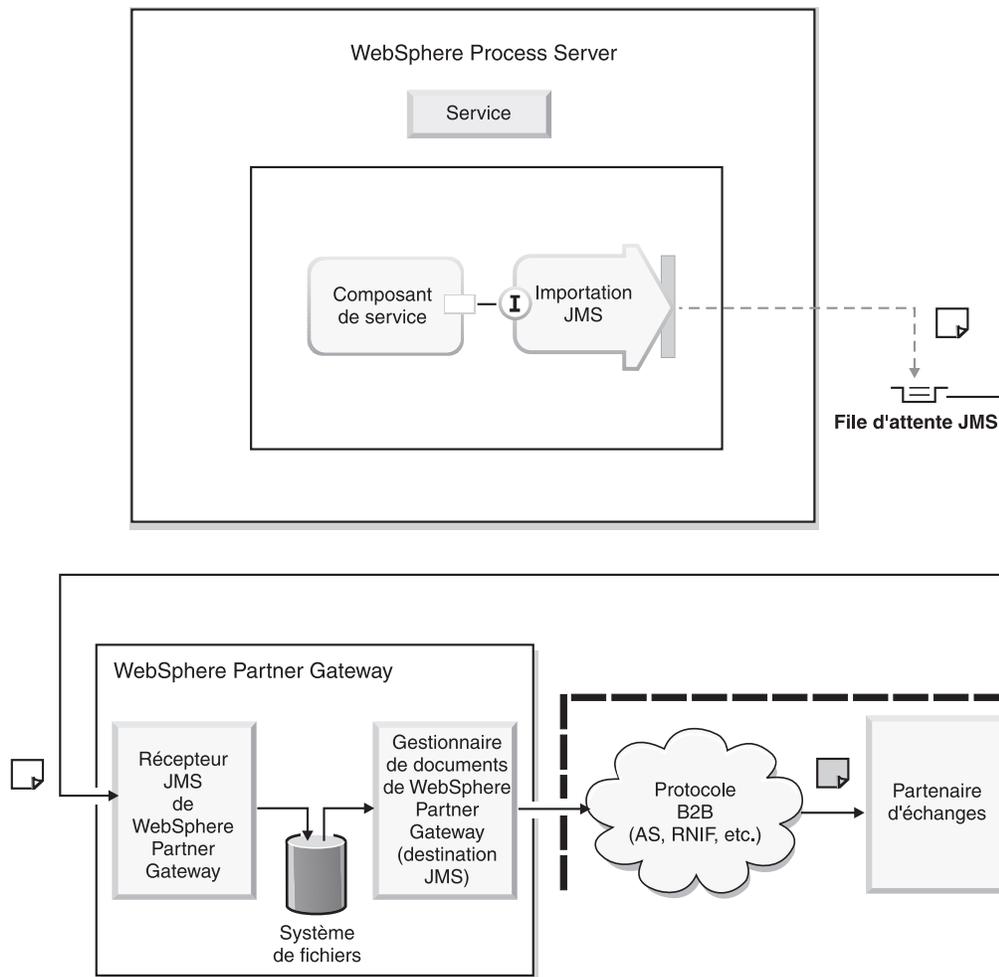


Figure 24. Importation JMS de réception de documents via JMS

Les étapes suivantes décrivent de quelle manière les liaisons d'importation JMS sont utilisées pour recevoir des documents de WebSphere Process Server :

1. Le Récepteur JMS de WebSphere Partner Gateway reçoit un document de gestion de WebSphere Process Server.
2. Grâce à la connexion de partenaire configurée pour ce document de gestion particulier, WebSphere Partner Gateway route le document vers le partenaire d'échange.
3. Le partenaire d'échange reçoit le document de gestion de WebSphere Partner Gateway via un protocole business-to-business mutuellement convenu.

Pour l'intégration basée sur JMS, la file d'attente JMS configurée dans le récepteur JMS est censée être la destination JMS sur laquelle les services de WebSphere Process Server Services enverront les messages JMS.

Cette destination de file d'attente JMS est la destination de file d'attente configurée dans la liaison d'importation JMS SCA du composant SCA.

---

## Configuration de l'environnement WebSphere Process Server sur WebSphere Application Server

Cette section présente les étapes de configuration de l'environnement WebSphere Process Server pour le transport JMS sur WebSphere Application Server. Elles impliquent la création et la configuration des artefacts de WebSphere Process Server.

Cette section contient les informations suivantes :

- «Création d'un service SCA avec WSDL»
- «Personnalisation des liaisons d'importation et d'exportation JMS»
- «Implémentation de la liaison de données JMS», à la page 99
- «Personnalisation d'un sélecteur de fonctions», à la page 100

### Création d'un service SCA avec WSDL

Un service SCA, associé à des liaisons d'exportation et d'importation, sert à tirer profit des fonctions business-to-business de WebSphere Partner Gateway et les fonctions d'interaction du partenaire d'échange. Pour les instructions d'utilisation du WSDL (Web Services Description Language) pour créer et définir des services SCA, consultez le centre de documentation de WebSphere Process Server.

### Personnalisation des liaisons d'importation et d'exportation JMS

La liaison d'importation JMS d'un composant sert à appeler des services externes via JMS. Les composants SCA peuvent envoyer des messages JMS dans la file d'attente JMS configurée dans leur liaison d'importation JMS. Pour permettre aux composants SCA d'envoyer des messages dans la file d'attente configurée dans le Récepteur JMS de WebSphere Partner Gateway :

1. Fournissez la liaison de données JMS requise par l'importation JMS SCA du composant. Reportez-vous à «Implémentation de la liaison de données JMS», à la page 99.
2. Indiquez les attributs de la liaison d'importation JMS propre à votre environnement. L'importation JMS d'un composant place les messages dans la destination JMS spécifiée dans la liaison d'importation JMS. La destination JMS que vous indiquerez dans la liaison d'importation JMS doit faire référence à la file d'attente JMS configurée dans le Récepteur JMS de WebSphere Partner Gateway. Pour plus de détails sur les attributs de la liaison d'importation JMS, consultez le centre de documentation de WebSphere Process Server.

La liaison d'exportation JMS d'un composant sert à appeler le composant WebSphere Process Server via JMS. Pour permettre aux composants SCA de recevoir des messages JMS depuis la destination JMS configurée dans leur liaison d'exportation JMS :

1. Fournissez la liaison de données JMS requise par la liaison d'exportation JMS SCA du composant. Reportez-vous à «Implémentation de la liaison de données JMS».
2. Fournissez le sélecteur de fonctions requis par la liaison d'exportation JMS SCA du composant. Reportez-vous à «Personnalisation d'un sélecteur de fonctions», à la page 100.
3. Indiquez les attributs de la liaison d'exportation JMS propre à votre environnement. L'exportation JMS d'un composant extrait le message JMS de la destination JMS spécifiée dans la liaison d'exportation JMS. La destination JMS que vous indiquerez dans la liaison d'exportation JMS doit faire référence à la file d'attente JMS configurée dans la Destination JMS de WebSphere Partner Gateway. Pour plus de détails sur les attributs de la liaison d'importation JMS, consultez le centre de documentation de WebSphere Process Server.

## Implémentation de la liaison de données JMS

Les composants d'importation et d'exportation JMS SCA de WebSphere Process Server donnent des informations de configuration pour la spécification de la liaison de données JMS. La liaison de données JMS sert dans l'importation/exportation JMS SCA à convertir des objets métier en messages JMS et inversement :

- Exportation JMS SCA : l'exportation JMS SCA d'un service exploite une liaison de données JMS configurée pour convertir un message JMS reçu de WebSphere Partner Gateway en objet métier, comme prévu dans la méthode du service. Cet objet métier sert à appeler la méthode d'un service.
- Importation JMS SCA : l'importation JMS SCA d'un service exploite une liaison de données JMS configurée pour convertir un objet métier en message JMS, comme prévu par WebSphere Partner Gateway.

Pour toute question concernant le développement et l'implémentation de la liaison de données JMS, consultez le centre de documentation de WebSphere Process Server.

Pour les intégrations dorsales effectuées via JMS, WebSphere Partner Gateway prend en charge les empaquetages de valeur AUCUN et Backend Integration.

Si c'est l'empaquetage AUCUN qui est utilisé, vous pouvez exploiter la liaison de données JMS par défaut fournie par WebSphere Process Server ou implémenter votre propre liaison de données. De manière générale, vous devrez écrire votre propre liaison de données si le format des messages JMS échangés entre WebSphere Partner Gateway et WebSphere Process Server est autre que XML.

Si c'est l'empaquetage Backend Integration qui est utilisé, vous pouvez exploiter la liaison de données

`com.ibm.bcg.dataBinding.BCGBackendIntegrationJMSDataBindingImpl` fournie par WebSphere Partner Gateway. Pour personnaliser le traitement, vous pouvez implémenter la liaison de données en vous servant de `com.ibm.bcg.dataBinding.BCGBackendIntegrationDataBindingUtil` ou `com.ibm.bcg.dataBinding.BCGBackendIntegrationJMSDataBindingImpl`, qui sont fournies dans WebSphere Partner Gateway. Pour plus d'informations sur ces classes, merci de consulter «Classe `BCGBackendIntegrationJMSDataBindingImpl`», à la page 84 et «Classe `BCGBackendIntegrationDataBindingUtil`», à la page 78.

Pour implémenter l'interface de la liaison de données JMS pour l'empaquetage Backend Integration, créez des méthodes `read` et `write` propres à l'utilisateur.

## Personnalisation d'un sélecteur de fonctions

Le sélecteur de fonctions est obligatoire pour la liaison d'exportation JMS SCA. Il s'emploie pour déterminer quelle méthode de service sera appelée pour un document de gestion reçu dans la destination JMS.

Implémentez un sélecteur de fonctions au moyen de l'une des techniques suivantes :

- Empaquetage de valeur AUCUN : si WebSphere Partner Gateway envoie des messages JMS à WebSphere Process Server avec l'empaquetage de valeur AUCUN, l'utilisateur doit déterminer quelle méthode de leur service SCA sera appelée. Avec l'empaquetage de valeur AUCUN, l'unique façon de déterminer le type d'un document de gestion est de l'analyser. Pour ce faire, vous pouvez utiliser l'une des deux techniques suivantes :
  - Le répartiteur : pour l'empaquetage de valeur AUCUN, IBM recommande de ne pas analyser le document à l'aide du sélecteur de fonctions. Ce serait redondant et coûterait un temps précieux puisque les documents nécessiteraient également une analyse dans la liaison de données. IBM vous recommande donc de développer un composant répartiteur ou sélecteur de fonctions qui serait le frontal (l'interface de façade) vers le véritable composant WebSphere Process Server intéressé par le document de gestion. La liaison d'exportation JMS de ce composant répartiteur recevra le document de gestion envoyé par WebSphere Partner Gateway. Ce composant répartiteur détiendra une méthode sur laquelle il pourra recevoir tous les documents de gestion possibles. La liaison d'exportation JMS déconvertira les messages JMS messages en objets métier. Le module répartiteur déterminera alors quel composant du service est intéressé par cet objet métier et appellera le composant avec cet objet métier.
  - Analyse minimale : si une liaison d'exportation JMS SCA possède plusieurs liaisons de méthode, le sélecteur de fonctions peut être implémenté pour effectuer une analyse minimale du document de gestion en vue de déterminer son type (par exemple : bon de commande d'achat, confirmation de commande d'achat, facture, etc.). Le sélecteur de fonctions peut alors renvoyer le type du document de gestion en tant que nom de méthode native. Au moment de la configuration, le nom de méthode native dans la liaison de méthode de liaison d'exportation JMS SCA peut être spécifié en tant que type de document de gestion. Grâce à la liaison de méthode, l'exportation JMS SCA peut résoudre quelle méthode du service il faut appeler.

**Remarque :** La liaison de données réalise une analyse complète du document de gestion pour construire l'objet métier. C'est pourquoi le sélecteur de fonctions doit lui procéder à une analyse restreinte du document de gestion. Si l'analyse via le sélecteur de fonctions s'avère trop coûteuse en temps, il n'est pas nécessaire d'utiliser cette technique.

- Sélecteur de fonctions banal : si l'utilisateur ne prévoit qu'un seul type de document de gestion, vous pouvez développer un sélecteur de fonctions banal qui n'analysera pas le document et qui renverra chaque fois le même nom de méthode native. Au moment de la configuration, le nom de méthode native dans la liaison de méthode de liaison d'exportation JMS SCA peut être spécifié comme étant ce nom de méthode. Grâce à la liaison de méthode, l'exportation JMS SCA pourra déterminer quelle méthode du service il faut appeler. L'avantage de cette technique est qu'aucune analyse du document de gestion entrant n'est requise. Toutefois, sa limitation est qu'il ne peut y avoir qu'un seul type de document de gestion dans la file d'attente JMS.

- **Empaquetage Backend Integration** : si WebSphere Partner Gateway envoie des messages JMS à WebSphere Process Server avec l’empaquetage Backend Integration, l’utilisateur peut exploiter les en-têtes JMS de cet empaquetage pour développer un sélecteur de fonctions. Les propriétés JMS pourront être lues par le sélecteur de fonctions dans le message JMS en entrée. Le sélecteur de fonctions peut utiliser la valeur de l’un des en-têtes JMS d’empaquetage Backend Integration pour constituer un nom de méthode native. Exemple : pour un message RosettaNet, la propriété JMS `x-aux-payload-root-tag` peut être lue pour déterminer le type du document de gestion. Le sélecteur de fonctions peut alors constituer le nom de méthode native de façon à ce qu’il corresponde à celui de la liaison de méthode de l’exportation JMS.

A l’aide de WebSphere Integration Developer, créez un sélecteur de fonctions personnalisé en implémentant l’interface `commonj.connector.runtime.FunctionSelector`. Par exemple :

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

---

## Configuration de JMS avec WebSphere Partner Gateway installé sur WebSphere Application Server

Cette section présente les étapes de configuration de l’environnement WebSphere Partner Gateway pour le transport JMS sur WebSphere Application Server lorsque l’on utilise WebSphere Platform Messaging ne fournisseur de messagerie. On part ici du principe que WebSphere Partner Gateway est installé sur WebSphere Application Server.

Il est supposé que l’utilisateur est familier de WebSphere Platform Messaging et des bus d’intégration de service.

Cette section contient les informations suivantes :

- «Création et configuration des bus, files d’attente JMS et fabriques de connexions», à la page 102
- «Création du récepteur JMS», à la page 105
- «Création de la destination JMS», à la page 105
- «Création d’une file d’attente de destination», à la page 106

L’utilisation du moteur de messagerie (ME) et de la JNDI de serveur d’applications du composant WebSphere Partner Gateway peut limiter les topologies de bus de vos clients. C’est pourquoi IBM recommande de connecter les composants WebSphere Partner Gateway (le récepteur et la destination JMS) au ME sur un autre serveur d’applications WebSphere (WAS), exécutant ou non WebSphere Process Server. Pour une présentation générale de cette topologie, voir figure 25, à la page 102.

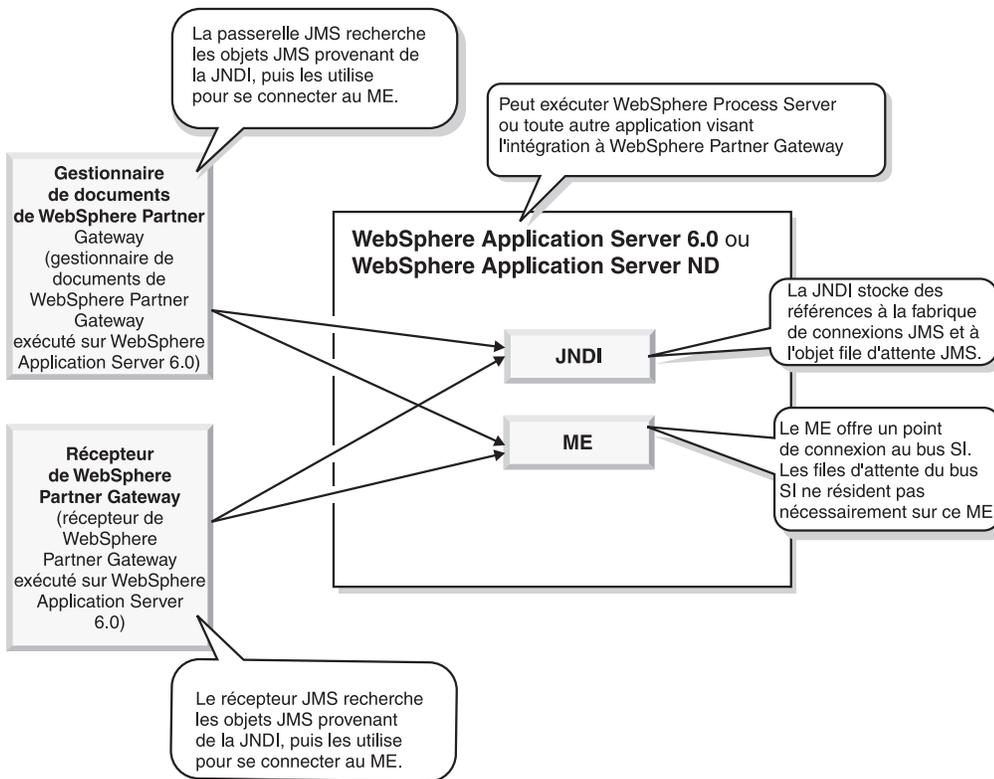


Figure 25. Accès au ME et à la JNDI à partir d'une autre instance de WebSphere Application Server

## Création et configuration des bus, files d'attente JMS et fabriques de connexions

Cette section explique comment créer et configurer des bus, files d'attente JMS et fabriques de connexions utilisables par WebSphere Partner Gateway pour envoyer et recevoir des messages.

### Création d'un bus d'intégration de service

Un bus d'intégration de service prend en charge des applications au moyen d'architectures basées sur les messages et orientées services. Un bus est un groupe de serveurs et de clusters interconnectés ayant été ajoutés comme membres du bus. Les applications se connectent à un bus au niveau de l'un des moteurs de messagerie (ME) associés aux membres de ce bus.

A l'aide de la console d'administration WebSphere :

1. Créez un bus auquel vous donnerez un nom.
  - a. Cliquez sur **Intégration de service > Bus**.
  - b. Cliquez sur **Nouveau** et donnez un nom au bus. Par exemple, BUSIS.
  - c. Cliquez sur **Appliquer**.
  - d. Sauvegardez la configuration en cliquant sur **Sauvegarder** dans la fenêtre Messages qui apparaît alors. Ce message est destiné à confirmer que vous souhaitez appliquer les modifications effectuées à la configuration principale.
  - e. Cliquez à nouveau sur **Sauvegarder** lorsqu'il vous sera demandé de mettre à jour le référentiel principal avec vos modifications.

2. Ajoutez des membres au bus.
  - a. Cliquez sur le nom du bus nouvellement créé.
  - b. Dans la sous-fenêtre Propriétés supplémentaires, cliquez sur **Membres du bus**.
  - c. Cliquez sur **Ajouter** et sélectionnez le serveur ou le cluster à ajouter.
  - d. Cliquez sur **Suivant**, puis sur **Terminer** pour confirmer l'ajout des nouveaux membres au bus.
  - e. Sauvegardez la configuration en cliquant sur **Sauvegarder** dans la fenêtre Messages qui apparaît alors. Ce message est destiné à confirmer que vous souhaitez appliquer les modifications effectuées à la configuration principale.
  - f. Cliquez à nouveau sur **Sauvegarder** lorsqu'il vous sera demandé de mettre à jour le référentiel principal avec vos modifications.
3. Créez une file d'attente de destination :
  - a. Dans la console par défaut de WebSphere Application Server, cliquez sur **Intégration système** dans le volet gauche.
  - b. Cliquez sur **Bus < BUSIS** (ou le nom du bus créé à l'étape 1).
  - c. Dans la sous-fenêtre Propriétés supplémentaires, cliquez sur **Destinations**. Cliquez sur **Nouveau**.
  - d. Sélectionnez avec votre pointeur le bouton d'options **File d'attente** pour définir le type de destination, puis cliquez sur **Suivant**.
  - e. Entrez un **Identificateur**. Par exemple, Demande. De cette manière, la file d'attente de destination sera créée sur le bus.
4. Sauvegardez la configuration en cliquant sur **Sauvegarder** dans la fenêtre Messages qui apparaît alors. Ce message est destiné à confirmer que vous souhaitez appliquer les modifications effectuées à la configuration principale.
5. Cliquez à nouveau sur **Sauvegarder** lorsqu'il vous sera demandé de mettre à jour le référentiel principal avec vos modifications.

## Création d'une fabrique de connexions pour la file d'attente JMS

Une fabrique de connexions de file d'attente JMS sert à créer des connexions avec le fournisseur JMS associé des files d'attente JMS, dans le but de la messagerie point-à-point.

A l'aide de la console d'administration WebSphere :

1. Créez une fabrique de connexions de file d'attente en renseignant les zones **Nom** et **Nom JNDI** en utilisant la syntaxe suivante :
  - **Nom** : SIBUS.JMSTargetQCF
  - **Nom JNDI**: BUSIS/JMSTargetQCFOù BUSIS est le nom du bus créé lors des précédentes étapes.
2. Sélectionnez le bus. Par exemple, **BUSIS**.
3. Dans la fenêtre résultante, cliquez sur la fabrique de connexions de file d'attente que vous venez de créer et saisissez les **Noeuds finals du fournisseur** de la manière suivante :

```
IPAddress/Name:7276:BootstrapBasicMessaging
```

Où IPAddress est l'adresse IP ou le nom de la machine sur laquelle s'exécute WebSphere Application Server. Le moteur de messagerie de ce bus d'intégration de service est censé s'exécuter sur cette machine. 7276 est le numéro de port spécifié comme SIB\_ENDPOINT\_ADDRESS pour l'instance WebSphere. Si votre moteur de messagerie s'exécute sur le système avec l'adresse IP 9.26.234.100 et que la valeur de SIB\_ENDPOINT\_ADDRESS pour l'instance WebSphere s'exécutant sur ce serveur est 7276, vous indiquerez les noeuds finals du fournisseur comme suit :

```
9.26.234.100:7276:BootstrapBasicMessaging
```
4. Sauvegardez la configuration en cliquant sur **Sauvegarder** dans la fenêtre Messages qui apparaît alors. Ce message est destiné à confirmer que vous souhaitez appliquer les modifications effectuées à la configuration principale.
5. Cliquez à nouveau sur **Sauvegarder** lorsqu'il vous sera demandé de mettre à jour le référentiel principal avec vos modifications.

## Création d'une file d'attente JMS

Une file d'attente JMS sert de destination pour la messagerie point-à-point.

A l'aide de la console d'administration WebSphere :

1. Ouvrez le menu Ressources et cliquez sur **Fournisseurs JMS > Messagerie par défaut** .
2. Cliquez sur **Files d'attente JMS** dans la section Destinations de la page résultante.
3. Cliquez sur **Nouveau**.
4. Saisissez un nom de file d'attente dans les zones **Nom** et **Nom JNDI** en utilisant la syntaxe suivante :
  - **Nom** : Request.JMSTarget
  - **Nom JNDI**: Request/JMSTarget
5. Sélectionnez le **Nom du bus** (par exemple, BUSIS) et le **Nom de la file d'attente** dans les listes déroulantes.
6. Cliquez sur **OK**.
7. Sauvegardez la configuration en cliquant sur **Sauvegarder** dans la fenêtre Messages qui apparaît alors. Ce message est destiné à confirmer que vous souhaitez appliquer les modifications effectuées à la configuration principale.

8. Cliquez à nouveau sur **Sauvegarder** lorsqu'il vous sera demandé de mettre à jour le référentiel principal avec vos modifications.

## Création du récepteur JMS

Cette section donne des consignes sur la création d'un récepteur dans WebSphere Partner Gateway.

A l'aide de la console de communauté de WebSphere Partner Gateway :

1. Pour afficher la liste des récepteurs, cliquez sur **Administrateur du concentrateur > Configuration du concentrateur > Récepteurs**.
2. Dans la page Liste des récepteurs, cliquez sur **Créer le récepteur**.
3. Dans la section Caractéristiques du récepteur, effectuez les étapes suivantes :
  - a. Attribuez un nom au récepteur. Vous pourriez appeler par exemple le récepteur JMSTarget. Cette zone est obligatoire. Le nom que vous entrez ici s'affichera dans la liste des récepteurs.
  - b. Indiquez éventuellement le statut du récepteur. Le statut par défaut est Activé. Un récepteur activé est prêt à l'acceptation des documents. Un récepteur désactivé ne peut accepter des documents.
  - c. Entrez éventuellement une description du récepteur.
4. Sélectionnez **JMS** dans la liste Transport.
5. Indiquez l'URL du fournisseur JMS. Par exemple,  
`iiop://systemip:2809/`  
Où systemip est l'adresse IP du système sur lequel s'exécute WebSphere Platform Messaging et 2809, le numéro de port par défaut sur lequel le serveur BOOTSTRAP s'exécute.
6. Renseignez la zone Nom de file d'attente JMS. Cette zone est obligatoire. Ce nom doit correspondre au nom JNDI de la file d'attente JMS créé dans la section précédente de ce chapitre. Par exemple, Request/JMSTarget.
7. Précisez le nom de la fabrique JMS. Cette zone est obligatoire. Ce nom doit correspondre au nom de fabrique de connexions de file d'attente créé plus tôt. Par exemple, BUSIS/JMSTargetQCF.
8. Entrez le nom de fabrique JNDI comme suit :  
`com.ibm.websphere.naming.WsnInitialContextFactory`
9. Cliquez sur **Sauvegarder**.

## Création de la destination JMS

Cette section donne des consignes sur la création d'une destination dans WebSphere Partner Gateway.

A l'aide de la console de communauté de WebSphere Partner Gateway, créez une destination JMS :

1. Cliquez sur **Administrateur du compte > Profils > Partenaire**.
2. Entrez les critères de recherche et cliquez sur **Rechercher**, ou cliquez sur **Rechercher** sans entrer de critère de recherche pour afficher la liste de tous les partenaires.
3. Cliquez sur l'icône Afficher les détails pour faire apparaître le profil du partenaire.
4. Cliquez sur **Destinations**.
5. Cliquez sur **Créer**.

6. Dans la page Listes des destinations, entrez un nom pour identifier la destination. Cette zone est obligatoire.
7. Sélectionnez **JMS** dans la liste **Transport**.
8. Dans la zone **Adresse**, entrez l'URI correspondant à l'emplacement de distribution du document. Par exemple :  
`iiop//systemip:2809`  
 Où systemip est l'adresse IP du système sur lequel s'exécute WebSphere Platform Messaging.
9. Dans la zone **Nom de la fabrique JMS**, indiquez le nom de la classe Java qu'utilise le fournisseur JMS pour se connecter à la file d'attente JMS. Il s'agit du nom JNDI de la fabrique JMS créée précédemment. Par exemple :  
`BUSIS/JMSTargetQCF`
10. Dans la zone **Nom de file d'attente JMS**, entrez le nom de la file d'attente vers laquelle les documents doivent être envoyés. Il s'agit du nom JNDI de la file d'attente dans laquelle les messages doivent être entreposés. Par exemple :  
`Response/JMSTarget`
11. Dans la zone **Nom JNDI de la fabrique JMS**, entrez la fabrique suivante :  
`com.ibm.websphere.naming.WsnInitialContextFactory`
12. Cliquez sur **Sauvegarder**.

## Création d'une file d'attente de destination

A l'aide de la console d'administration WebSphere, créez une file d'attente de destination sur WebSphere Application Server :

1. Cliquez sur **Intégration système** sur le volet gauche.
2. Cliquez sur **Bus < BUSIS** (ou le nom du bus créé dans la précédente section).
3. Dans la sous-fenêtre Propriétés supplémentaires, cliquez sur **Destinations**. Cliquez sur **Nouveau**.
4. Sélectionnez avec votre pointeur le bouton d'options **File d'attente** pour définir le type de destination, puis cliquez sur **Suivant**.
5. Entrez un **Identificateur**. Par exemple, Response. De cette manière, la file d'attente de destination sera créée sur le bus.

---

## Chapitre 6. Intégration de WebSphere Process Server à SOAP/HTTP

Ce chapitre décrit le processus d'intégration de WebSphere Partner Gateway à WebSphere Process Server à l'aide de SOAP via protocole de transport HTTP. Il contient les informations suivantes :

- «Description de l'échange des messages SOAP via le protocole HTTP»
- «Appel de services Web hébergés par WebSphere Process Server», à la page 110
- «Appel de services Web hébergés par les partenaires externes», à la page 110

---

### Description de l'échange des messages SOAP via le protocole HTTP

Lorsqu'un partenaire externe sollicite un service Web auprès de WebSphere Process Server ou que WebSphere Process Server sollicite un service Web auprès d'un partenaire externe, le message SOAP associé est envoyé à un récepteur sur le concentrateur WebSphere Partner Gateway. WebSphere Partner Gateway fait office de proxy (serveur mandataire) en transmettant la demande à l'URL d'emplacement du service.

## De quelle manière les services Web hébergés par WebSphere Process Server sont-ils appelés

Lorsqu'un partenaire externe appelle un service Web hébergé sur WebSphere Process Server, l'enchaînement suivant se déroule :

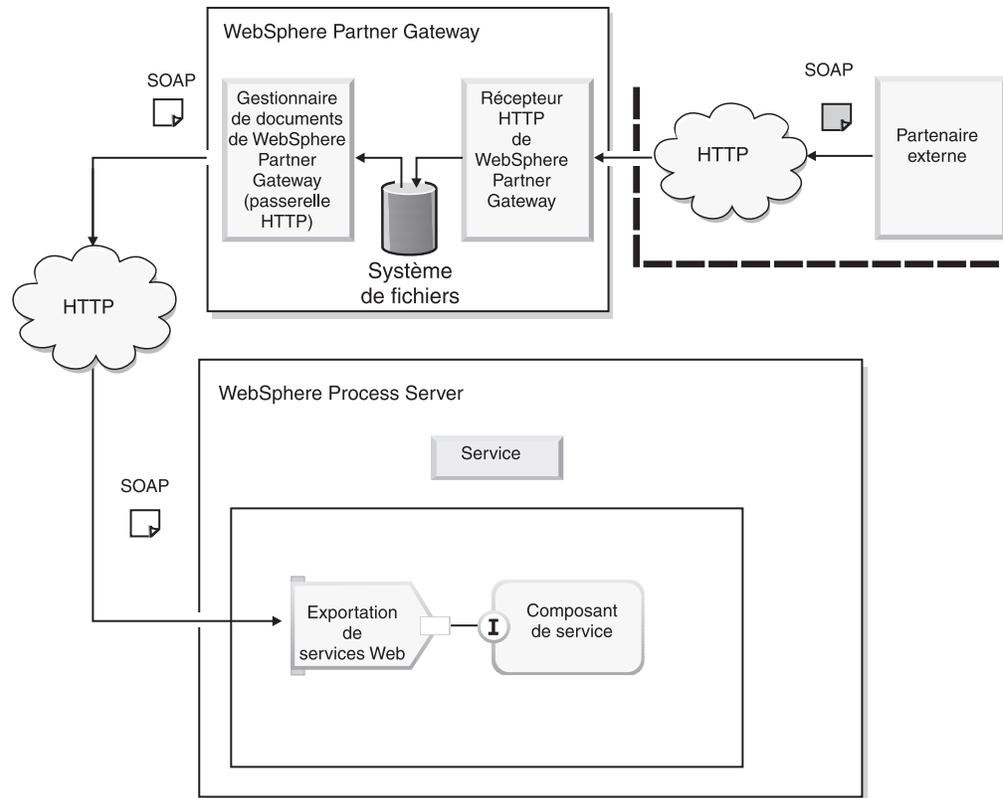


Figure 26. Partenaire externe appelant un service Web

WebSphere Partner Gateway route la demande de service Web faite à WebSphere Process Server, sur laquelle la liaison d'exportation de service Web est utilisée pour appeler le service.

Vous indiquez à votre partenaire externe le WSDL public, qui spécifie l'URL de WebSphere Partner Gateway comme noeud final. Voir «Configuration de WebSphere Process Server», à la page 111 pour plus d'informations sur la spécification de cette URL. WebSphere Partner Gateway agit en tant que proxy. Il reçoit un message SOAP en provenance du partenaire et détermine quel est le service Web correspondant. Il appelle ensuite le service Web privé sur WebSphere Process Server au moyen du même message SOAP. La réponse HTTP renvoyée par WebSphere Process Server (une réponse contenant des informations de transport et le service Web) est alors renvoyée au partenaire

## De quelle manière les services Web hébergés par les partenaires sont-ils appelés

Lorsque WebSphere Process Server appelle un service Web auprès d'un partenaire externe, l'enchaînement suivant se déroule :

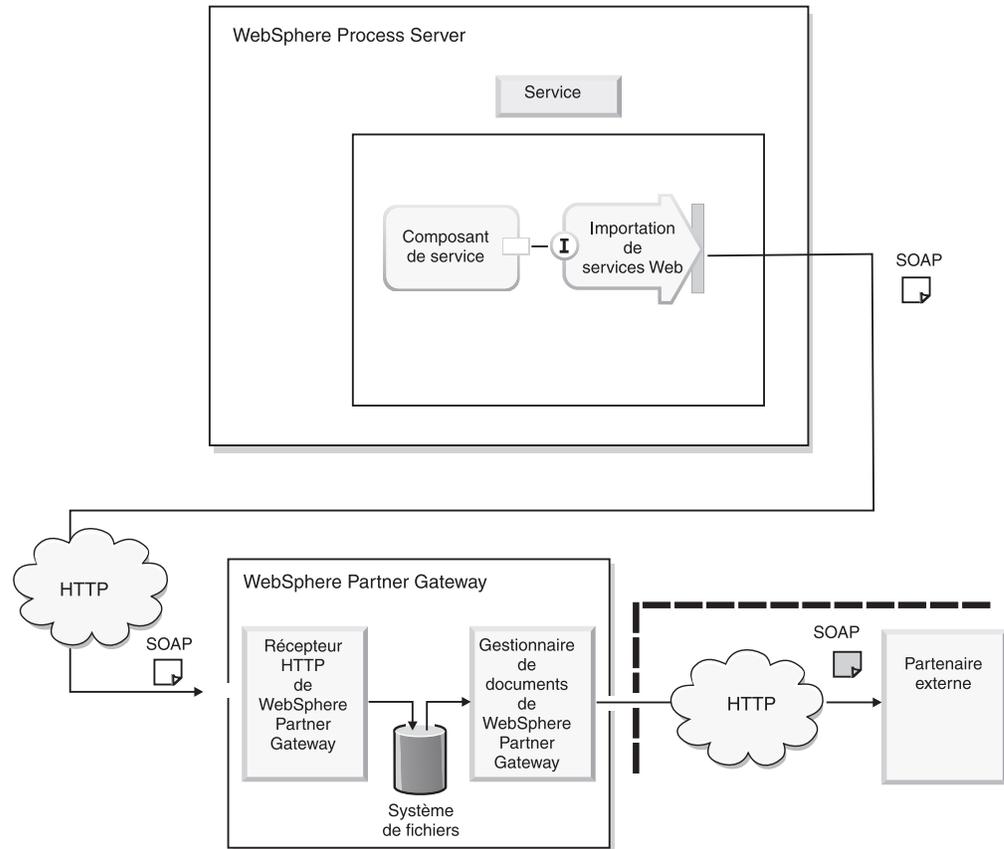


Figure 27. Service de WebSphere Process Server appelant un service Web

WebSphere Process Server exploite sa liaison d'importation de services Web pour appeler le service Web. WebSphere Partner Gateway route la demande de service Web issue de WebSphere Process Server vers le partenaire externe approprié.

WebSphere Partner Gateway met le service Web à la disposition de WebSphere Process Server au niveau de l'URL du service Web spécifié dans la console de communauté pour le chargement du service Web. En outre, le service de WebSphere Process Server doit fournir le paramètre de l'URL pour identifier "Vers le partenaire". Pour plus d'informations, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*. WebSphere Partner Gateway agit en tant que proxy. Il reçoit un message SOAP en provenance de WebSphere Process Server et détermine quel est le service Web privé correspondant, ainsi que la valeur de "Vers le partenaire". Il appelle ensuite le service Web fourni par le partenaire externe au moyen du même message SOAP. La réponse HTTP renvoyée par le partenaire externe (une réponse contenant des informations de transport et le service Web) est alors renvoyée à WebSphere Process Server.

La réponse reçue du partenaire externe est renvoyée à WebSphere Process Server via la même connexion HTTP que celle qui a servi à la demande. La procédure est la même pour les services Web en demande uniquement et ceux en demande-réponse.

---

## Appel de services Web hébergés par WebSphere Process Server

Cette section décrit comment un partenaire externe appelle un service Web hébergé sur WebSphere Process Server.

Un partenaire externe envoie la demande à un récepteur HTTP situé sur le concentrateur WebSphere Partner Gateway. WebSphere Partner Gateway authentifie l'utilisateur, consulte la connexion entre le partenaire externe et le partenaire interne et transfère la demande à une destination définie au niveau du partenaire interne.

**Remarque :** Lorsque le partenaire externe envoie le message SOAP, il doit s'authentifier auprès de WebSphere Partner Gateway. Il peut pour cela utiliser l'authentification HTTP standard, en renseignant l'ID entreprise du partenaire, le nom de console et le mot de passe de console. Autre solution, le partenaire peut présenter un certificat client SSL, déjà configuré dans WebSphere Partner Gateway . Pour plus d'informations sur les certificats, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

### Configuration de WebSphere Partner Gateway

Vous devez configurer WebSphere Partner Gateway selon la procédure décrite dans le *Guide de configuration du concentrateur WebSphere Partner Gateway*. Lors de la configuration de WebSphere Partner Gateway, prenez note des indications suivantes :

1. Ajoutez le descripteur `com.ibm.bcg.server.sync.SoapSyncHdlr` au paramètre de configuration `syncCheck` du récepteur HTTP (si vous envisagez de traiter des demandes bidirectionnelles).
2. A partir de WebSphere Process Server, obtenez le WSDL généré par la liaison d'exportation de service Web de votre composant.
3. Assurez-vous que la destination HTTP du partenaire interne indique l'URL qu'utilisera WebSphere Partner Gateway pour envoyer des messages à WebSphere Process Server. Cette destination pointe sur l'EndPointURL (URL de noeud final) spécifié dans le WSDL que vous avez chargé à l'étape 2.

### Configuration de WebSphere Process Server

Configurez les composants de WebSphere Process Server en suivant la documentation publiée dans le centre de documentation de WebSphere Process Server. Lorsque vous créez une exportation pour le composant, veillez à sélectionner **Liaison de service Web**.

---

## Appel de services Web hébergés par les partenaires externes

Lorsqu'un service de WebSphere Process Server fait une demande SOAP à un URL de WebSphere Partner Gateway, celui-ci identifie le partenaire apte à traiter la demande SOAP. Il utilise l'authentification standard fournie par WebSphere Process Server en fonction du profil du partenaire. Si la demande SOAP est bidirectionnelle, WebSphere Partner Gateway obtient la réponse du partenaire externe et la renvoie à WebSphere Process Server.

## Configuration de WebSphere Partner Gateway

Vous devez configurer WebSphere Partner Gateway selon la procédure décrite dans le *Guide de configuration du concentrateur WebSphere Partner Gateway*. Lors de la configuration de WebSphere Partner Gateway, prenez note des indications suivantes :

1. Ajoutez le descripteur `com.ibm.bcg.server.sync.SoapSyncHdlr` au paramètre de configuration `syncCheck` du récepteur HTTP (si vous envisagez de traiter des demandes bidirectionnelles).
2. Assurez-vous que la destination HTTP du partenaire interne indique l'URL qu'utilisera WebSphere Partner Gateway pour envoyer des messages. Cette destination pointe sur l'EndPointURL (URL de noeud final) spécifié dans le WSDL que vous avez chargé au cours du processus de configuration.

## Configuration de WebSphere Process Server

Configurez les composants de WebSphere Process Server en suivant la documentation publiée dans le centre de documentation de WebSphere Process Server. Notez bien que vous devez obtenir le fichier WSDL décrivant le service Web auprès du prestataire de ce service. Ce fichier WSDL doit être importé dans le module pour les composants.

WebSphere Partner Gateway met en général à disposition sur son URL les services Web fournis par le partenaire externe. C'est pourquoi WebSphere Process Server doit appeler le service sur l'URL du récepteur HTTP de WebSphere Partner Gateway. WebSphere Partner Gateway requiert également que le partenaire interne renseigne un nom d'utilisateur et un mot de passe pour l'authentification HTTP standard. Pour plus d'informations sur le nom d'utilisateur et le mot de passe, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*. Pour permettre à WebSphere Process Server d'offrir l'authentification HTTP standard et de modifier le noeud final du service Web qui sera appelé par WebSphere Process Server, procédez comme suit :

Pour modifier l'adresse du noeud final :

1. Connectez-vous à la console d'administration de WebSphere Process Server
2. Naviguez jusqu'aux liaisons client de service Web du service que vous appelez auprès de WebSphere Partner Gateway. Vous pouvez les localiser en suivant le chemin : **Applications d'entreprise** > <votre\_application> > **Modules EJB** > <votre\_fichier\_JAR> > **Liaisons du client service Web**
3. Indiquez l'**URL de noeud final de substitution** du port de service que vous souhaitez appeler auprès de WebSphere Partner Gateway. Indiquez l'URL du récepteur HTTP de WebSphere Partner Gateway qui recevra la demande de service Web issue de WebSphere Process Server. L'URL du noeud final a la forme suivante :  
`<adresse_IP:numéro_de_port>bcgreceiver/<nom_de_la_cible>`
4. Cliquez sur **Appliquer**, puis sur **OK**.

Pour préciser le nom d'utilisateur et le mot de passe requis pour l'authentification standard, procédez comme suit :

1. Connectez-vous à la console d'administration de WebSphere Process Server
2. Naviguez jusqu'aux liaisons client de service Web du service que vous appelez auprès de WebSphere Partner Gateway. Vous pouvez les localiser en suivant le chemin : **Applications d'entreprise** > <votre\_application> > **Modules EJB** > <votre\_fichier\_JAR> > **Liaisons de sécurité du client de services Web** > **Authentification HTTP standard**

3. Indiquez l'ID et le mot de passe pour l'authentification standard.

L'ID d'authentification a la forme suivante :

*IDentreprise/nomutilisateur*

Par exemple, si l'ID entreprise du partenaire interne est 987654321 et que son nom d'utilisateur est admin, son ID d'authentification sera :

987654321/admin

4. Cliquez sur **Appliquer**, puis sur **OK**.

---

## Chapitre 7. Intégration de WebSphere Process Server avec le système de fichiers comme protocole de transport

Ce chapitre décrit le processus d'intégration de WebSphere Partner Gateway à WebSphere Process Server via le protocole système de fichiers.

Le protocole système de fichiers permet à WebSphere Partner Gateway d'envoyer des messages en les plaçant dans une arborescence de répertoires définie. WebSphere Partner Gateway reçoit également les messages en les lisant à partir de l'arborescence de répertoires. Le protocole système de fichiers ne prend en charge que l'empaquetage None (Aucun).

Ce chapitre contient les informations suivantes :

- «Envoi de documents via le protocole système de fichiers»
- «Réception de documents via le protocole système de fichiers»
- «Configuration de l'environnement WebSphere Process Server», à la page 114
- «Configuration de l'environnement WebSphere Partner Gateway», à la page 115

---

### Envoi de documents via le protocole système de fichiers

Les composants de WebSphere Process Server optimisent la liaison d'importation SCA (Service Component Architecture) J2C pour envoyer les données de gestion vers l'adaptateur WebSphere Adapter for Flat Files. Cet adaptateur écrit les données de gestion dans un répertoire du système de fichiers. Le récepteur de système de fichiers de WebSphere Partner Gateway lit le document de gestion à partir de ce répertoire, puis le transmet au partenaire d'échanges approprié.

---

### Réception de documents via le protocole système de fichiers

WebSphere Partner Gateway reçoit un document de gestion du partenaire d'échanges. Le gestionnaire de documents de WebSphere Partner Gateway écrit ce document dans le répertoire du système de fichiers. L'adaptateur WebSphere Adapter for Flat Files interroge le répertoire et lit le document de gestion. Ensuite, il appelle l'exportation de service SCA (Service Component Architecture) qui attend l'occurrence d'événements au niveau de l'adaptateur. Le service SCA reçoit le document.

**Remarque :** WebSphere Adapter for Flat Files attend une chaîne ou des données brutes qu'il peut lire ou écrire dans un fichier. Par conséquent, l'environnement WebSphere Process Server doit sérialiser l'objet métier sous forme de chaîne ou de données brutes. Les services de WebSphere Process Server doivent créer un composant qui sérialisera un objet métier en un document de gestion et vice-versa.

---

## Configuration de l'environnement WebSphere Process Server

Cette section présente les étapes de configuration de l'environnement WebSphere Process Server pour le transport à l'aide du système de fichiers sur WebSphere Application Server. Elles impliquent la création et la configuration des artefacts de WebSphere Process Server.

Cette section contient les informations suivantes :

- «Déploiement et configuration de WebSphere Adapter for Flat Files»
- «Création de composants SCA», à la page 115

### Déploiement et configuration de WebSphere Adapter for Flat Files

Dans une intégration reposant sur le système de fichiers, WebSphere Process Server optimise WebSphere Adapter for Flat Files, un adaptateur bidirectionnel capable de lire et écrire des données de gestion dans le système de fichiers.

Pour les communications entrantes, l'adaptateur prend en charge la fonction Read (lecture).

Pour les communications sortantes, il prend en charge les fonctions suivantes :

- Create (création)
- Append (ajout)
- Delete (suppression)
- Overwrite (écrasement)
- Retrieve (extraction)
- List (énumération)
- Exists (existence)

L'installation de WebSphere Adapter for Flat Files comprend un fichier RAR (Resource Adapter Archive) déployable sur WebSphere Process Server. Cette archive RAR contient des fichiers livrés avec l'adaptateur. Avant de déployer WebSphere Adapter for Flat Files, vous devez importer le fichier RAR de l'adaptateur et créer votre projet dans WebSphere Integration Developer.

Les sections suivantes résument les étapes nécessaires au déploiement et à la configuration de WebSphere Adapter for Flat Files. Reportez-vous à la documentation de WebSphere Integration Developer pour plus d'informations.

#### Importation du fichier RAR :

1. Basculez vers la perspective J2EE et importez le fichier RAR.
2. Indiquez l'emplacement d'origine du fichier (lieu où vous avez copié le fichier de l'adaptateur lors de l'installation), puis spécifiez un nom de projet.

Un nouveau projet J2EE Connector sera créé dans l'espace de travail.

#### Ajout de dépendances externes au projet :

Copiez les dépendances externes dans le module connectorModule de votre projet WebSphere Integration Developer et ajoutez-les au projet en tant que bibliothèques internes. Il n'est pas nécessaire de regrouper les dépendances dans le fichier EAR qui sera exporté. Reportez-vous à la documentation de WebSphere Integration Developer pour des informations plus détaillées.

### Configuration de l'adaptateur :

La configuration s'effectue à l'aide de l'assistant de reconnaissance des services d'entreprise (Enterprise Service Discovery) de WebSphere Integration Developer. Ce processus vous permet d'entrer toutes les informations nécessaires à la toute première configuration de l'adaptateur. Le résultat de l'exécution de l'assistant Enterprise Service Discovery est sauvegardé dans un module d'intégration métier, qui contient : les objets métier ; le fichier d'importation (qui décrit le traitement des événements sortants, tel que défini par ActivationSpec) ; le fichier d'exportation (qui décrit le traitement des événements entrants, tel que défini par InteractionSpec) ; et le fichier WSDL (Web Services Description Language). Votre module contient ainsi un composant qui vous permet de lire et écrire dans le système de fichiers.

### Création de composants SCA

Si vous souhaitez que votre composant SCA reçoive des documents de gestion depuis WebSphere Partner Gateway via le protocole système de fichiers, vous pouvez optimiser l'outil Enterprise Service Discovery de manière à générer les liaisons d'importation nécessaires à l'adaptateur pour fichiers à plat. Pour ce faire, vous devez indiquer Inbound (entrant) comme type de service dans l'assistant Enterprise Service Discovery. Pour le type de service entrant, le nom de la méthode est READ. Connectez votre exportation de composant SCA à la liaison d'importation générée par l'outil Enterprise Service Discovery.

Si vous souhaitez que votre composant SCA envoie des documents de gestion à WebSphere Partner Gateway via le protocole système de fichiers, vous pouvez optimiser l'outil Enterprise Service Discovery de manière à générer les liaisons d'exportation nécessaires à l'adaptateur pour fichiers à plat. Pour ce faire, vous devez indiquer Outbound (sortant) comme type de service dans l'assistant Enterprise Service Discovery. Connectez la liaison d'importation générée par l'outil Enterprise Service Discovery à la liaison d'exportation de votre composant SCA.

---

## Configuration de l'environnement WebSphere Partner Gateway

Cette section présente les étapes de configuration de l'environnement WebSphere Partner Gateway pour le transport à l'aide du système de fichiers sur WebSphere Application Server.

A l'aide de la console de communauté de WebSphere Partner Gateway :

1. Créez un récepteur de système de fichiers sur le concentrateur afin qu'il reçoive les documents envoyés au concentrateur à partir de WebSphere Process Server ou de partenaires externes.
2. Créez un profil de partenaire interne (s'il n'en existe aucun), incluant une destination de système de fichiers que WebSphere Partner Gateway utilisera pour envoyer des documents à WebSphere Process Server.
3. Créez des profils de partenaire externe, incluant des destinations de système de fichiers que WebSphere Partner Gateway utilisera pour envoyer des documents aux partenaires.
4. Importez tous les fichiers WSDL, mappes de transformation, packages RosettaNet, ou autres mécanismes de définition de document, afin d'afficher, sur la page Définition de flux de document de la console de communauté de WebSphere Partner Gateway, une définition de document pour le type de document en cours d'échange.

5. Créez des interactions entre les types de document que le concentrateur recevra (depuis WebSphere Process Server ou un partenaire externe) et ceux qu'il enverra (à WebSphere Process Server ou aux partenaires externes).
6. Créez des fonctions business-to-business dans les profils du partenaire interne et des partenaires externes pour indiquer les types de documents qu'ils peuvent envoyer ou recevoir.
7. Créez des interactions entre les types de document que le concentrateur recevra (depuis WebSphere Process Server ou un partenaire externe) et ceux qu'il enverra (à WebSphere Process Server ou aux partenaires externes).
8. Créez des connexions de partenaire entre le partenaire interne et les partenaires externes pour indiquer le partenaire source (expéditeur du document), le partenaire cible (son destinataire) et l'action que le concentrateur doit accomplir (le cas échéant) pour transformer le document.

---

## **Partie 3. Intégration à WebSphere InterChange Server**

Les chapitres suivants décrivent la procédure permettant d'intégrer WebSphere Partner Gateway à WebSphere InterChange Server.



---

## Chapitre 8. Présentation de l'intégration à InterChange Server

Ce chapitre décrit le processus d'intégration de WebSphere Partner Gateway à WebSphere InterChange Server.

### Remarques :

1. Pour obtenir une description du processus général utilisé pour intégrer WebSphere Partner Gateway à un système dorsal, voir Chapitre 2, «Planification de l'intégration dorsale», à la page 9.
2. Ce chapitre suppose que vous connaissiez WebSphere InterChange Server et les composants qui lui sont associés, tels que les collaborations, les objets métier et les adaptateurs.

En général, l'intégration de WebSphere Partner Gateway à un système dorsal est effectuée par deux personnes ou rôles distincts. Chaque rôle configure un composant particulier pour lequel ce rôle est compétent. Par conséquent, le présent chapitre répartit l'intégration à WebSphere InterChange Server en deux tâches : la configuration de WebSphere Partner Gateway et la configuration d'InterChange Server. Le tableau 35 répertorie ces rôles de configuration et indique les sections de ce chapitre qui contiennent les informations de configuration associées.

Tableau 35. Rôles de l'intégration à InterChange Server

Rôle de configuration	Pour plus d'informations
Configuration de WebSphere Partner Gateway	<ol style="list-style-type: none"><li>1. «Planification de l'intégration à InterChange Server», à la page 120.</li><li>2. «Configuration de WebSphere Partner Gateway pour InterChange Server», à la page 123.</li></ol>
Configuration de WebSphere InterChange Server	<ol style="list-style-type: none"><li>1. «Planification de l'intégration à InterChange Server», à la page 120.</li><li>2. «Configuration d'InterChange Server», à la page 127.</li></ol>

**Remarque :** Chacun de ces rôles de configuration peut être effectué séparément ; cependant des informations communes sont requises pour chaque rôle de sorte que les deux composants puissent communiquer.

Ce chapitre contient les informations suivantes :

- «Planification de l'intégration à InterChange Server», à la page 120
- «Configuration de WebSphere Partner Gateway pour InterChange Server», à la page 123
- «Configuration d'InterChange Server», à la page 127
- «Gestion des documents contenant des pièces jointes», à la page 132

## Planification de l'intégration à InterChange Server

Pour planifier l'intégration à WebSphere InterChange Server, suivez les étapes décrites à la section Chapitre 2, «Planification de l'intégration dorsale», à la page 9. Le tableau 36 récapitule les étapes d'intégration de WebSphere Partner Gateway à InterChange Server (ICS).

Tableau 36. Planification de l'intégration à WebSphere InterChange Server

Etape d'intégration	Pour plus d'informations
1. Confirmez qu'une version prise en charge de WebSphere InterChange Server est installée et accessible à WebSphere Partner Gateway.	Chapitre 3: «Versions d'InterChange Server prises en charge par WebSphere Partner Gateway»
2. Déterminez le protocole de gestion du document de WebSphere Partner Gateway.	Chapitre 2: «Quel est le protocole de gestion utilisé ?», à la page 9
3. Déterminez le type d'emballage pour le document : Aucun ou Backend Integration.	Chapitre 2: «Quel emballage allez-vous utiliser ?», à la page 26
4. Déterminez le protocole de transport à utiliser entre WebSphere Partner Gateway et WebSphere InterChange Server.	Chapitre 3: «Transports de messages pris en charge par InterChange Server», à la page 121
5. Configurez WebSphere Partner Gateway.	Chapitre 3: «Configuration de WebSphere Partner Gateway pour InterChange Server», à la page 123
6. Configurez les composants de WebSphere InterChange Server à utiliser via le protocole de transport choisi.	Chapitre 3: «Configuration d'InterChange Server», à la page 127

## Versions d'InterChange Server prises en charge par WebSphere Partner Gateway

La version 6.1 de WebSphere Partner Gateway prend en charge l'intégration aux versions suivantes d'InterChange Server :

- 4.3.0

InterChange Server est disponible sur de nombreuses plateformes, y compris Windows 2000 et plusieurs plateformes UNIX. Pour plus d'informations, consultez le guide d'installation d'InterChange Server dans la documentation WebSphere InterChange Server.

Les adaptateurs WBI permettent d'intégrer WebSphere Partner Gateway à Interchange Server. Les versions de ces adaptateurs prises en charge sont les suivantes :

- WBIA HTTP Adapter v1.3
- WBIA JMS Adapter v2.8.x
- WBIA web services adapter v3.4.x
- 

Les composants WBI Data Handler sont utilisés avec les adaptateurs. Pour le traitement des messages XML, veillez à utiliser WebSphere Business Integration Data Handler for XML Version 2.3.1 ou ultérieure. Pour les messages cXML, vous devez utiliser le composant Data Handler for XML Version 2.4.1 ou ultérieure.

## Transports de messages pris en charge par InterChange Server

Lorsque WebSphere Partner Gateway envoie votre message à InterChange Server via un protocole de transport de message particulier, ce message est transmis à l'adaptateur approprié, qui comprend le protocole de transport et route le message vers InterChange Server. De même, lorsqu'InterChange Server envoie un message à WebSphere Partner Gateway, ce message est transmis à l'adaptateur approprié pour être routé vers WebSphere Partner Gateway via le protocole de transport approprié.

L'adaptateur fonctionne avec un gestionnaire de données, qui effectue la conversion des données sérialisées en objet métier ou inversement. Le gestionnaire de données correspondant au type de données utiles permet d'effectuer ces conversions.

Les deux protocoles de transport de message suivants sont pris en charge par InterChange Server :

- protocole de transport HTTP

**Remarque :** L'échange des services Web via le protocole HTTP est traité dans une autre section car il s'agit d'une opération différente du transfert de documents via HTTP. Voir «Envoi de documents SOAP via HTTP/S», à la page 165.

- Protocole de transport JMS

**Remarque :** InterChange Server fournit d'autres types d'options d'intégration, comme l'intégration basée sur les fichiers. Pour plus d'informations sur l'activation de l'échange de documents via l'intégration basée sur les fichiers, voir la documentation de WebSphere InterChange Server.

Utilisez le protocole de transport qui répond le mieux aux besoins de votre entreprise. Pour cela, tenez compte des informations suivantes :

- Avant toute chose, déterminez si le protocole de transport que vous utilisez entre le partenaire externe et WebSphere Partner Gateway est disponible avec le mécanisme d'intégration utilisé. Voir «Quel mode de transport de messages allez-vous utiliser ?», à la page 37.
- L'échange de documents SOAP sur WebSphere InterChange Server requiert l'utilisation du protocole de transport HTTP. Pour plus d'informations, voir «Envoi de documents SOAP via HTTP/S», à la page 165.

### HTTP

La détermination des adaptateurs requis pour l'échange de documents entre WebSphere Partner Gateway et InterChange Server via HTTP dépend des éléments suivants :

- le type de document envoyé ;
- la version d'InterChange Server avec laquelle vous effectuez l'intégration.

**Remarque :** Toutes les informations relatives au protocole de transport HTTP concernent également le protocole HTTPS.

Le tableau 37 indique où vous pouvez trouver des informations sur la manière de configurer les adaptateurs à utiliser avec InterChange Server.

Tableau 37. Configuration du protocole HTTP avec InterChange Server

Condition	Pour plus d'informations
Si vous transférez des documents autres que des documents SOAP à l'aide d'InterChange Server	«Utilisation du protocole de transport HTTP avec ICS», à la page 151
Si vous transférez des documents SOAP	«Envoi de documents SOAP via HTTP/S», à la page 165

## JMS

Les composants requis pour échanger des documents entre WebSphere Partner Gateway et InterChange Server via JMS sont présentés dans le tableau 63, à la page 170. En règle générale, la prise en charge du protocole JMS nécessite l'utilisation de WebSphere Business Integration Adapter for JMS. Le composant Adapter for JMS appelle des collaborations de manière asynchrone au sein d'InterChange Server.

## Avantages de chaque protocole de transport

Lorsque vous décidez du protocole de transport à utiliser avec InterChange Server, tenez compte des avantages que présente chaque mode :

- Le composant Adapter for HTTP prend en charge la communication synchrone. Si des transactions synchrones sont nécessaires, vous *devez* utiliser le protocole HTTP.
- Le composant Adapter for JMS peut assurer une "distribution garantie des événements" entre le composant WebSphere Partner Gateway et WebSphere InterChange Server.

La distribution garantie des événements offre l'assurance que les événements ne seront pas perdus ou envoyés deux fois.

## Prise en charge de l'intégration à InterChange Server

WebSphere Partner Gateway propose des exemples qui vous aident dans le processus d'intégration à InterChange Server. Ces exemples se trouvent dans le sous-répertoire suivant du répertoire de produit de WebSphere Partner Gateway : `Integration/WBI/WICS/samples`

Le tableau 38 énumère les sous-répertoires du répertoire `samples` pour les différents protocoles de transport pris en charge par InterChange Server.

Tableau 38. Exemples d'intégration à InterChange Server

Protocole de transport	Version d'InterChange Server	Sous-répertoire d'exemples
HTTP	Toutes les versions prises en charge	Exemples généraux : HTTP Exemples RosettaNet : RosettaNet/HTTP
JMS	Toutes les versions prises en charge	Exemples généraux : JMS Exemples RosettaNet : RosettaNet/JMS

---

## Configuration de WebSphere Partner Gateway pour InterChange Server

La section «Configuration de WebSphere Partner Gateway», à la page 47 donne un aperçu général sur la manière dont WebSphere Partner Gateway est configuré pour communiquer avec un système dorsal. Cette section résume les étapes nécessaires pour configurer WebSphere Partner Gateway afin de communiquer avec InterChange Server.

La configuration de WebSphere Partner Gateway implique les étapes suivantes :

- Configuration de la prise en charge des documents sortants  
Pour plus d'informations sur l'envoi de documents à partir de WebSphere Partner Gateway à InterChange Server, voir «Prise en charge des documents sortants».
- Configuration des documents entrants  
Pour plus d'informations sur l'envoi de documents à partir d'InterChange Server à WebSphere Partner Gateway, voir «Prise en charge des documents entrants», à la page 125.

### Prise en charge des documents sortants

Pour que WebSphere Partner Gateway puisse envoyer des documents à un système dorsal, vous devez effectuer les opérations expliquées à la section «Définition de la cible d'envoi du document du partenaire», à la page 49. Lorsque votre système dorsal est InterChange Server, vous devez créer une destination dont le type de transport correspond au protocole de transport utilisé pour l'échange de messages entre WebSphere Partner Gateway et InterChange Server. Lorsque le concentrateur envoie un document à InterChange Server, il doit savoir vers où router le document. Cet emplacement doit être conforme au protocole de transport utilisé. Le protocole de transport doit être pris en charge par InterChange Server (voir «Transports de messages pris en charge par InterChange Server», à la page 121).

Les sections ci-dessous résument comment créer des destinations pour les protocoles de transport suivants pris en charge par InterChange Server :

- «Configuration pour l'envoi de documents à ICS via le protocole de transport HTTP»
- «Configuration pour l'envoi de documents à ICS via le protocole de transport JMS», à la page 124

### Configuration pour l'envoi de documents à ICS via le protocole de transport HTTP

Lorsque le concentrateur envoie un document à InterChange Server via le protocole HTTP, il transmet le message via la destination définie. Cette destination identifie l'adresse URL à laquelle InterChange Server peut recevoir le document. Si InterChange Server utilise le protocole HTTP, un adaptateur reçoit le document à l'adresse URL en question, d'où il peut ensuite le transmettre à InterChange Server.

Pour que le concentrateur puisse envoyer des documents via une destination via le protocole de transport HTTP, créez une destination sur la page Caractéristiques de la destination de la console de communauté. Cette destination doit être configurée pour utiliser le protocole de transport HTTP 1.1 et écrire à l'adresse URL de l'adaptateur situé à l'extrémité.

**Remarque :** La section «Définition de la cible d'envoi du document du partenaire», à la page 49 décrit comment créer une destination.

## Configuration pour l'envoi de documents à ICS via le protocole de transport JMS

Lorsque le concentrateur envoie un document à InterChange Server via le protocole JMS, il transmet le document à la file d'attente JMS correspondante, d'où il peut être ensuite récupéré par InterChange Server. Pour que le concentrateur puisse obtenir cet emplacement JMS, vous devez créer dans WebSphere Partner Gateway une destination qui utilise le protocole de transport JMS. Cette destination doit être configurée pour écrire dans la file d'attente à partir de laquelle l'adaptateur JMS effectue l'interrogation.

**Remarque :** Pour plus d'informations sur la création d'une destination, voir «Définition de la cible d'envoi du document du partenaire», à la page 49.

Pour que le concentrateur puisse envoyer des documents via une destination via le protocole de transport JMS, créez une destination sur la page Caractéristiques de la destination de la console de communauté. Si vous utilisez comme fournisseur JMS la version 6.0 de WebSphere MQ incluant le dernier groupe de correctifs, reportez-vous au *Guide de configuration du concentrateur WebSphere Partner Gateway* pour consulter les étapes détaillées. Par ailleurs, indiquez les informations contenues dans le tableau 39 concernant le protocole JMS sur la page Caractéristiques de la destination.

Tableau 39. Valeurs JMS de la page Caractéristiques de la destination pour la communication avec ICS

Zone Caractéristiques de la destination	Valeur	Remarques et limitations
Classe de message JMS	TextMessage, BytesMessage, or StreamMessage	
Nom de file d'attente JMS	Nom de file d'attente JMS identique à celui de la file d'attente d'entrée définie pour Adapter for JMS	Cette file d'attente doit être incluse dans la liste des files d'attente d'entrée d'Adapter for JMS ; à savoir que l'adaptateur doit interroger cette file d'attente pour obtenir les événements entrants. Pour plus d'informations, voir «Identification des files d'attente JMS», à la page 177.

## Prise en charge des documents entrants

Pour que WebSphere Partner Gateway puisse recevoir des messages à partir d'un système dorsal, vous devez effectuer les opérations indiquées à la section «Définition de l'emplacement à partir duquel le document dorsal sera récupéré», à la page 53. Si InterChange Server est votre système dorsal, vous devez effectuer les opérations suivantes dans le concentrateur :

1. Dans le cadre de la définition du profil de votre partenaire interne, spécifiez le type de destination et indiquez l'adresse IP associée à laquelle le récepteur écoute.
2. Créez un récepteur dont le type de transport correspond au protocole de transport utilisé pour l'échange de documents entre WebSphere Partner Gateway et InterChange Server.

Pour que le concentrateur reçoive un document depuis InterChange Server, il doit connaître l'emplacement où récupérer les messages. Cet emplacement doit être compatible avec le protocole de transport utilisé. Le protocole de transport doit être pris en charge par InterChange Server (voir «Transports de messages pris en charge par InterChange Server», à la page 121).

Les sections suivantes résument les étapes de création de récepteurs pour les protocoles de transport pris en charge par InterChange Server.

## Configuration pour la réception de documents depuis ICS via le protocole de transport HTTP

Lorsque le concentrateur reçoit un document via le protocole de transport HTTP, son récepteur extrait le document du récepteur défini. Ce récepteur défini identifie l'adresse URL à laquelle le récepteur du concentrateur récupère les documents envoyés par InterChange Server. Lorsque InterChange Server utilise le protocole HTTP, un adaptateur envoie le document à une adresse URL, où il est récupéré par le concentrateur.

Pour que le concentrateur puisse recevoir les documents via un récepteur via le protocole de transport HTTP, vous devez créer un récepteur sur la page Liste des récepteurs de la console de communauté. Ce récepteur doit utiliser le protocole de transport HTTP 1.1. Le concentrateur détermine cette URL sous la forme d'une combinaison des informations suivantes :

- l'adresse IP de l'ordinateur hôte, obtenue auprès du profil du partenaire interne ;
- l'adresse URL du récepteur, obtenue dans la zone URL de la définition du récepteur.

**Remarque :** La section «Définition de l'emplacement à partir duquel le document dorsal sera récupéré», à la page 53 explique comment créer un récepteur.

Pour qu'InterChange Server puisse transmettre des documents à ce récepteur, son adaptateur doit être configuré pour transmettre des documents à cette adresse URL. Par conséquent, vous devez vous assurer que ce récepteur URL est disponible pour la configuration d'InterChange Server.

## Configuration pour la réception de documents depuis ICS via le protocole de transport JMS

Lorsque le concentrateur reçoit un document à partir d'InterChange Server via le protocole JMS, il récupère le document transmis par InterChange Server dans une file d'attente JMS. Pour que le concentrateur puisse obtenir cet emplacement JMS, vous devez créer dans WebSphere Partner Gateway un récepteur qui utilise le protocole de transport JMS. Par le biais de ce récepteur, le concentrateur interroge la file d'attente d'entrée pour savoir si des documents ont été transmis, puis les récupère.

**Remarque :** Pour plus d'informations sur la création d'un récepteur, voir «Définition de l'emplacement à partir duquel le document dorsal sera récupéré», à la page 53.

Pour que le concentrateur puisse recevoir les documents via un récepteur via le protocole de transport JMS, vous devez créer un récepteur sur la page Liste des récepteurs de la console de communauté. Si vous utilisez comme fournisseur JMS la version 6.0 de WebSphere MQ incluant le dernier groupe de correctifs, reportez-vous au *Guide de configuration du concentrateur WebSphere Partner Gateway* pour consulter les étapes détaillées. Par ailleurs, indiquez les informations contenues dans le tableau 40 concernant le protocole JMS sur la page Caractéristiques du récepteur.

Tableau 40. Valeurs JMS de la page Caractéristiques du récepteur pour la communication avec ICS

Zone Caractéristiques du récepteur	Valeur	Remarques et limitations
Classe de message JMS	TextMessage	

Tableau 40. Valeurs JMS de la page Caractéristiques du récepteur pour la communication avec ICS (suite)

Zone Caractéristiques du récepteur	Valeur	Remarques et limitations
Nom de file d'attente JMS	Nom de file d'attente JMS identique à la file d'attente de sortie définie pour Adapter for JMS	Cette file d'attente doit être incluse dans la liste des files d'attente de sortie d'Adapter for JMS ; à savoir que l'adaptateur doit envoyer les documents à cette file d'attente. Pour plus d'informations, voir «Identification des files d'attente JMS», à la page 177.

## Configuration d'InterChange Server

Pour vos interactions entre WebSphere Partner Gateway et InterChange Server, vous devez créer un composant Integration Component Library (ICL) dans l'outil System Manager. Cet ICL contiendra les artefacts suivants :

- des définitions d'objet métier ;
- des objets de connecteur ;
- des modèles et objets de collaboration.

Vous devez également créer un produit utilisateur et sélectionner les artefacts ICL requis pour l'interaction spécifique entre InterChange Server et WebSphere Partner Gateway.

**Remarque :** Pour plus d'informations sur la manière de créer des ICL et de configurer InterChange Server, voir le document *System Implementation Guide* dans la documentation de WebSphere InterChange Server.

## Création de définitions d'objets métier

WebSphere Partner Gateway envoie votre message à un adaptateur, qui le transfère à InterChange Server sous la forme d'un ou plusieurs *objets métier*. Pour qu'InterChange Server identifie un objet métier, il doit au préalable localiser un modèle, appelé *définition d'objet métier*, qui est utilisé pour décrire la structure des informations présentes dans l'objet métier. Chaque information présente dans une définition d'objet métier est contenue dans un *attribut*. Par conséquent, vous devez créer des définitions d'objet métier pour représenter les informations contenues dans votre message. Pour créer des définitions d'objet métier, utilisez l'outil Business Object Designer.

**Remarque :** L'outil Business Object Designer est inclus dans les produits WebSphere InterChange Server et WebSphere Business Integration Adapter. Pour plus d'informations sur l'utilisation de cet outil, voir le document *Business Object Development Guide*.

InterChange Server utilise les objets métier pour indiquer les informations suivantes :

- «Objet métier défini pour un document», à la page 128
- «Objets métier définis pour les informations de configuration», à la page 130

## Objet métier défini pour un document

Pour contenir les données utiles du document ou du message envoyé par WebSphere Partner Gateway, vous devez réaliser une définition d'objet métier représentant l'objet métier données utiles. L'adaptateur transmet le document à InterChange Server ou le récupère sous la forme d'un objet métier données utiles. Cette section contient des informations relatives à l'objet métier données utiles :

- «Structure d'objet métier»
- «Conversion d'objet métier», à la page 129
- «Terminologie d'InterChange Server», à la page 130

**Structure d'objet métier :** L'objet métier données utiles doit être conçu de sorte que chaque information contenue dans le document que vous voulez transférer possède un attribut dans la définition d'objet métier données utiles associée. Comme le montre le tableau 41, le contenu de l'objet métier données utiles est fonction de la structure du document et du type d'emballage utilisé par le document.

Tableau 41. Relation entre l'emballage et la structure de l'objet métier données utiles

Structure du document	Type d'emballage	Définition d'objet métier données utiles
Données utiles uniquement	Aucun	Conserve les informations de données utiles du document
Données utiles uniquement	Backend Integration	Contient : <ul style="list-style-type: none"> <li>• les informations de données utiles du document ;</li> <li>• les en-têtes de niveau transport.</li> </ul>
Données utiles et pièces jointes	Aucun	<i>Non applicable.</i> Vous devez utiliser l'emballage Backend Integration si votre document contient des pièces jointes.
Données utiles et pièces jointes  Le document contient un encapsuleur XML, appelé enveloppe de transport, qui encapsule les données utiles et les pièces jointes.	Backend Integration	Contient : <ul style="list-style-type: none"> <li>• les informations de données utiles du document ;</li> <li>• les en-têtes de niveau transport.</li> <li>• le conteneur de pièces jointes, qui stocke les données de pièces jointes et tous les objets métier Pièce jointe.</li> </ul> <p>Un gestionnaire de données fourni par WebSphere Partner Gateway, appelé gestionnaire de données de pièces jointes, est requis pour le traitement de l'enveloppe de transport. Pour plus d'informations, voir «Gestion des documents contenant des pièces jointes», à la page 132.</p>

L'objet métier données utiles doit également être conçu selon les besoins de l'adaptateur particulier qui est utilisé pour l'intégration à WebSphere Partner Gateway. Le tableau 42 indique comment obtenir des informations sur la manière de créer un objet métier données utiles pour le transférer via un protocole de transport donné.

Tableau 42. Création d'objets métier données utiles pour chaque protocole de transport

Protocole de transport	Remarques et limitations	Pour plus d'informations
HTTP		«Création de définitions d'objet métier pour ICS via HTTP», à la page 157

Tableau 42. Création d'objets métier données utiles pour chaque protocole de transport (suite)

Protocole de transport	Remarques et limitations	Pour plus d'informations
JMS	Si le document utilise l'empaquetage Backend Integration	«Création de définitions d'objet métier pour JMS», à la page 177
Tous	Si le document contient des pièces jointes	«Création de définitions d'objets métier liées aux pièces jointes», à la page 144

**Conversion d'objet métier :** En général, l'adaptateur utilise un gestionnaire de données pour la conversion du format document en représentation d'objet métier associée. Ce gestionnaire de données est appelé *gestionnaire de données utiles*. L'adaptateur doit être configuré pour appeler le gestionnaire de données approprié au type de contenu des données utiles. En général, WebSphere Business Integration Data Handler for XML est configuré sous forme de gestionnaire de données utiles car il permet la conversion des messages XML en objets métier. Cependant, vous pouvez créer des gestionnaires de données personnalisés pour les formats de message qui n'ont pas de gestionnaire de données correspondant fourni par WebSphere Business Integration Server.

**Remarque :** Pour le traitement des messages XML, veuillez à utiliser WebSphere Business Integration Data Handler for XML Version 2.3.1 ou ultérieure. Pour les messages cXML, vous devez utiliser le composant Data Handler for XML Version 2.4.1 ou ultérieure.

Vous devez vous assurer que les gestionnaires de données utiles que vous utilisez peuvent ignorer les méta-objets enfant requis par le protocole de transport utilisé. Avant d'utiliser un gestionnaire de données (qu'il soit fourni par WebSphere Business Integration ou qu'il s'agisse d'un gestionnaire de données personnalisé), assurez-vous qu'il prend en charge les méta-objets enfant. Reportez-vous à la section relative à la balise *cw\_mo\_libellé* dans les informations propres à l'application de l'objet métier, dans la section appropriée de votre protocole de transport (voir tableau 42, à la page 128).

Pour indiquer quel gestionnaire de données utiliser pour convertir les données utiles, vous devez procéder comme suit :

1. Identifiez le type MIME que le gestionnaire de données doit prendre en charge pour convertir les données utiles et localiser un gestionnaire de données capable de traiter ce type MIME.  
Le document *Data Handler Guide*, dans la documentation de WebSphere Business Integration Adapter, décrit les gestionnaires de données fournis par IBM. Si vous ne pouvez utiliser aucun de ces gestionnaires de données, vous pouvez créer un gestionnaire de données personnalisé.
2. Dans Business Object Designer, créez un méta-objet enfant pour le gestionnaire de données que vous devez utiliser. Si vous utilisez un gestionnaire de données fourni par IBM, voir le document *Data Handler Guide* pour plus d'informations sur la structure du méta-objet enfant.
3. Dans Business Object Designer, mettez à jour le méta-objet du gestionnaire de données de niveau supérieur des connecteurs pour inclure un attribut correspondant au type MIME pris en charge. Le type de cet attribut correspond au méta-objet enfant du gestionnaire de données.

4. Dans Connector Configurator, définissez les propriétés de configuration du connecteur afin d'identifier le gestionnaire de données à utiliser :
  - Attribuez aux propriétés `DataHandlerConfigMO` et `DataHandlerMimeType`, respectivement, le nom du méta-objet du gestionnaire de données de niveau supérieur et le type MIME pris en charge.
  - Affectez à la propriété `DataHandlerClassName` le nom de la classe du gestionnaire de données à instancier.

**Remarque :** Vous définissez *soit* les propriétés `DataHandlerConfigMO` et `DataHandlerMimeType`, *soit* la propriété `DataHandlerClassName`.

5. Dans Connector Configurator, ajoutez le méta-objet du gestionnaire de données de niveau supérieur dans la liste des objets métier pris en charge.

**Terminologie d'InterChange Server :** Pour InterChange Server, le nom de l'objet métier données utiles dépend du sens de la communication :

- Lorsque WebSphere Partner Gateway *envoie* un document à InterChange Server, il participe à la *notification d'événements* d'InterChange Server.

Dans ce cas, l'objet métier données est appelé *objet métier événement* (parfois appelé simplement événement) et il informe InterChange Server du déclenchement d'un événement chez un partenaire de la communauté.

- Lorsque WebSphere Partner Gateway *reçoit* un document d'InterChange Server, il participe au *traitement de demandes* d'InterChange Server.

Dans ce cas, l'objet métier données est un *objet métier de type demande* qu'InterChange Server a envoyé pour demander des informations à un partenaire externe. En réponse, InterChange Server peut envoyer un *objet métier de type réponse* à la communauté du concentrateur.

## Objets métier définis pour les informations de configuration

Pour de nombreux adaptateurs, vous créez des définitions d'objets métier afin d'y intégrer les informations de configuration. Ces objets métier sont souvent appelés des *méta-objets*.

Le tableau 43 indique comment obtenir des informations sur la manière de créer un objet métier données pour le transférer via un protocole de transport donné.

Tableau 43. Sections sur la création d'objets métier

Protocole de transport	Composant associé	Pour plus d'informations
HTTP	Adapter for HTTP	«Création d'informations d'en-tête de niveau transport HTTP pour ICS», à la page 162
JMS	Adapter for JMS	«Création des informations d'en-tête JMS», à la page 179
Tous	Gestionnaire de données de pièces jointes	«Création du méta-objet enfant de pièces jointes», à la page 140

## Création des connecteurs

Vous devez créer un objet connecteur pour l'adaptateur que vous utiliserez. Cet objet connecteur représente une instance de l'adaptateur lors de l'exécution. Vous pouvez créer des objets de connecteur dans l'outil System Manager d'InterChange Server.

**Remarque :** Pour plus d'informations sur la création d'objets connecteur, voir *System Implementation Guide* dans la documentation de WebSphere InterChange Server.

Le tableau 44 indique où trouver des informations sur la procédure de création d'objets connecteur, en fonction du protocole de transport utilisé.

Tableau 44. Création d'objets connecteur pour chaque protocole de transport

Protocole de transport	Adaptateur	Pour plus d'informations
HTTP	Adapter for HTTP	«Création de l'objet connecteur HTTP», à la page 165
JMS	Adapter for JMS	«Création de l'objet connecteur JMS», à la page 182

## Création des collaborations

C'est la *collaboration*, dans InterChange Server, qui exécute le processus métier en cours dont vous avez besoin. Par conséquent, vous devez créer la collaboration appropriée pour InterChange Server afin de traiter correctement les documents de WebSphere Partner Gateway. Effectuez les opérations suivantes afin de rendre la collaboration appropriée disponible lors de l'exécution :

1. Vérifiez qu'un modèle de collaboration contenant le processus métier dont vous avez besoin existe :
  - Si ce modèle de collaboration n'existe *pas*, vous devez le créer et le compiler.
  - Si un modèle de collaboration existe, vous devez connaître suffisamment son fonctionnement pour être capable de configurer son objet collaboration.
2. Créez un objet collaboration et associez ses ports, comme suit :
  - Pour le traitement des demandes, vous devez affecter l'adaptateur au port "cible", qui envoie des demandes à WebSphere Partner Gateway.
  - Pour la notification d'événements, vous devez affecter l'adaptateur au port "source", qui reçoit les événements à partir de WebSphere Partner Gateway.

Le tableau 45 indique où trouver des informations sur la procédure de création d'objets connecteur, en fonction du protocole de transport utilisé.

Tableau 45. Définition des accès à la collaboration pour chaque protocole de transport

Protocole de transport	Adaptateur	Pour plus d'informations
HTTP	Adapter for HTTP	«Liaison de collaborations pour communiquer avec Adapter for HTTP», à la page 165
JMS	Adapter for JMS	«Mise en place de liens de collaboration en vue d'une communication avec Adapter for JMS», à la page 183

## Déploiement du projet

Une fois que le projet de l'utilisateur contient les artefacts qui définissent les composants d'exécution requis, vous devez le déployer dans le référentiel InterChange Server. Vous pouvez déployer un projet d'utilisateur dans System Manager.

---

## Gestion des documents contenant des pièces jointes

WebSphere Partner Gateway fournit le gestionnaire de données de pièces jointes pour traiter les documents qui sont envoyés de WebSphere Partner Gateway à InterChange Server. Le gestionnaire de données de pièces jointes convertit un document dans l'enveloppe de transport XML (contenant ou non des pièces jointes) du format sérialisé en représentation d'objet métier associée. Vous devez configurer le gestionnaire de données de pièces jointes sous forme de gestionnaire de données utiles dans l'un des cas suivants :

- L'indicateur d'enveloppe de l'empaquetage Backend Integration a été paramétré sur Oui.

Lorsque cet indicateur a pour valeur Oui, WebSphere Partner Gateway encapsule systématiquement un document dans une enveloppe de transport XML, qu'il contienne ou non des pièces jointes. Le paramétrage de cet indicateur sur Oui pour l'empaquetage Backend Integration s'effectue sur la page Fonctions B2B du profil. Pour plus d'informations, voir «Données utiles», à la page 34.

- Le document à traiter peut contenir des pièces jointes.

Lorsqu'un document contient des pièces jointes, WebSphere Partner Gateway l'encapsule dans une enveloppe de transport XML. Tout flux de document contient des données utiles, et le cas échéant, des pièces jointes. Si vous envoyez ou recevez des documents qui contiennent des pièces jointes, l'objet métier données utiles doit contenir des informations relatives aux pièces jointes.

**Remarque :** Le gestionnaire de données de pièces jointes n'est *PAS* requis pour les documents SOAP qui contiennent des pièces jointes. Pour plus d'informations sur le mode de traitement des documents SOAP, voir «Envoi de documents SOAP via HTTP/S», à la page 165.

Le gestionnaire de données de pièces jointes est appelé par un adaptateur WebSphere Business Integration.

- Si WebSphere Partner Gateway et InterChange Server utilisent le protocole de transport HTTP, le gestionnaire de données de pièces jointes est appelé par le composant Adapter for HTTP.
- Si WebSphere Partner Gateway et InterChange Server utilisent le protocole de transport JMS, le gestionnaire de données de pièces jointes est appelé par le composant Adapter for JMS.

Lorsque l'adaptateur reçoit un document dans une enveloppe de transport XML, il appelle le gestionnaire de données de pièces jointes pour convertir le document en représentation d'objet métier appropriée. Par exemple, la figure 35, à la page 171 présente le composant Adapter for JMS qui appelle le gestionnaire de données de pièces jointes pour convertir le format sérialisé du document en représentation d'objet métier correspondante. A l'inverse, lorsque l'adaptateur reçoit une représentation d'objet métier pour un document dans une enveloppe de transport XML, il appelle le gestionnaire de données de pièces jointes pour convertir la structure d'objet métier au format de document approprié. Par exemple, la figure 36, à la page 173 présente le composant Adapter for JMS qui appelle le gestionnaire de données de pièces jointes pour convertir la représentation d'objet métier du document au format sérialisé.

Cette section contient des informations relatives au gestionnaire de données de pièces jointes :

- «Exécution de la conversion par le gestionnaire de données de pièces jointes»
- «Configuration de l'environnement pour le gestionnaire de données de pièces jointes», à la page 138
- «configuration du gestionnaire de données de pièces jointes», à la page 139
- «Création de définitions d'objets métier liées aux pièces jointes», à la page 144

## Exécution de la conversion par le gestionnaire de données de pièces jointes

Le gestionnaire de données de pièces jointes peut interpréter la structure de l'enveloppe de transport XML et traiter la conversion entre les données qu'elle contient et la représentation d'objet métier correspondante, en procédant comme suit :

- «Conversion de documents en objets métier» pour envoyer un document à InterChange Server ;
- «Conversion d'objets métier en documents», à la page 136 pour recevoir un document d'InterChange Server.

### Conversion de documents en objets métier

Avant que WebSphere Partner Gateway envoie un document à InterChange Server, il doit déterminer si le contenu doit ou non être encapsulé dans une enveloppe de transport XML. Si WebSphere Partner Gateway crée l'enveloppe de transport, les données utiles et toutes les pièces jointes sont codées au format Base64. WebSphere Partner Gateway envoie ensuite l'enveloppe de transport XML à l'adaptateur approprié à l'aide des en-têtes de niveau transport correspondants. Cet adaptateur peut être configuré pour appeler le gestionnaire de données de pièces jointes afin qu'il traite la conversion des données utiles et de toutes les pièces jointes d'un document encapsulé XML en représentation d'objet métier correspondante.

Pour convertir un document encapsulé dans une enveloppe de transport XML en représentation d'objet métier correspondante, l'entité appelante instancie le gestionnaire de données de pièces jointes afin de lui transmettre le document (dans son enveloppe de transport). Le gestionnaire de données de pièces jointes effectue ensuite les opérations suivantes :

1. Il charge les mappes de type de contenu définies dans le méta-objet enfant du gestionnaire de données.

Les mappes de type de contenu sont définies dans les propriétés de configuration `ContentTypeMap_x` du méta-objet enfant. Le méta-objet enfant est un objet métier qui contient des informations de configuration destinées au gestionnaire de données de pièces jointes. Les attributs de cet objet métier associent des mappes de type de contenu à des types de contenu. Pour plus d'informations, voir «Création du méta-objet enfant de pièces jointes», à la page 140.

2. Il vérifie le document pour savoir s'il est encapsulé dans une enveloppe de transport XML.

- Si le gestionnaire de données de pièces jointes ne détecte *PAS* l'enveloppe de transport, il n'a pas besoin d'extraire les données utiles de cette structure d'enveloppe.

Le document ne contient que des données utiles que le gestionnaire de données de pièces jointes doit convertir en représentation d'objet métier

associée. Pour plus d'informations, voir «Traitement des documents sans enveloppe de transport».

- Si le gestionnaire de données de pièces jointes *détecte* l'enveloppe de transport, il doit extraire les données utiles et toutes les pièces jointes de cette structure d'enveloppe.

Le document contient des données utiles et éventuellement quelques pièces jointes. Par conséquent, le gestionnaire de données de pièces jointes doit convertir les données utiles *et* toutes les pièces jointes en représentation d'objet métier associée. Pour plus d'informations, voir «Traitement des documents avec enveloppe de transport».

3. Il définit l'objet métier données utiles résultant et renvoie cet objet métier à l'entité appelante.

**Traitement des documents sans enveloppe de transport :** Si le gestionnaire de données de pièces jointes détermine que le document n'est *PAS* contenu dans une enveloppe de transport XML, il n'a pas besoin d'extraire les données utiles de cette structure d'enveloppe. Par conséquent, le gestionnaire de données utilise la propriété de configuration `PayloadDataHandlerMimeType` (définie dans son méta-objet) pour obtenir le type MIME qui identifie le gestionnaire de données utiles par défaut à instancier pour les données utiles du document. Ce gestionnaire de données convertit les données utiles en objet métier données utiles correspondant et renvoie l'objet métier données utiles résultant à son entité appelante.

**Traitement des documents avec enveloppe de transport :** Si le gestionnaire de données de pièces jointes détermine que le document est contenu dans une enveloppe de transport XML, il doit extraire les données utiles et toutes les pièces jointes de cette structure d'enveloppe avant de les traiter. Par conséquent, le gestionnaire de données effectue les opérations suivantes pour traiter et convertir le document :

1. Il extrait les données utiles et toutes les pièces jointes de l'enveloppe de transport et décode les données utiles.

Les données utiles sont contenues dans la balise XML `<payload>`. Chaque pièce jointe est contenue dans une balise XML `<attachment>`.

2. Il recherche les mappes d'un type de contenu qui correspond à celui des données utiles.

Il utilise le type MIME spécifié dans la mappe de type de contenu correspondant pour créer une instance d'un gestionnaire de données. Ce dernier convertit les données utiles en objet métier données utiles correspondant et renvoie l'objet métier résultant au gestionnaire de données de pièces jointes.

3. Il crée l'objet métier informations de contenu pour les données utiles.

Le gestionnaire de données de pièces jointes vérifie les informations propres à l'application de niveau objet métier qui figurent dans la définition d'objet métier données utiles, puis détermine le nom de l'objet métier informations de contenu au moyen du nom d'attribut spécifié par la balise `cw_mo_bcg_content_info`. Le gestionnaire crée une instance de cet objet métier informations de contenu et définit les valeurs du type de contenu et du codage des données utiles.

4. Il crée l'objet métier conteneur de pièces jointes pour les données utiles.

Le gestionnaire de données de pièces jointes vérifie les informations propres à l'application qui figurent dans l'objet métier données utiles et détermine le nom de l'objet métier conteneur de pièces jointes dont le nom d'attribut est spécifié

par la balise `cw_mo_bcg_attachment`. Il crée une instance de l'objet métier conteneur de pièces jointes et le sauvegarde dans l'attribut approprié de l'objet métier données utiles.

Si la balise `cw_mo_bcg_attachment` n'existe pas (ou qu'elle est vide), on part du principe que le document ne contient pas de pièces jointes. Aucune autre étape de traitement n'est donc requise. Le gestionnaire de données de pièces jointes renvoie l'objet métier données utiles converti.

5. Il crée l'objet métier pièce jointe par défaut pour le conteneur de pièces jointes. Le gestionnaire de données de pièces jointes vérifie les informations propres à l'application qui figurent dans l'objet métier conteneur de pièces jointes et détermine le nom de l'objet métier pièce jointe par défaut dont le nom d'attribut est spécifié par la balise `cw_mo_bcg_default_attribute`. Il crée une instance de l'objet métier pièce jointe par défaut et le sauvegarde dans l'attribut approprié de l'objet métier conteneur de pièces jointes.
6. Il détermine si la pièce jointe doit être convertie en objet métier, en recherchant dans les mappes de type de contenu un type de contenu correspondant à celui de la pièce jointe.

Le gestionnaire de données de pièces jointes examine le type de contenu et le codage du jeu de caractères de la pièce jointe et vérifie si une entrée correspondante existe dans une mappe de type de contenu.

- Si *aucune* mappe de type de contenu correspondante n'est détectée, le gestionnaire de données de pièces jointes ne crée pas d'objet métier pour les données de pièces jointes.

Le gestionnaire de données crée donc une instance de l'objet métier pièce jointe par défaut, définit les valeurs du type de contenu et du codage dans son objet métier informations de contenu, puis définit les données de pièces jointes codées en base64 (sous la forme d'une chaîne) dans l'attribut `attachment`.

Le gestionnaire de données de pièces jointes renseigne ensuite l'objet métier conteneur de pièces jointes à l'aide de l'objet métier pièce jointe par défaut.

- S'il *trouve* une mappe du type de contenu, le gestionnaire de données de pièces jointes détermine si la pièce jointe doit être convertie en objet métier :
  - Si la propriété de configuration `ConvertAttachment` de la mappe de type de contenu correspondante a pour valeur `false`, le gestionnaire de données de pièces jointes crée une instance de l'objet métier pièce jointe par défaut, définit les valeurs du type de contenu et du codage dans l'objet métier informations de contenu, puis définit les données de pièces jointes codées en base64 (sous forme de chaîne) dans l'attribut `attachment`.

Le gestionnaire de données de pièces jointes renseigne ensuite l'objet métier conteneur de pièces jointes à l'aide de l'objet métier pièce jointe par défaut.

- Si la propriété de configuration `ConvertAttachment` de la mappe de type de contenu a pour valeur `true`, le gestionnaire de données de pièces jointes décode les données de pièces jointes et crée une instance d'un gestionnaire de données pour traiter les données de pièces jointes. Ce gestionnaire de données traite les octets décodés et renvoie l'objet métier pièce jointe correspondant.

Le gestionnaire de données de pièces jointes vérifie ensuite les informations propres à l'application de niveau objet métier qui figurent dans la définition d'objet métier pièce jointe, puis détermine le nom de l'objet métier informations de contenu au moyen du nom d'attribut spécifié par la balise `cw_mo_bcg_content_info`. Si cette balise existe, le

gestionnaire de données crée l'objet métier informations de contenu pour la pièce jointe et définit la valeur du type de contenu et du codage de la pièce jointe.

Pour finir, le gestionnaire de données de pièces jointes renseigne l'objet métier conteneur de pièces jointes à l'aide de l'objet métier pièce jointe.

## Conversion d'objets métier en documents

Avant que WebSphere Partner Gateway reçoive un document d'InterChange Server, un adaptateur doit déterminer si la représentation d'objet métier données utiles et de toutes les pièces jointes doit ou non être encapsulée dans l'enveloppe de transport XML. InterChange Server envoie l'objet métier à l'adaptateur approprié qui traite la conversion réelle. Ce gestionnaire de données peut être configuré pour appeler le gestionnaire de données de pièces jointes afin qu'il traite la conversion des données utiles et de tous les objets métier pièce jointe en données utiles et pièces jointes correspondantes, ainsi que la création d'une enveloppe de transport XML.

Pour convertir un objet métier données utiles comportant des pièces jointes en représentation d'enveloppe de transport associée, l'entité appelante instancie le gestionnaire de données de pièces jointes afin de lui transmettre l'objet métier données utiles. Le gestionnaire de données de pièces jointes effectue les opérations suivantes :

1. Il charge les mappes de type de contenu définies dans son méta-objet de configuration.  
Les mappes de type de contenu sont définies dans les propriétés de configuration `ContentTypeMap_x` du méta-objet enfant. Le méta-objet enfant est un objet métier qui contient des informations de configuration destinées au gestionnaire de données de pièces jointes. Les attributs de cet objet métier associent des mappes de type de contenu à des types de contenu. Pour plus d'informations, voir «Création du méta-objet enfant de pièces jointes», à la page 140.
2. Il vérifie l'objet métier pour déterminer si une enveloppe de transport XML doit être créée.
  - Si le gestionnaire de données de pièces jointes ne détermine *PAS* que le document requiert une enveloppe de transport, il n'a pas besoin d'encapsuler les données utiles dans cette structure d'enveloppe.  
Le document ne contient que des données utiles que le gestionnaire de données de pièces jointes doit créer à partir de la représentation d'objet métier associée. Le gestionnaire de données n'a pas besoin de créer une enveloppe de transport pour le document. Pour plus d'informations, voir «Création des documents sans enveloppe de transport», à la page 137.
  - Au contraire, si le gestionnaire de données de pièces jointes détermine que le document requiert une enveloppe de transport, il doit encapsuler les données utiles et toutes les pièces jointes dans cette structure d'enveloppe.  
Le document contient des données utiles et éventuellement quelques pièces jointes. Par conséquent, le gestionnaire de données de pièces jointes doit convertir la représentation d'objet métier données utiles en données utiles *et* pièces jointes et encapsuler ces composants dans une enveloppe de transport. Pour plus d'informations, voir «Création des documents avec enveloppe de transport», à la page 137.
3. Il définit les données utiles et toutes les balises de pièces jointes dans le document WebSphere Partner Gateway et renvoie ce dernier à l'entité appelante.

**Création des documents sans enveloppe de transport :** Si le gestionnaire de données de pièces jointes détermine que l'objet métier données utiles ne nécessite PAS d'enveloppe de transport XML, il n'a pas besoin d'encapsuler les données utiles dans la structure d'enveloppe. Par conséquent, le gestionnaire de données recourt au gestionnaire de données utiles par défaut pour convertir l'objet métier données utiles en document de données utiles correspondant. La propriété de configuration `PayloadDataHandlerMimeType` (définie dans le méta-objet enfant du gestionnaire de données de pièces jointes) contient le type MIME qui identifie le gestionnaire de données utiles par défaut à instancier pour l'objet métier données utiles. Ce gestionnaire de données reçoit l'objet métier données utiles sous forme d'argument et renvoie le document de données utiles résultant à son entité appelante.

**Création des documents avec enveloppe de transport :** Si le gestionnaire de données de pièces jointes détermine que l'objet métier données utiles *nécessite* une enveloppe de transport XML, il doit encapsuler les documents de données utiles et de pièces jointes dans cette structure d'enveloppe. Par conséquent, le gestionnaire de données effectue les opérations suivantes pour traiter et convertir l'objet :

1. Il extrait le type de contenu et le codage du jeu de caractères des données utiles.

La balise `cw_mo_bcg_content_info` des informations propres à l'application au niveau de l'objet métier données utiles indique le nom de l'attribut d'informations de contenu. Cet attribut contient le type de contenu et le codage des données utiles.

**Remarque :** Si l'attribut d'informations de contenu n'existe pas, utilisez le gestionnaire de données par défaut (identifié par le type MIME contenu dans la propriété de configuration `PayloadDataHandlerMimeType` du méta-objet enfant du gestionnaire de données de pièces jointes) pour convertir l'objet métier données utiles.

2. Il recherche les mappes d'un type de contenu qui correspond à celui des données utiles.

Le gestionnaire de données de pièces jointes utilise le type MIME spécifié dans la mappe de type de contenu correspondante pour créer une instance d'un gestionnaire de données utiles. Ce dernier convertit l'objet métier données utiles en document de données utiles correspondant et renvoie le document résultant au gestionnaire de données de pièces jointes. A partir de la chaîne renvoyée par le gestionnaire de données utiles, le gestionnaire de données de pièces jointes code les octets au format Base64 et enregistre le résultat dans la balise de données utiles de l'enveloppe de transport XML.

3. Il extrait le conteneur de pièces jointes de l'objet métier données utiles.

Le conteneur de pièces jointes réside dans l'attribut de conteneur de pièces jointes de l'objet métier données utiles. Les informations propres à l'application au niveau de l'objet métier données utiles contiennent la balise `cw_mo_bcg_attachment`, qui identifie l'attribut de conteneur de pièces jointes. Cet attribut contient les pièces jointes.

Si la balise `cw_mo_bcg_attachment` n'existe pas (ou qu'elle est vide), on part du principe que le document ne contient pas de pièces jointes. Aucune autre étape de traitement n'est donc requise. Le gestionnaire de données de pièces jointes renvoie les données utiles converties dans son enveloppe de transport.

4. Pour chaque pièce jointe, il détermine si celle-ci est représentée sous forme d'objet métier ou simplement de données.

- S'il s'agit simplement de données de pièces jointes, les informations propres à l'application au niveau de l'objet métier conteneur de pièces jointes

contiennent la balise `cw_mo_bcg_default_attribute`, qui identifie l'attribut de pièces jointes par défaut. Cet attribut contient les données de pièces jointes, qui sont récupérées par le gestionnaire de données de pièces jointes, extrait les données codées au format Base64, puis enregistre le résultat dans le document.

- Si la pièce jointe est représentée par un objet métier, ses informations propres à l'application de niveau attribut contiennent la balise `wbic_type` indiquant qu'elle contient un objet métier pièce jointe.

Le gestionnaire de données de pièces jointes effectue les opérations suivantes pour traiter l'objet métier pièce jointe :

- a. Il récupère le contenu de l'attribut de pièces jointes et recherche le type de contenu et le codage de la pièce jointe.

Les informations propres à l'application au niveau de l'objet métier pièce jointe contiennent la balise `cw_mo_bcg_content_info`, qui identifie l'attribut d'informations de contenu. Cet attribut contient le type de contenu et le codage de cette pièce jointe. Le gestionnaire de données de pièces jointes enregistre ces informations de contenu dans la balise de pièce jointe du document.

- b. Il recherche les mappes d'un type de contenu qui correspond à celui de la pièce jointe.

Le gestionnaire de données de pièces jointes utilise le type MIME spécifié dans la mappe de type de contenu correspondante pour créer une instance d'un gestionnaire de données. Ce dernier convertit l'objet métier pièce jointe en document de pièces jointes correspondant et renvoie le document résultant (sous forme de chaîne) au gestionnaire de données de pièces jointes.

- c. Il enregistre le résultat codé dans la balise de pièce jointe de l'encapsuleur XML du document.

Le gestionnaire de données de pièces jointes extrait les octets de la chaîne renvoyée (à l'aide du jeu de caractères, s'il y a lieu) et code les octets au format Base64. Il enregistre ensuite le résultat dans la balise de pièce jointe.

## Configuration de l'environnement pour le gestionnaire de données de pièces jointes

L'utilisation du gestionnaire de données de pièces jointes fourni par WebSphere Partner Gateway implique les opérations suivantes :

- «Spécification du schéma à utiliser»
- «Déploiement du gestionnaire de données de pièces jointes», à la page 139
- «configuration du gestionnaire de données de pièces jointes», à la page 139

### Spécification du schéma à utiliser

Vous pouvez utiliser le schéma par défaut pour le gestionnaire de données de pièces jointes ou un schéma (`wbipackaging_v1.1_ns.xsd`) permettant d'inclure l'ID de contenu dans l'emballage Backend Integration.

Pour utiliser le schéma `wbipackaging_v1.1_ns.xsd`, configurez la propriété `wbipackaging_version` dans le fichier `bcg.properties`. (Le fichier `bcg.properties` est décrit dans le *Guide de l'administrateur*.) Cette propriété est :

```
wbipackaging_version=1.n
```

où *n* est égal à 0 ou 1. La valeur par défaut de la propriété est 1.0.

Le méta-objet du gestionnaire de données de pièces jointes dispose de l'attribut `wbpackaging_version`, dont la valeur peut être 1.0 ou 1.1. Si vous donnez la valeur 1.1 à l'attribut, le gestionnaire de données de pièces jointes analyse et génère les messages XML contenant l'ID de contenu de la pièce jointe.

Pour définir l'ID de contenu de la pièce jointe, l'objet métier codage utilise l'attribut `contentId`. Lorsque le gestionnaire de données de pièces jointes génère le message XML à partir de l'objet métier, il utilise cet attribut pour créer la balise `contentId` de la pièce jointe. Lorsque le gestionnaire de données de pièces jointes génère l'objet métier à partir du message XML, il définit cet attribut, à l'aide de la valeur spécifiée dans la balise `contentId` du message XML.

### Déploiement du gestionnaire de données de pièces jointes

Le gestionnaire de données de pièces jointes et le fichier référentiel associé sont disponibles sur le support d'installation de WebSphere Partner Gateway, aux emplacements indiqués dans le tableau 46.

Tableau 46. Emplacement des composants pour le gestionnaire de données de pièces jointes

Composant	Emplacement
Gestionnaire de données de pièces jointes	Integration/WBI/WICS/Attachment/ bcgwbiattachmentdh.jar
Fichier référentiel	Integration/WBI/WICS/Attachment/ MO_DataHandler_DefaultAttachmentConfigV1.0.in  ou Integration/WBI/WICS/Attachment/ MO_DataHandler_DefaultAttachmentConfigV1.1.in

Déployez les fichiers sur le serveur Web conformément à la documentation relative au serveur Web.

### Spécification de l'emplacement du gestionnaire de données de pièces jointes

WebSphere InterChange Server a besoin de l'emplacement du gestionnaire de données de pièces jointes, afin de le charger au moment de l'exécution. Pour indiquer son emplacement, procédez comme suit :

1. Modifiez le script de démarrage ICS, `start_server.bat`, situé dans le sous-répertoire `bin` du répertoire produit d'InterChange Server (sur l'ordinateur où réside InterChange Server).
2. A la variable `CLASSPATH` dans ce fichier, ajoutez le fichier jar du gestionnaire de données de pièces jointes : `bcgwbiattachmentdh.jar`.

### configuration du gestionnaire de données de pièces jointes

La configuration du gestionnaire de données de pièces jointes comprend les étapes suivantes, relatives à la création d'objets métier de configuration :

- «Création du méta-objet enfant de pièces jointes», à la page 140
- «Mise à jour du méta-objet du gestionnaire de données de niveau supérieur», à la page 143

**Remarque :** Vous devez également créer les définitions d'objet métier liées aux pièces jointes pour le gestionnaire de données de pièces jointes. Pour plus d'informations, voir «Création de définitions d'objets métier liées aux pièces jointes», à la page 144.

## Création du méta-objet enfant de pièces jointes

Pour configurer le gestionnaire de données de pièces jointes, vous devez créer un méta-objet enfant afin d'indiquer le nom de classe et les propriétés de configuration dont a besoin le gestionnaire de données de pièces jointes. Pour créer ce méta-objet, vous devez créer une définition d'objet métier contenant les attributs indiqués dans le tableau 47. Utilisez Business Object Designer, qui fait partie des outils de WebSphere Business Integration, pour créer cette définition d'objet métier.

Le méta-objet enfant fournit le nom de classe et les propriétés de configuration dont a besoin le gestionnaire de données de pièces jointes. Dans l'outil Business Object Designer, créez un méta-objet enfant qui inclut les types MIME destinés aux données utiles et aux types de pièces jointes que vous avez prévu de recevoir.

Les attributs du méta-objet enfant sont indiqués dans le tableau 47. Un exemple de méta-objet pour le gestionnaire de données de pièces jointes est représenté dans la figure 28, à la page 142.

**Remarque :** Les exemples d'objets métier mentionnés dans ce chapitre n'incluent *pas* les attributs standard (comme `ObjectEventId`) qui sont requis par WebSphere InterChange Server mais non utilisés par le gestionnaire de données de pièces jointes.

Tableau 47. Propriétés de configuration du méta-objet enfant de pièces jointes

Nom d'attribut	Description
<code>ClassName</code>	Nom de classe (obligatoire), qui désigne la classe du gestionnaire de données suivante : <code>com.ibm.bcg.DataHandlers.AttachmentDataHandler</code>
<code>ContentTypeMap_x</code>	La mappe du type de contenu correspondant aux données utiles et aux types de pièce jointe que vous vous attendez à recevoir dans l'encapsuleur XML.  Pour plus d'informations, voir «Mappes de type de contenu».
<code>PayloadDataHandlerMimeType</code>	Type MIME utilisé pour identifier le gestionnaire de données par défaut, qui traite les données utiles auxquelles <i>aucune</i> pièce jointe n'est associée.
<code>wbipackaging_version</code>	Cet attribut peut avoir la valeur 1.0 ou 1.1. Si vous donnez la valeur 1.1 à l'attribut, le gestionnaire de données de pièces jointes analyse et génère les messages XML contenant l'ID de contenu de la pièce jointe.

**Important :** Pour affecter une valeur aux attributs figurant dans le tableau 47, définissez la valeur par défaut de l'attribut. Par exemple, si le gestionnaire de données de pièces jointes doit utiliser le gestionnaire de données XML comme son gestionnaire de données par défaut, affectez à la valeur par défaut de l'attribut `PayloadDataHandlerMimeType` la valeur `text/xml`.

**Mappes de type de contenu :** La mappe de type de contenu détermine le gestionnaire de données appelé par le gestionnaire de données de pièces jointes pour convertir les informations mises en forme dans le type de contenu associé. Par exemple, si le type de contenu des données utiles est `application/xml`, le gestionnaire de données de pièces jointes recherche une mappe de type de contenu dont l'attribut `ContentType` contient la valeur `application/xml`. Si aucun type de contenu correspondant n'est détecté, le gestionnaire de données considère qu'il ne doit *PAS* convertir la pièce jointe associée en objet métier.

Vous pouvez créer une mappe pour chacun de ces types de contenu, à l'aide des informations propres à l'application au niveau de l'attribut, comme indiqué dans le tableau 49.

Lorsque vous créez un attribut dans le méta-objet enfant qui représente une mappe de type de contenu, rappelez-vous les éléments suivants :

- Le format du nom de l'attribut content-type-map est le suivant :

ContentTypeMap\_x

où x correspond à un nombre entier qui identifie de manière exclusive la mappe de type de contenu dans la définition de l'objet métier.

**Remarque :** Vous devez classer les attributs ContentTypeMap\_x dans l'ordre. Par exemple, si vous avez trois mappes de type de contenu, leurs attributs doivent s'appeler ContentType\_1, ContentType\_2 et ContentType\_3.

- La valeur par défaut de l'attribut content-type-map doit contenir une combinaison de balises valides.

Le tableau 48 contient les balises contenues dans la valeur par défaut de cet attribut.

Tableau 48. Balises valides pour la valeur par défaut de l'attribut content-type-map

Nom de balise	Description	Obligatoire
ContentType	Type de contenu réel fourni dans l'enveloppe de transport (par exemple, text/xml).	Oui
MimeType	Type MIME utilisé pour identifier le gestionnaire de données pour qu'il convertisse le type de contenu associé en objet métier. Si l'élément MimeType n'est pas spécifié, le gestionnaire de données utilise la valeur de l'élément ContentType pour instancier le gestionnaire de données.	Non
CharSet	Nom du jeu de caractères (par exemple, UTF-8) que le gestionnaire de données de pièces jointes utilise pour convertir les octets en chaîne ou inversement.  Si vous ne spécifiez pas CharSet, le gestionnaire de données de pièces jointes procède comme suit : <ul style="list-style-type: none"> <li>• Pour les données entrantes, les octets de données qui résultent du décodage du message au format base64 sont utilisés pour la conversion en objet métier.</li> <li>• Pour les données sortantes, des appels sont effectués à la méthode du gestionnaire de données enfant qui renvoie les octets (et non pas une chaîne).</li> </ul>	Non
ConvertAttachment	Valeur booléenne qui indique si la pièce jointe doit ou non être convertie en objet métier. La valeur par défaut est false.	Non

La mappe du type de contenu peut également indiquer le jeu de caractères utilisé pour le codage et spécifier si une pièce jointe doit être ou non convertie en objet métier. Pour plus d'informations sur les attributs du méta-objet enfant, voir «Création du méta-objet enfant de pièces jointes», à la page 140.

Par exemple, supposez que le document contient les types de contenus suivants :

- application/xml
- text/xml
- application/octet-stream

Tableau 49. Exemples de mappes de type de contenu

Type de contenu	Nom d'attribut	Valeur par défaut
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

**Exemple de méta-objet :** WebSphere Partner Gateway fournit les fichiers référentiels d'entrée d'InterChange Server suivants, qui contiennent un exemple de méta-objet enfant pour le gestionnaire de données de pièces jointes :

```
RépProduit/Integration/WBI/WICS/Attachment/
MO_DataHandler_DefaultAttachmentConfigV1.0.in
RépProduit/Integration/WBI/WICS/Attachment/
MO_DataHandler_DefaultAttachmentConfigV1.1.in
```

où *RépProduit* est le répertoire d'installation de votre produit WebSphere Partner Gateway. Les fichiers référentiels définissent un gestionnaire de données de pièces jointes unique dont le méta-objet enfant associé est *MO\_DataHandler\_DefaultAttachmentConfig*. La figure 28 présente l'exemple de méta-objet enfant du gestionnaire de données de pièces jointes. Ce méta-objet définit deux mappes de type de contenu, *ContentTypeMap\_1* et *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

Figure 28. Exemple de méta-objet enfant pour le gestionnaire de données de pièces jointes

MO\_DataHandler\_DefaultAttachmentConfigV1.1.in contient l'attribut supplémentaire :

```
[Attribute]
  Name = WBIpackaging_Version
  Type = String
  MaxLength = 255
  IsKey = false
  IsForeignKey = false
  IsRequired = true
  DefaultValue = 1.0
  IsRequiredServerBound = false
[End]
```

## Mise à jour du méta-objet du gestionnaire de données de niveau supérieur

Un adaptateur WebSphere Business Integration (tel que Adapter for JMS) utilise le méta-objet MO\_DataHandler\_Default pour identifier les gestionnaires de données à utiliser. Ajoutez une référence au gestionnaire de données de pièces jointes dans le méta-objet.

Apportez les modifications suivantes au méta-objet MO\_DataHandler\_Default :

1. Ajoutez un attribut dont le nom identifie le type MIME associé à l'instance de gestionnaire de données de pièces jointes ; en d'autres termes, pour un document qui contient ce type MIME, le gestionnaire de données associé peut traiter sa conversion en objet métier.

Le type de cet attribut est la définition d'objet métier du méta-objet enfant du gestionnaire de données de pièces jointes (voir «Création du méta-objet enfant de pièces jointes», à la page 140).

2. Ajoutez un attribut pour chacun des types MIME de pièce jointe pris en charge, si ces derniers n'existent pas déjà dans le méta-objet de gestionnaire de données de niveau supérieur.

Le type de ces attributs serait le méta-objet enfant du gestionnaire de données associé.

Par exemple, supposez que vous disposez du gestionnaire de données de pièces jointes tel qu'il est configuré dans la figure 28, à la page 142. La figure 29 présente le méta-objet MO\_DataHandler\_Default dont l'attribut associe le type MIME `wbic_attachment` à l'instance du gestionnaire de données de pièces jointes qui est configuré par le méta-objet enfant MO\_DataHandler\_DefaultAttachmentConfig. Ce méta-objet de gestionnaire de données de niveau supérieur associe également le type MIME de document (`text/xml`) au méta-objet enfant du gestionnaire de données XML.

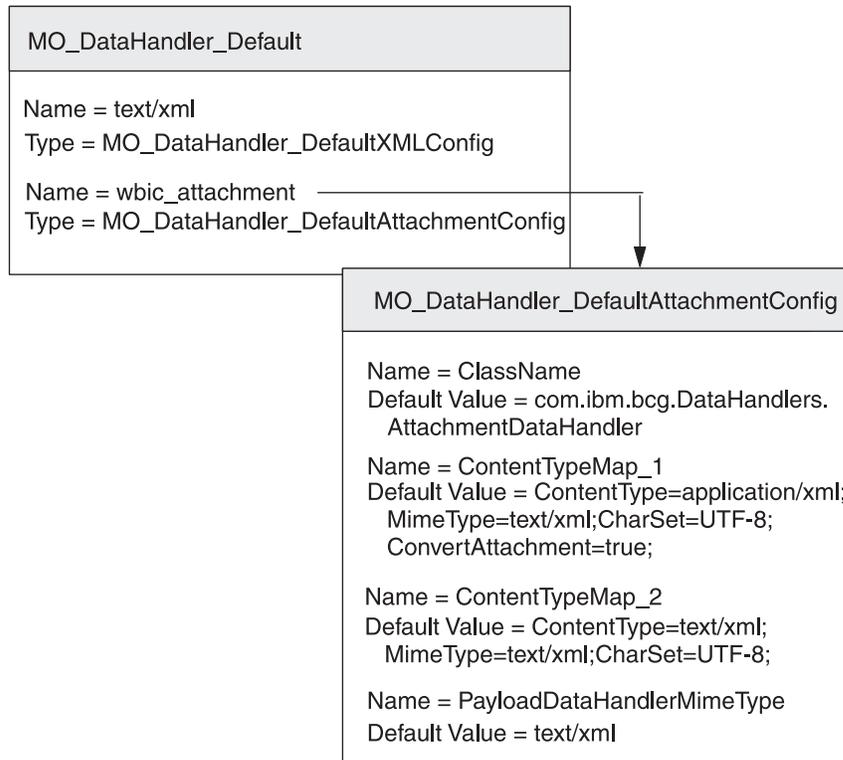


Figure 29. Association du type MIME wbic\_attachment MIME au gestionnaire de données de pièces jointes

Pour chaque combinaison unique des types de contenu pris en charge que vous devez prendre en charge, répétez ce processus en ajoutant un attribut dans le méta-objet de gestionnaire de données de niveau supérieur approprié dont le nom d'attribut est le type MIME associé à l'instance de gestionnaire de données de pièces jointes et dont le type est le nom du méta-objet enfant associé. Assurez-vous également que les types MIME configurés (ainsi que leurs méta-objets enfant) existent dans le méta-objet de niveau supérieur.

## Création de définitions d'objets métier liées aux pièces jointes

Si vous envoyez ou recevez des documents qui sont encapsulés dans une enveloppe de transport XML, l'objet métier données utiles doit contenir des informations relatives aux pièces jointes. Tout flux de document contient des données utiles, et le cas échéant, des pièces jointes. Le gestionnaire de données de pièces jointes s'attend à ce que ces informations de pièces jointes se trouvent dans des *objets métier liés aux pièces jointes*. Par conséquent, vous devez créer des définitions d'objet métier pour représenter ces informations. Une définition d'objet métier est la forme que prennent les informations utilisées par InterChange Server. Vous devez utiliser l'outil Business Object Designer pour créer des définitions d'objet métier.

La figure 30 présente la structure d'objet métier données utiles qui est encapsulée dans une enveloppe de transport XML.

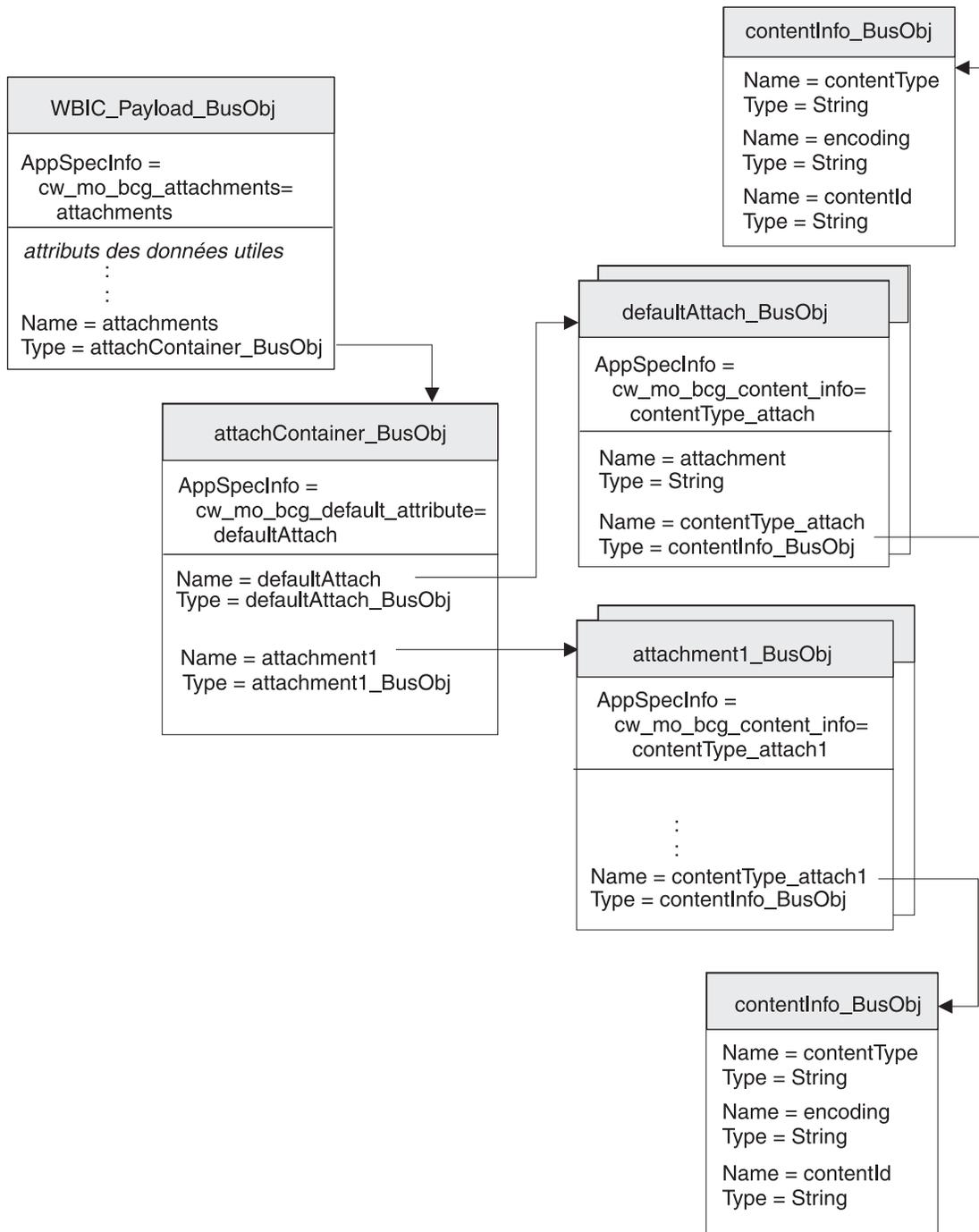


Figure 30. Relation entre l'objet métier données utiles et les objets métier pièce jointe

Comme le montre la figure 30, toutes les pièces jointes sont contenues dans l'objet métier conteneur de pièces jointes. S'il existe des pièces jointes, l'objet métier données utiles possède un attribut qui correspond à l'objet métier conteneur de pièces jointes.

Assurez-vous que la structure de l'objet métier contient des objets métier requis par des pièces jointes, en procédant comme suit :

1. Créez une définition d'objet métier destinée à contenir les propriétés de codage du type de contenu requises par l'empaquetage Backend Integration.

2. Créez une définition d'objet métier pour chaque type de pièce jointe.
3. Créez une définition d'objet métier pour l'objet métier conteneur de pièces jointes.
4. Modifiez la définition d'objet métier pour votre objet métier données utiles.

Chaque étape est décrite dans les sections ci-dessous.

## Représentation des informations de contenu

Pour stocker le type de contenu et le codage des données utiles ou des pièces jointes associées, créez l'objet métier *informations de contenu*. Pour créer une définition d'objet métier informations de contenu, créez les attributs présentés dans le tableau 50.

Tableau 50. Attributs de l'objet métier informations de contenu

Attribut	Type d'attribut	Description	S'agit-il d'un attribut clé ?
contentType	Chaîne	Type de contenu des données utiles ou des pièces jointes associées	Oui
encoding	Chaîne	Codage de caractères des données utiles ou des pièces jointes associées	Non

Dans la figure 30, à la page 145, la définition d'objet métier `contentInfo_BusObj` contient des attributs pour le type de contenu et le codage de la pièce jointe. Ces attributs possèdent tous des informations propres à l'application au niveau des attributs afin de spécifier le nom de l'en-tête du protocole associé. Par exemple, les informations propres à l'application de l'attribut `x-aux-sender-id` sont définies comme suit :

```
name=x-aux-sender-id
```

Vous pouvez choisir le nom de votre choix pour identifier la définition d'objet métier informations de contenu. Les informations propres à l'application de l'objet métier pièce jointe déterminent s'il s'agit d'un type d'objet métier Codage du type de contenu. La figure 30, à la page 145 illustre un exemple de définition d'objet métier Codage du type de contenu appelé `contentType_BusObj`.

## Représentation des données de pièces jointes

Pour les données de pièces jointes qui ne doivent pas être converties en objet métier, créez l'objet métier *pièce jointe par défaut*. Cet objet métier permet de stocker des données codées au format base64 qui émanent de l'enveloppe de transport.

Pour créer une définition d'objet métier pièce jointe par défaut, procédez comme suit :

1. Créez les attributs présentés dans le tableau 51, à la page 147.
2. Si vous créez l'objet métier informations de contenu, ajoutez, dans les informations propres à l'application pour la définition d'objet métier pièce jointe par défaut, la balise `cw_mo_bcg_content_info` afin d'identifier l'attribut qui contient les informations de contenu.

Cette balise `cw_mo_bcg_content_info` est au format suivant :

```
cw_mo_bcg_content_info=attributInformationsContenu
```

où `attributInformationsContenu` représente le nom de l'attribut qui contient l'objet métier conteneur de pièces jointes.

Tableau 51. Attributs de l'objet métier informations de pièces jointes par défaut

Attribut	Type d'attribut	Description	S'agit-il d'un attribut clé ?
attachment	Chaîne	Élément de données de pièces jointes. <b>Remarque :</b> Il s'agit de l'attribut clé de la définition d'objet métier.	Oui
Attribut stockant les informations de contenu	Objet métier	Attribut facultatif stockant l'objet métier informations de contenu, qui fournit le type de contenu et le codage des données de pièces jointes. Cet attribut doit avoir une cardinalité unique. <b>Remarque :</b> Si cet attribut n'existe PAS, le gestionnaire de données de pièces jointes ne définit pas les données de pièces jointes dans l'objet métier.  Pour plus d'informations sur le format de l'objet métier informations de contenu, voir «Représentation des informations de contenu», à la page 146.	Non

Dans la figure 30, à la page 145, la définition d'objet métier defaultAttach\_BusObj contient des attributs relatifs à l'élément de données de pièces jointes, y compris un objet métier informations de contenu stockant le type de contenu et le codage associés. L'élément de données de pièces jointes représenté par cet objet métier pièce jointe par défaut comporte pas un codage de type de contenu qui est représenté par l'attribut contentType\_attach. Par conséquent, la définition d'objet métier pièce jointe par défaut inclut la balise suivante dans ses informations propres à l'application au niveau de l'objet métier :

`cw_mo_bcg_content_info=contentType_attach`

## Représentation des pièces jointes

Pour chaque type de pièce jointe contenue dans votre document qui est converti en objet métier, vous devez créer une *définition d'objet métier pièce jointe* distincte. Celle-ci représente les données réelles figurant dans une pièce jointe au document. Pour créer une définition d'objet métier pièce jointe, procédez comme suit :

1. Créez un attribut pour chaque élément de données de pièces jointes.  
Les types d'attributs possibles peuvent inclure une chaîne (pour les éléments de données simples) ou une définition d'objet métier (pour les données complexes).
2. Si la pièce jointe nécessite un codage de type de contenu :
  - a. Créez un attribut de codage de type de contenu.  
Le type de cet attribut est la définition d'objet métier codage de type de contenu (voir «Représentation des informations de contenu», à la page 146).
  - b. Ajoutez aux informations propres à l'application au niveau de l'objet métier de la définition d'objet métier pièce jointe la balise `cw_mo_bcg_content_info`, afin d'identifier l'attribut qui contient le codage de type de contenu.  
Cette balise `cw_mo_bcg_content_info` est au format suivant :  
`cw_mo_bcg_content_info=attributCodageTypeContenu`  
où `attributCodageTypeContenu` est le nom de l'attribut qui contient l'objet métier Codage du type de contenu.

Dans la figure 30, à la page 145, le document de données utiles possède une pièce jointe, qui est représentée par la définition d'objet métier attachment1\_BusObj.

Cette pièce jointe comporte un codage de type de contenu représenté par l'attribut `contentTypeEncoding`. Par conséquent, la définition objet métier pièce jointe inclut la balise suivante dans ses informations propres à l'application au niveau de l'objet métier :

```
cw_mo_bcg_content_info=contentTypeEncoding
```

## Représentation du conteneur de pièces jointes

Le conteneur de pièces jointes contient toutes les pièces jointes de document dans l'enveloppe de transport. Pour représenter le conteneur de pièces jointes pour InterChange Server, créez l'objet métier conteneur de pièces jointes. Chaque attribut présent dans l'objet métier conteneur de pièces jointes représente une seule pièce jointe.

Pour créer une définition d'objet métier conteneur de pièces jointes, procédez comme suit :

1. Ajoutez un attribut pour chaque pièce jointe du document qui doit être convertie en objet métier.

Le type de chacun de ces attributs correspond à l'objet métier pièce jointe associé (voir «Représentation des pièces jointes», à la page 147). Chaque attribut doit avoir plusieurs cardinalités.

2. Ajoutez la balise `wbic_type` aux informations propres à l'application relatives à chaque attribut, afin d'identifier l'attribut sous la forme d'une pièce jointe.

La balise `wbic_type` est au format suivant :

```
wbic_type=Attachment
```

**Remarque :** Un attribut de pièce jointe peut avoir plusieurs cardinalités.

3. Si les données utiles contiennent des données de pièces jointes qui ne doivent PAS être converties en objet métier :

- a. Ajoutez un attribut pour l'objet métier pièce jointe par défaut.

Le type de cet attribut correspond à l'objet métier pièce jointe par défaut (voir «Représentation des données de pièces jointes», à la page 146). Il s'agit de l'attribut clé de l'objet métier conteneur de pièces jointes. Cet attribut ne nécessite PAS la balise `wbic_type` dans ses informations propres à l'application.

**Remarque :** L'objet métier conteneur de pièces jointes peut contenir *un seul* attribut de pièce jointe par défaut. Toutefois, cet attribut peut avoir plusieurs cardinalités.

- b. Ajoutez aux informations propres à l'application au niveau de l'objet métier de la définition d'objet métier pièce jointe la balise `cw_mo_bcg_default_attribute`, afin d'identifier l'attribut qui contient les données de pièces jointes.

Cette balise `cw_mo_bcg_default_attribute` est au format suivant :

```
cw_mo_bcg_content_info=attribut_pièce_jointe_par_défaut
```

où `attribut_pièce_jointe_par_défaut` représente le nom de l'attribut qui contient l'objet métier pièce jointe par défaut.

**Important :** Si aucun attribut de pièces jointes par défaut n'existe, le gestionnaire de données de pièces jointes *ne peut pas* convertir les pièces jointes qui ne possèdent pas une mappe de type de contenu associée ou des pièces jointes qui ne sont pas converties en objets métier. Ces pièces jointes sont perdues lors de la conversion en représentation d'objet métier.

Dans la figure 30, à la page 145, le conteneur de pièces jointes est représenté par la définition d'objet métier `attachContainer_BusObj`. Cette définition d'objet métier conteneur de pièces jointes comporte les attributs suivants :

- L'attribut `attachment1` représente la pièce jointe unique du document. Par conséquent, la définition objet métier conteneur de pièces jointes inclut la balise suivante dans ses informations propres à l'application au niveau de l'attribut :

```
wbic_type=Attachment
```

Cette pièce jointe est représentée par la définition d'objet métier `attachment1_BusObj`.

- L'attribut `defaultAttach` représente les données de pièces jointes qui n'ont PAS besoin d'être converties en représentation d'objet métier. Par conséquent, la définition objet métier conteneur de pièces jointes inclut la balise suivante dans ses informations propres à l'application au niveau de l'objet métier :

```
cw_mo_bcg_default_attribute=defaultAttach
```

### Modification de la définition d'un objet métier données utiles

La définition de l'objet métier données utiles représente les informations contenues dans votre document. Elle contient un attribut pour chaque information que vous devez transférer entre WebSphere Partner Gateway et InterChange Server. Pour plus d'informations sur la création de la définition d'un objet métier données utiles, voir «Objet métier défini pour un document», à la page 128.

Si vous envoyez ou recevez des documents qui contiennent des pièces jointes, l'objet métier données utiles doit contenir des informations relatives aux pièces jointes. Tout flux de document contient des données utiles, et le cas échéant, des pièces jointes. Si les données utiles de votre document contiennent des pièces jointes, vous devez modifier la définition de l'objet métier données utiles comme suit :

1. Créez un attribut stockant les données utiles.

Il est peut-être plus facile à utiliser si vos données utiles réelles sont stockées dans une définition d'objet métier données utiles distincte. Dans ce cas, l'objet métier données utiles de niveau supérieur contient pour les données utiles un attribut dont le type est la définition d'objet métier données utiles réelles.

2. Ajoutez un conteneur de pièces jointes :

- a. Ajoutez un attribut stockant le conteneur de pièces jointes.

Le type de cet attribut est la définition d'objet métier conteneur de pièces jointes (voir «Représentation du conteneur de pièces jointes», à la page 148). Cet attribut doit avoir une cardinalité unique.

- b. Dans les informations propres à l'application pour la définition d'objet métier données utiles, ajoutez la balise `cw_mo_bcg_attachment` afin d'identifier l'attribut qui contient le conteneur de pièces jointes.

Cette balise `cw_mo_bcg_attachment` est au format suivant :

```
cw_mo_bcg_attachment=attributConteneurPiècesJointes
```

où `attributConteneurPiècesJointes` est le nom de l'attribut qui contient l'objet métier Conteneur de pièces jointes.

3. Le cas échéant, vous pouvez indiquer le type de contenu des données utiles. Le gestionnaire de données de pièces jointes utilise ce type de contenu pour déterminer quel gestionnaire de données à instancier pour convertir les données utiles. S'il trouve un type de contenu correspondant dans les mappes de type de contenu, il instancie le gestionnaire de données pour ce type de contenu.

- a. Ajoutez un attribut d'informations de contenu correspondant à un attribut facultatif qui stocke le type de contenu et le codage des données utiles. Cet attribut doit avoir une cardinalité unique.

**Remarque :** Si cet attribut n'existe *PAS*, le gestionnaire de données de pièces jointes fait en sorte que le gestionnaire de données convertisse les données utiles à partir de la valeur de la propriété de configuration `PayloadDataHandlerMimeType` dans son méta-objet enfant.

- b. Dans les informations propres à l'application pour la définition d'objet métier données utiles, ajoutez la balise `cw_mo_bcg_content_info` afin d'identifier l'attribut qui contient les informations de contenu.

Cette balise `cw_mo_bcg_content_info` est au format suivant :

```
cw_mo_bcg_attachment=attributInformationsContenu
```

où *attributInformationsContenu* représente le nom de l'attribut qui contient l'objet métier informations de contenu. Pour plus d'informations sur le format de l'objet métier informations de contenu, voir «Représentation des informations de contenu», à la page 146.

4. Ajoutez tous les attributs de configuration requis pour votre protocole de transport.

Par exemple, si vous utilisez le protocole de transport JMS, votre définition d'objet métier données utiles doit contenir l'objet métier dynamique JMS. Pour plus d'informations, consultez la section relative à la création de définitions d'objet métier à l'appui de votre protocole de transport.

---

## Chapitre 9. Intégration d'InterChange Server via HTTP

Ce chapitre décrit le processus d'intégration de WebSphere Partner Gateway à WebSphere InterChange Server via le protocole de transport HTTP. Il contient des informations sur la manière de configurer InterChange Server (ICS) et les adaptateurs nécessaires à la communication via HTTP.

**Remarque :** Pour plus d'informations sur la manière de configurer WebSphere Partner Gateway pour communiquer avec InterChange Server via HTTP, voir «Configuration de WebSphere Partner Gateway pour InterChange Server», à la page 123. Pour obtenir des informations générales sur la configuration d'InterChange Server, voir «Configuration d'InterChange Server», à la page 127.

Ce chapitre explique comment échanger des documents entre WebSphere Partner Gateway et WebSphere InterChange Server via le protocole de transport HTTP :

- «Utilisation du protocole de transport HTTP avec ICS»
- «Envoi de documents SOAP via HTTP/S», à la page 165

---

### Utilisation du protocole de transport HTTP avec ICS

WebSphere Partner Gateway vous permet d'échanger des documents avec WebSphere InterChange Server (ICS) via le protocole de transport HTTP.

**Remarque :** Si vous échangez des documents SOAP via le protocole HTTP, voir «Envoi de documents SOAP via HTTP/S», à la page 165.

Cette section fournit les informations suivantes sur la manière de configurer InterChange Server et les adaptateurs appropriés en vue de leur utilisation avec WebSphere Partner Gateway via HTTP :

- «Composants requis pour les documents envoyés à ICS via HTTP»
- «Configuration de l'environnement pour le transport HTTP avec ICS», à la page 155
- «Création de définitions d'objet métier pour ICS via HTTP», à la page 157
- «Création d'artefacts ICS pour HTTP», à la page 164

### Composants requis pour les documents envoyés à ICS via HTTP

Pour que WebSphere Partner Gateway puisse communiquer avec InterChange Server via le protocole HTTP, vous devez configurer ces deux composants. tableau 52 résume ces étapes de configuration.

Tableau 52. Configuration de WebSphere Partner Gateway et d'InterChange Server

Composant	Version	Pour plus d'informations
WebSphere Partner Gateway	6.1	«Configuration pour l'envoi de documents à ICS via le protocole de transport HTTP», à la page 123  «Configuration pour la réception de documents depuis ICS via le protocole de transport HTTP», à la page 126
WebSphere InterChange Server	4.3	«Création d'artefacts ICS pour HTTP», à la page 164

Pour échanger un document entre WebSphere Partner Gateway et InterChange Server via le protocole HTTP, vous devez utiliser les composants répertoriés dans le tableau 53.

Tableau 53. Composants requis pour envoyer des documents à l'aide d'InterChange Server via HTTP

Composant	Description	Remarques et limitations
WebSphere Business Integration Adapter for HTTP (Adapter for HTTP)	Ce composant permet à InterChange Server d'échanger des objets métier avec des applications qui envoient ou reçoivent des données sous la forme de flux HTTP.	Utilisez la version 4.2.1 d'Adapter for HTTP.
Un gestionnaire de données utiles	Ce gestionnaire de données convertit les données utiles du document (généralement au format XML) en représentation d'objet métier.	Ce gestionnaire de données est indispensable et doit prendre en charge le type MIME du document contenant les données utiles.
Gestionnaire de données de pièces jointes	Ce gestionnaire de données traite les documents en pièce jointe de votre message.	Ce gestionnaire de données est nécessaire <i>uniquement</i> si votre document contient des pièces jointes.

Les sections suivantes décrivent la façon dont les composants répertoriés dans le tableau 53 fonctionnent conjointement pour permettre l'échange de documents entre WebSphere Partner Gateway et InterChange Server via le protocole HTTP.

### Envoi de documents à ICS via HTTP

Pour que WebSphere Partner Gateway envoie un document à InterChange Server par le biais du protocole HTTP, vous devez utiliser le composant Adapter for HTTP afin de récupérer le document que WebSphere Partner Gateway a envoyé sous la forme d'un flux HTTP. L'adaptateur route ensuite le document vers InterChange Server. La figure 31 illustre comment WebSphere Partner Gateway envoie des documents à InterChange Server via le protocole HTTP.

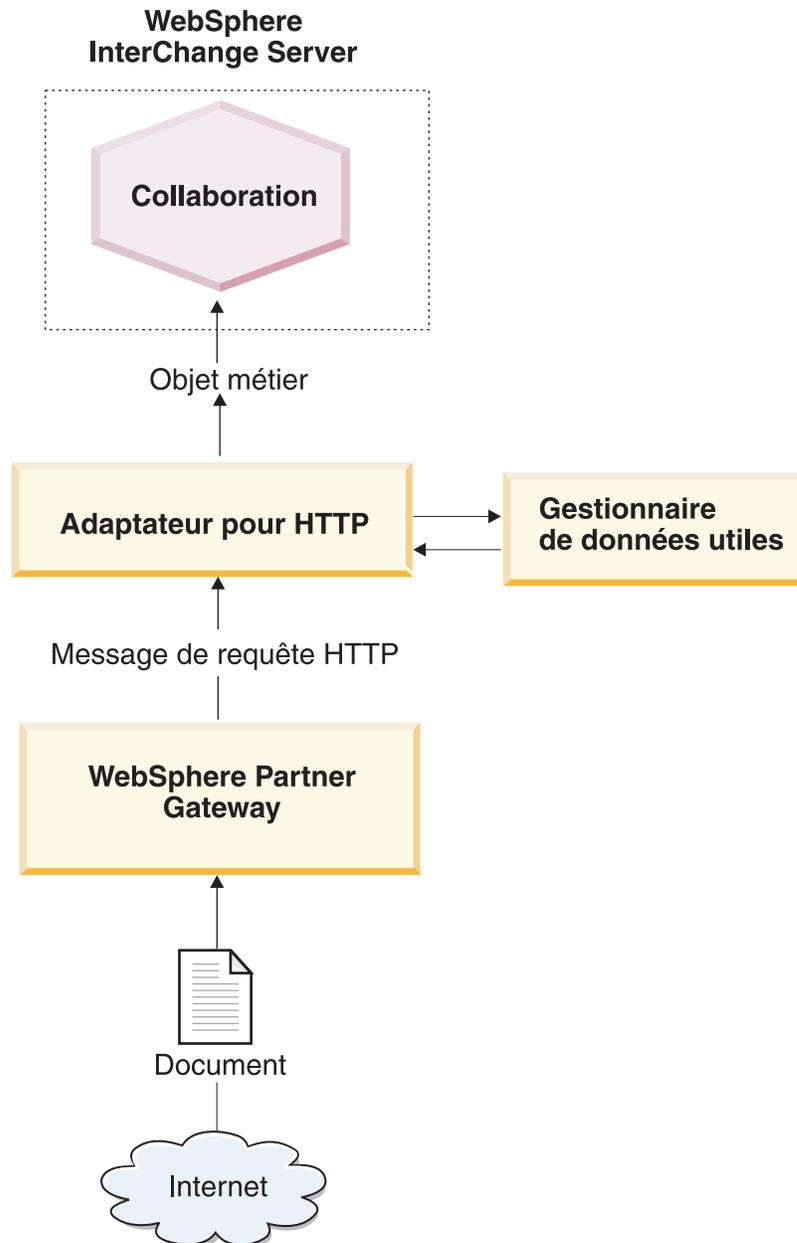


Figure 31. Flux de messages entre WebSphere Partner Gateway et une collaboration via HTTP

### Réception de documents à partir d'ICS via HTTP

Pour que WebSphere Partner Gateway puisse recevoir un document issu d'InterChange Server par le biais du protocole HTTP, vous devez utiliser le composant Adapter for HTTP qui envoie le message reçu depuis InterChange Server sous la forme d'un flux HTTP que WebSphere Partner Gateway peut extraire. La figure 32 illustre comment WebSphere Partner Gateway reçoit des documents à partir d'InterChange Server via le protocole HTTP.

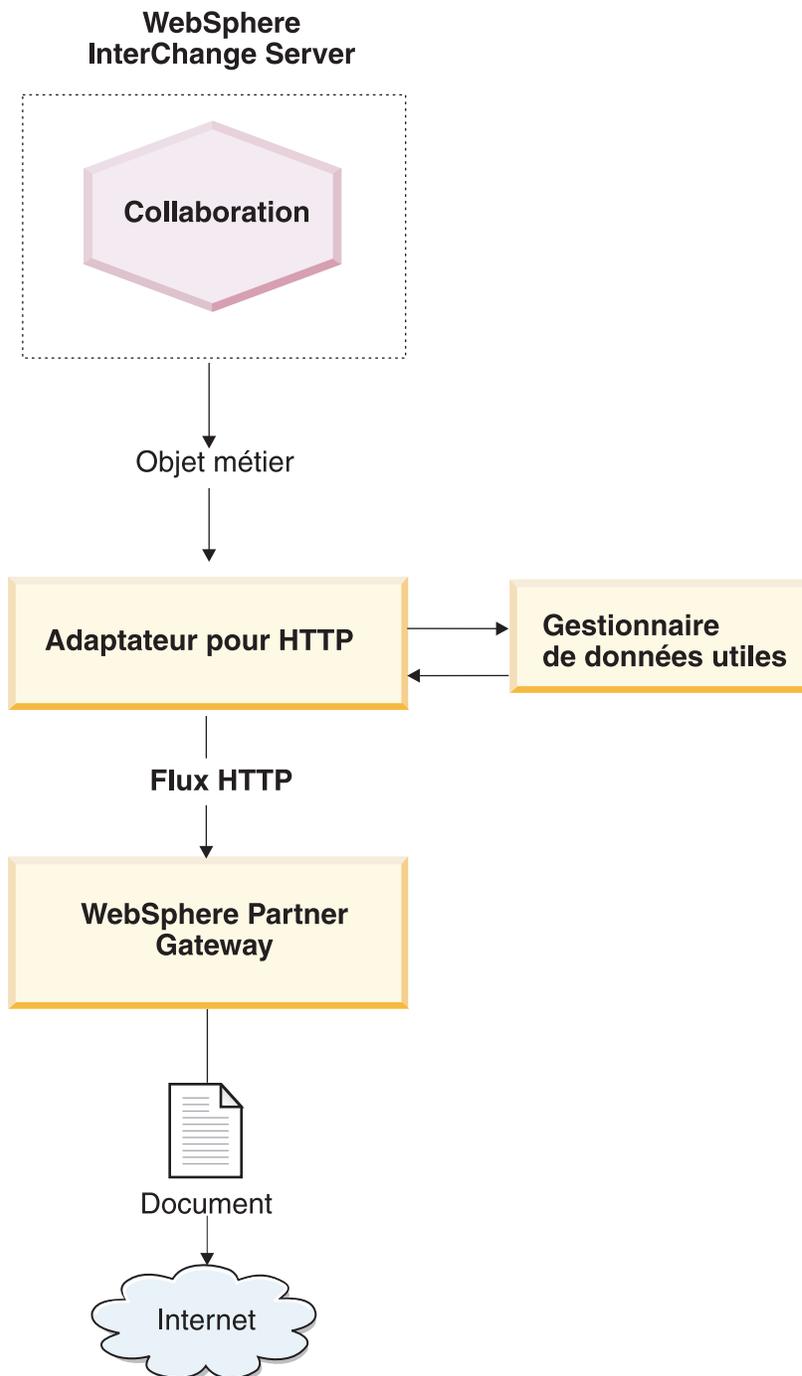


Figure 32. Flux de messages entre une collaboration et WebSphere Partner Gateway via HTTP

## Configuration de l'environnement pour le transport HTTP avec ICS

Etant donné que l'échange de documents sur InterChange Server nécessite l'utilisation d'adaptateurs et de gestionnaires de données, vous devez effectuer les tâches d'installation et de configuration sur Adapter for HTTP. Pour plus d'informations sur la configuration de WebSphere Partner Gateway pour une utilisation avec InterChange Server via HTTP, voir «Configuration de WebSphere Partner Gateway pour InterChange Server», à la page 123.

Adapter for HTTP permet à WebSphere Partner Gateway d'échanger des documents avec InterChange Server sous la forme de messages HTTP. Il prend en charge les interactions suivantes avec InterChange Server :

- Pour le traitement des demandes, il reçoit l'objet métier de type demande à partir d'InterChange Server, le convertit en un flux HTTP et l'envoie à l'URL spécifiée, à partir de laquelle WebSphere Partner Gateway peut le recevoir.
- Pour la notification d'événements, il interroge une URL spécifiée, sur laquelle WebSphere Partner Gateway envoie des documents. Lorsqu'il reçoit un document, il le convertit en objet métier événement (à l'aide d'un gestionnaire de données) et l'envoie à InterChange Server.

**Important :** WebSphere Partner Gateway n'inclut *pas* WebSphere Business Integration Adapter for HTTP. Vous devez vous procurer ce produit séparément et l'installer conformément aux instructions figurant dans le document *Adapter for HTTP User Guide*. Reportez-vous à la documentation relative à l'adaptateur pour vérifier que la version de l'adaptateur est compatible avec la version d'InterChange Server que vous utilisez.

Lorsque vous avez configuré Adapter for HTTP pour communiquer avec InterChange Server, suivez les étapes décrites dans ces sections afin de configurer cet adaptateur pour qu'il soit à l'écoute des messages HTTP envoyés par WebSphere Partner Gateway :

### Spécification du gestionnaire de données utiles

Comme l'illustre la figure 32, à la page 154, Adapter for HTTP utilise un gestionnaire de données pour convertir les objets métier reçus depuis InterChange Server dans un flux HTTP approprié.

**Remarque :** Le gestionnaire de données que le composant Adapter for HTTP appelle convertit les données utiles du document. Si votre document est inclus dans une enveloppe XML (car il comprend des pièces jointes ou l'indicateur d'enveloppe est sur Oui), configurer le gestionnaire de données de pièces jointes de sorte qu'il fonctionne comme un gestionnaire de données utiles. Pour plus d'informations, voir «Gestion des documents contenant des pièces jointes», à la page 132.

Pour indiquer quel gestionnaire de données utiliser pour convertir les données utiles, vous devez suivre les étapes spécifiées dans la section «Conversion d'objet métier», à la page 129. Par ailleurs, vous devez configurer le composant Adapter for HTTP pour utiliser ce gestionnaire de données utiles. Vous pouvez définir le gestionnaire de données utiles de l'une des façons suivantes :

- Dans Connector Configurator, définissez la propriété de configuration du connecteur `DataHandlerMetaObjectName` pour spécifier le nom du méta-objet du gestionnaire de données de niveau supérieur que le composant Adapter for HTTP utilise pour identifier les gestionnaires de données. Veillez à ajouter le

méta-objet du gestionnaire de données de niveau supérieur dans la liste des objets métier pris en charge pour l'adaptateur.

- Dans l'objet métier de niveau supérieur, utilisez l'attribut `MimeType` pour contenir le type MIME permettant d'identifier le gestionnaire de données utiles. Pour plus d'informations sur cet objet métier, voir «Objet métier de niveau supérieur», à la page 159.

## Configuration du nom des packages du gestionnaire de protocole

Le composant Adapter for HTTP utilise la propriété de configuration du connecteur `JavaProtocolHandlerPackages` pour identifier le nom des packages du gestionnaire de protocole Java. Pour permettre l'intégration avec WebSphere Partner Gateway, assurez-vous que la propriété `JavaProtocolHandlerPackage` est définie sur sa valeur par défaut :

```
com.ibm.net.ssl.internal.www.protocol
```

## Configuration du programme d'écoute du protocole HTTP

Le composant Adapter for HTTP prend en charge les propriétés de configuration hiérarchiques pour obtenir les informations dont il a besoin pour configurer ses programmes d'écoute de protocole. La propriété de configuration de niveau supérieur s'appelle `ProtocolListenerFramework`. Cette propriété de niveau supérieur comporte plusieurs niveaux de sous-propriétés. Pour configurer les gestionnaires de protocole en vue de leur utilisation avec le composant Adapter for HTTP, assurez-vous que les propriétés sont configurées dans la propriété `ProtocolListener`, comme décrit dans la procédure suivante :

1. Configurez un programme d'écoute de protocole avec des sous-propriétés situées sous la propriété de configuration suivante :

```
ProtocolListenerFramework
  ProtocolListeners
    HttpListener1
```

Pour configurer votre programme d'écoute de protocole, définissez les sous-propriétés répertoriées dans le tableau 54.

Tableau 54. Configuration du programme d'écoute de protocole

Propriété	Description	Valeur
Protocol	Type de programme d'écoute de protocole : <ul style="list-style-type: none"> <li>• HTTP</li> <li>• HTTPS</li> </ul>	http ou https
Host	Adresse IP sur laquelle le programme d'écoute de protocole est à l'écoute	Adresse IP de l'ordinateur local sur lequel est exécuté WebSphere Partner Gateway
Port	Port sur lequel le programme d'écoute est à l'écoute des demandes	8080

2. Configurez les configurations d'URL que le programme d'écoute de protocole prend en charge avec des sous-propriétés situées en dessous de la propriété de configuration suivante :

```
ProtocolListenerFramework
  ProtocolListeners
    HttpListener1
      URLsConfiguration
        URL1
```

Définissez la propriété `ContextPath` en lui affectant comme valeur l'identificateur URI des requêtes HTTP que le programme d'écoute de protocole reçoit.

**Remarque :** Ce répertoire doit correspondre à celui qui est spécifié par WebSphere Partner Gateway comme identificateur URI du récepteur. Pour plus d'informations, voir «Configuration pour l'envoi de documents à ICS via le protocole de transport HTTP», à la page 123.

- Si votre document contient des pièces jointes, vous devez configurer une transformation pour le programme d'écoute de protocole en définissant des sous-propriétés pour la propriété de configuration suivante :

```

ProtocolListenerFramework
  ProtocolListeners
    HttpListener1
      URLsConfiguration
        URL1
          TransformationRules
            TransformationRule1
  
```

Pour configurer la transformation des pièces jointes pour votre programme d'écoute de protocole, définissez les sous-propriétés répertoriées dans le tableau 55. Vous avez besoin d'une règle de transformation pour chaque instance du gestionnaire de données de pièces jointes que vous utilisez. Pour plus d'informations sur gestionnaire de données de pièces jointes, voir «Gestion des documents contenant des pièces jointes», à la page 132.

Tableau 55. Configuration de la transformation des pièces jointes pour le programme d'écoute de protocole

Propriété	Description	Valeur
<code>ContentType</code>	Type de contenu des données à transformer à l'aide d'un gestionnaire de données	Type de contenu associé aux données relatives aux pièces jointe
<code>MimeType</code>	Type MIME à utiliser pour identifier le gestionnaire de données à appeler	Type MIME associé à l'instance du gestionnaire de données de pièces jointes
<code>Charset</code>	Jeu de caractères à utiliser lors de la transformation des données du type de contenu spécifié	Jeu de caractères pour les données relatives aux pièces jointes

Pour plus d'informations sur ces propriétés, voir le document *Adapter for HTTP User Guide*.

## Création de définitions d'objet métier pour ICS via HTTP

Le composant *Adapter for HTTP* envoie votre document à *InterChange Server* et le reçoit sous la forme d'un objet métier données utiles. *Adapter for HTTP* appelle le gestionnaire de données utiles pour gérer cet objet métier lorsqu'il reçoit ou envoie un document de *WebSphere Partner Gateway*, comme suit :

- Pour le traitement des demandes, le gestionnaire de données utiles convertit l'objet métier de type demande en son flux HTTP correspondant.
- Pour la notification d'événements, le gestionnaire de données convertit le flux HTTP en objet métier événement.

Par conséquent, vous devez créer des définitions d'objet métier comme illustré dans le tableau 56 pour représenter la structure d'objet métier données utiles que le composant *Adapter for HTTP* attend.

Tableau 56. Définitions d'objets métier pour Adapter for HTTP

Condition	Définition d'objet métier	Pour plus d'informations
Si vous utilisez l'empaquetage None ou Backend Integration pour votre message <i>et</i> si vos documents ne possèdent <i>pas</i> de pièces jointes	Objet métier données utiles : <ul style="list-style-type: none"> <li>• objet métier de niveau supérieur</li> <li>• objet métier de type demande</li> <li>• un objet métier de type réponse (facultatif)</li> <li>• un objet métier de type erreur (facultatif)</li> </ul>	«Création de la structure d'objet métier données utiles pour ICS via HTTP»
Si vous utilisez l'empaquetage Backend Integration pour votre message.	Ajoutez à l'objet métier données utiles les objets métier destinés à contenir les informations d'en-tête du message : <ul style="list-style-type: none"> <li>• Méta-objet dynamique</li> <li>• Objet métier Propriétés HTTP</li> </ul>	«Création d'informations d'en-tête de niveau transport HTTP pour ICS», à la page 162.
Si le document contient des pièces jointes	Vous devez également créer des objets métier supplémentaires pour représenter les pièces jointes.	«Création de définitions d'objets métier liées aux pièces jointes», à la page 144

### Création de la structure d'objet métier données utiles pour ICS via HTTP

Adapter for HTTP attend une structure d'objet métier données utiles qui comprend les objets métier suivants :

- un objet métier de niveau supérieur ;
- un objet métier de type demande ;
- un objet métier de type erreur (facultatif) ;
- un objet métier de type réponse (facultatif).

La figure 33 illustre un exemple de structure d'objet métier correspondant à la définition d'un objet métier données utiles à utiliser avec InterChange Server via le protocole HTTP.

**Remarque :** Pour obtenir une description détaillée de cette structure d'objet métier, voir le document *Adapter for HTTP User Guide*.

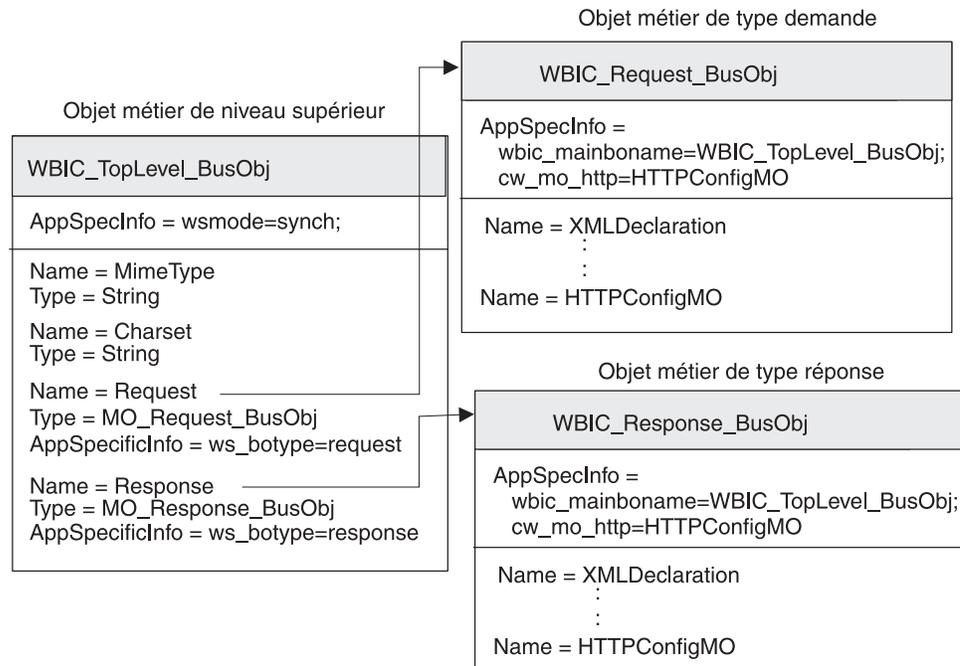


Figure 33. Structure d'objet métier pour l'objet métier données utiles HTTP pour ICS

**Objet métier de niveau supérieur:** L'objet métier de niveau supérieur est un encapsuleur pour les objets métier de type demande et de type réponse. Vous devez créer une définition d'objet métier pour cet objet métier. Le tableau 57 contient les attributs de la définition de l'objet métier de niveau supérieur.

Tableau 57. Attributs de l'objet métier de niveau supérieur

Attribut	Type d'attribut	Description
MimeType	Chaîne	Définit le type de contenu et le format des données qui sont transmises à l'URL.
Charset	Chaîne	Utilisé pour déterminer quel gestionnaire de données appeler.
Request	Objet métier	Objet métier enfant qui représente le message de demande. L'objectif de cet objet métier dépend de sa participation ou non au traitement des demandes ou à la notification des événements. Pour plus d'informations sur la structure de l'objet métier, voir «Objet métier de type demande», à la page 160.
Response	Objet métier	Objet métier enfant qui représente le message de réponse (si vous attendez une réponse). L'objectif de cet objet métier dépend de sa participation ou non au traitement des demandes ou à la notification des événements. Pour plus d'informations sur la structure de l'objet métier, voir «Objet métier de type réponse», à la page 161.

**Remarque :** Lorsque vous utilisez le composant Adapter for HTTP avec WebSphere Partner Gateway, vous n'avez *pas* besoin d'inclure les objets métier de type erreur dans votre objet métier de niveau supérieur.

Le tableau 58 récapitule les informations propres à l'application que la définition d'objet métier de niveau supérieur peut contenir.

Tableau 58. Informations propres à l'application relatives à la définition d'objet métier de niveau supérieur

Informations propres à l'application	Balise	Description
Niveau objet métier	ws_mode	Définit si l'interaction est synchrone ou asynchrone
Niveau attribut	ws_botype	Définit l'attribut qui contient l'objet métier de type demande ou de type réponse

Pour obtenir une description complète de la structure de l'objet métier de niveau supérieur et les informations propres à l'application qui lui sont associées, voir le document *Adapter for HTTP User Guide*.

**Objet métier de type demande :** L'objet métier de type demande contient les données à transmettre à l'URL. Il représente le message de requête HTTP. L'objectif de cet objet métier de type demande dépend de la tâche à laquelle InterChange Server participe, par exemple :

- Pour la notification d'événements (envoi d'un document à InterChange Server), l'objet métier de type demande contient le message de requête émis par WebSphere Partner Gateway, qui correspond à l'événement à envoyer à InterChange Server.
- Pour le traitement des demandes (réception d'un document depuis InterChange Server), l'objet métier de type demande contient la demande qu'InterChange Server effectue auprès de WebSphere Partner Gateway.

**Remarque :** L'objet métier de niveau supérieur identifie ses objets métier enfant comme des objets métier de type demande et de type réponse. Toutefois, cette structure est utilisée *à la fois* dans le traitement des demandes et la notification d'événements.

Pour obtenir une description de la structure de l'objet métier de type demande, voir le document *Adapter for HTTP User Guide*. Pour l'utiliser avec WebSphere Partner Gateway, vous devez effectuer deux personnalisations de la structure de la définition de l'objet métier de type demande :

- Si le document que WebSphere Partner Gateway envoie à InterChange Server utilise l'emballage Backend Integration, vous devez ajouter à la définition de l'objet métier de type demande un attribut spécifique permettant d'identifier le méta-objet de configuration du protocole HTTP.  
Cet attribut contient des informations de configuration pour les en-têtes de transport du message. Pour plus d'informations, voir «Création d'informations d'en-tête de niveau transport HTTP pour ICS», à la page 162.
- Ajoutez les balises indiquées dans le tableau 59 aux informations propres à l'application niveau objet métier qui concernent la définition de l'objet métier de type demande.

Tableau 59. Balises contenues dans les informations d'application sur l'objet métier de type demande

Balise des informations propres à l'application	Description	Obligatoire
ws_tloname	Donne le nom de l'objet métier de niveau supérieur	Obligatoire uniquement si la définition d'objet métier participe à la notification d'événements
cw_mo_http	Indique le méta-objet de configuration de protocole HTTP, qui contient les zones d'en-tête de transport HTTP. Pour plus d'informations, voir «Création d'informations d'en-tête de niveau transport HTTP pour ICS», à la page 162.	Obligatoire uniquement si vous utilisez l'empaquetage Backend Integration

**Remarque :** Si vous utilisez le gestionnaire de données de pièces jointes pour traiter des documents encapsulés dans une enveloppe de transport XML, vous devez modifier votre objet métier de type demande pour qu'il contienne les pièces jointes, comme décrit dans «Création de définitions d'objets métier liées aux pièces jointes», à la page 144.

**Objet métier de type réponse :** L'objet métier de type réponse contient les données à recevoir depuis l'URL. Il contient les attributs des différentes balises XML utilisés dans le message de réponse. L'objectif de cet objet métier de type réponse dépend de la tâche à laquelle InterChange Server participe, par exemple :

- Pour la notification d'événements, l'objet métier de type réponse contient le message de réponse qui est envoyé par la collaboration dans InterChange Server.
- Pour le traitement des demandes, l'objet métier de type réponse contient les informations transmises par WebSphere Partner Gateway en réponse à la demande envoyée par InterChange Server.

Que la réponse ait lieu dans le cadre du traitement des demandes ou de la notification d'événements, un objet métier de type réponse est envoyé *uniquement* si l'échange entre WebSphere Partner Gateway et InterChange Server est *synchrone* et que cet objet métier est attendu en réponse à votre demande.

Pour obtenir une description de la structure de l'objet métier de type erreur, reportez-vous au document *Adapter for HTTP User Guide*. Pour l'utiliser avec WebSphere Partner Gateway, vous devez effectuer des personnalisations de la structure de la définition de l'objet métier de type demande :

- Si le document que WebSphere Partner Gateway envoie à InterChange Server utilise l'empaquetage Backend Integration, vous devez ajouter à la définition de l'objet métier de type réponse un attribut spécifique permettant d'identifier le méta-objet de configuration du protocole HTTP.

Cet attribut contient des informations de configuration pour les en-têtes de transport du message. Pour plus d'informations, voir «Création d'informations d'en-tête de niveau transport HTTP pour ICS», à la page 162.

- Ajoutez les balises indiquées dans le tableau 59 aux informations propres à l'application au niveau de l'objet métier concernant la définition de l'objet métier de type réponse.

- Dans l'objet métier de niveau supérieur, ajoutez la balise ws\_botype aux informations d'application au niveau de l'attribut pour l'attribut qui correspond à l'objet métier de type réponse.

La syntaxe de cette balise est la suivante :

ws\_botype=response

Si l'échange entre WebSphere Partner Gateway et InterChange Server est *asynchrone*, WebSphere Partner Gateway n'attend *pas* de réponse et vous n'avez donc pas besoin de créer un objet métier de type réponse.

## Création d'informations d'en-tête de niveau transport HTTP pour ICS

Si vous envoyez des documents avec l'empaquetage Backend Integration via le protocole de transport HTTP, votre objet métier de type demande doit contenir des informations d'en-tête de niveau transport personnalisé. Le composant Adapter for HTTP s'attend à ce que les informations d'en-tête personnalisé soient contenues dans un *méta-objet dynamique*.

La figure 34 illustre la structure de l'objet métier définie pour un objet métier de type demande qui représente un document WebSphere Partner Gateway utilisant l'empaquetage Backend Integration via le protocole HTTP.

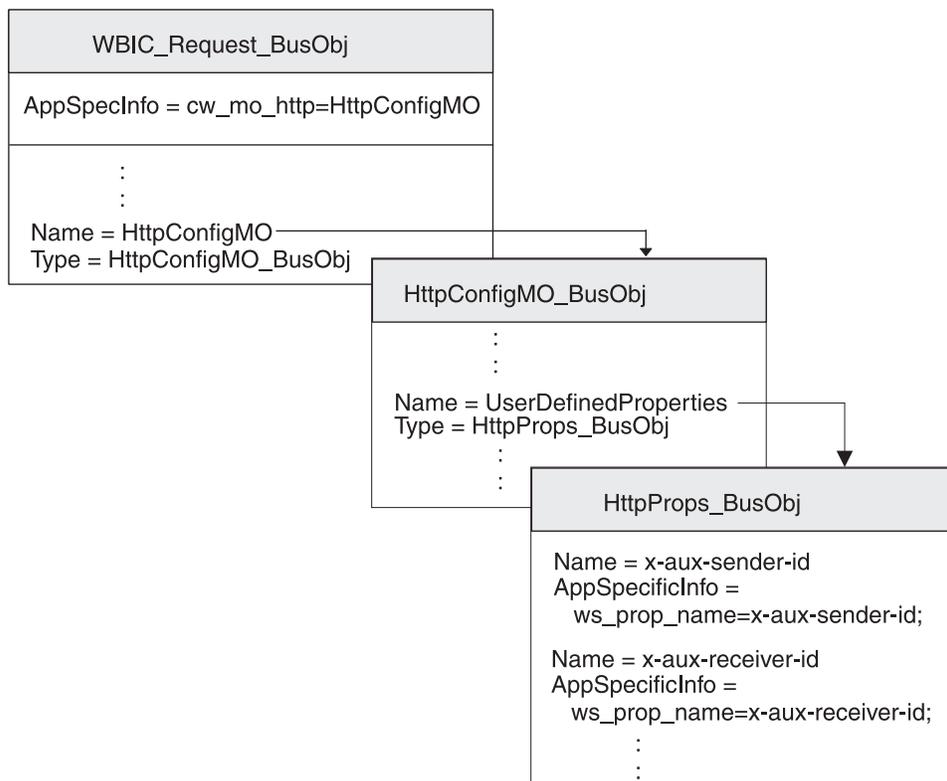


Figure 34. Relation entre l'objet métier de type demande et le méta-objet de configuration de protocole HTTP

Assurez-vous que la structure de l'objet métier inclut un méta-objet de configuration de protocole HTTP en procédant comme suit :

1. Créez une définition d'objet métier destinée à contenir les propriétés HTTP requises par l'empaquetage Backend Integration.

2. Créez une définition d'objet métier pour le méta-objet de configuration de protocole HTTP.
3. Modifiez la définition d'objet métier pour votre objet métier de type demande afin d'inclure un attribut pour le méta-objet de configuration de protocole HTTP.

Chaque étape est décrite dans les sections ci-dessous.

**Création de l'objet métier propriétés définies par l'utilisateur :** Le composant Adapter for HTTP prend en charge un *objet métier propriétés définies par l'utilisateur* qui contient les propriétés personnalisées dans le méta-objet de configuration de protocole HTTP. WebSphere Partner Gateway utilise cet objet métier pour inclure les propriétés HTTP requises par l'empaquetage Backend Integration. Il peut également contenir l'attribut Content-Type, qui spécifie l'en-tête du type de contenu à définir dans le message de requête ainsi que l'attribut de longueur du contenu qui détermine la longueur du message, en octets. Le tableau 8, à la page 27 décrit chaque zone valide de l'en-tête de transport.

Pour créer une définition d'objet métier propriétés définies par l'utilisateur pour les zones d'en-tête HTTP, procédez comme suit :

1. Créez un attribut au sein de la définition de l'objet métier pour chaque zone d'en-tête de transport.

Tous les attributs doivent comporter un type d'attribut Chaîne. Vous pouvez affecter à l'attribut le même nom que la propriété HTTP (comme indiqué dans la colonne de la zone En-tête du tableau 8, à la page 27).

2. Pour chacun des attributs contenus dans l'objet métier Propriétés HTTP, ajoutez les informations propres à l'application qui permettent d'identifier l'objectif de l'attribut associé.

Ces informations propres à l'application au niveau de l'attribut ont le format suivant :

```
ws_prop_name=HTTPproperty
```

où *HTTPproperty* est l'une des valeurs contenues dans la colonne zone En-tête du tableau 8, à la page 27.

Dans la figure 34, à la page 162, la définition de l'objet métier HttpProps\_BusObj contient des attributs pour les différentes zones d'en-tête de transport. Ces attributs possèdent tous des informations propres à l'application au niveau des attributs afin de spécifier le nom de l'en-tête du protocole associé. Par exemple, les informations propres à l'application de l'attribut x-aux-sender-id sont définies comme suit :

```
ws_prop_name=x-aux-sender-id
```

**Création du méta-objet de configuration de protocole HTTP :** Pour la notification d'événements, l'objet métier de type demande, réponse ou erreur peut contenir un méta-objet dynamique appelé *méta-objet de configuration de protocole HTTP* et destiné à contenir les informations de configuration (telles que les informations d'en-tête).

Pour obtenir une description de la structure de l'objet métier de configuration de protocole HTTP, reportez-vous au document *Adapter for HTTP User Guide*. Pour l'utiliser avec WebSphere Partner Gateway, vous devez effectuer les personnalisations suivantes de la structure de la définition de l'objet métier de configuration de protocole HTTP :

1. Créez un attribut au sein de la définition de l'objet métier pour chaque zone requise.

Tous les attributs doivent comporter un type d'attribut Chaîne.

**Remarque :** Pour obtenir la liste complète des attributs contenus dans le méta-objet de configuration du protocole HTTP, voir le document *Adapter for HTTP User Guide*.

2. Ajoutez l'attribut `UserDefinedProperties` à cette définition d'objet métier.  
Le type de cet attribut est la définition de l'objet métier propriétés définies par l'utilisateur (voir «Création de l'objet métier propriétés définies par l'utilisateur», à la page 163).

Par exemple, dans la figure 34, à la page 162, la définition de l'objet métier `HttpConfigM0_BusObj` contient l'attribut `UserDefinedProperties`, de type `HttpProps_BusObj`.

**Modification de la définition de l'objet métier de type demande :** La définition de l'objet métier de type demande représente les informations demandées à WebSphere Partner Gateway. Pour plus d'informations sur la création d'un objet métier de type demande, voir «Objet métier de type demande», à la page 160. Pour intégrer le méta-objet dynamique à votre structure d'objet métier données utiles, vous devez effectuer les modifications suivantes dans la définition de l'objet métier de type demande :

1. Ajoutez un attribut à votre définition d'objet métier de type demande afin qu'il contienne le méta-objet de configuration de protocole HTTP.

Le type de cet attribut est la définition d'objet métier du méta-objet de configuration de protocole HTTP (voir «Création du méta-objet de configuration de protocole HTTP», à la page 163).

2. Ajoutez la balise `cw_mo_http` aux informations propres à l'application au niveau de l'objet métier de votre définition d'objet métier de type demande afin d'identifier l'attribut qui contient le méta-objet de configuration de protocole HTTP.

La balise `cw_mo_http` a le format suivant :

```
cw_mo_http=AttrMétaObjConfigHttp
```

où `AttrMétaObjConfigHttp` est le nom de l'attribut, dans l'objet métier de type demande, qui contient le méta-objet de configuration de protocole HTTP.

Par exemple, dans la figure 34, à la page 162, un attribut nommé `HttpConfigM0` a été ajouté à la définition d'objet métier de type demande, `hub_HttpRequest_BusObj`. Cet attribut contient le méta-objet dynamique, qui est un objet métier enfant du type `HttpConfigM0_BusObj`. Par ailleurs, les informations propres à l'application de l'objet métier de type demande ont été modifiées pour inclure la balise `cw_mo_http` suivante afin d'identifier ce méta-objet dynamique :

```
cw_mo_http=HttpConfigM0
```

## Création d'artefacts ICS pour HTTP

Pour configurer un InterChange Server pour qu'il communique avec WebSphere Partner Gateway via le protocole HTTP, vous devez créer les artefacts InterChange Server indiqués dans le tableau 60.

Tableau 60. Artefacts pour la communication avec ICS via le protocole HTTP

Artefact ICS	Objet	Pour plus d'informations
Définitions d'objet métier	Représente le document	«Création de définitions d'objet métier pour ICS via HTTP», à la page 157

Tableau 60. Artefacts pour la communication avec ICS via le protocole HTTP (suite)

Artefact ICS	Objet	Pour plus d'informations
Objet connecteur	Représente le composant Adapter for HTTP au moment de l'exécution	«Création de l'objet connecteur HTTP»
Modèle et objet collaboration	Représente le processus métier qu'InterChange Server utilise pour traiter le document	«Liaison de collaborations pour communiquer avec Adapter for HTTP»

## Création de l'objet connecteur HTTP

Pour obtenir une instance du composant Adapter for HTTP au moment de l'exécution, exécutez la procédure suivante dans l'outil System Manager :

1. Créez les objets de connecteur :
  - Créez un objet connecteur afin de représenter une instance du composant Adapter for HTTP.

**Remarque :** Dans l'onglet des objets métier pris en charge de Connector Configurator, vérifiez que vous avez indiqué toutes les définitions d'objet métier que vous avez créées afin de les utiliser avec Adapter for HTTP. Pour obtenir la description de ces définitions d'objet métier, voir «Création de définitions d'objet métier pour ICS via HTTP», à la page 157.

- Si cela est requis par votre collaboration, créez un objet connecteur pour le connecteur de port.
2. Configurez les objets de connecteur.
 

Pour plus d'informations sur la manière de configurer l'objet connecteur du composant Adapter for HTTP afin de l'utiliser avec WebSphere Partner Gateway, voir «Configuration de l'environnement pour le transport HTTP avec ICS», à la page 155.

## Liaison de collaborations pour communiquer avec Adapter for HTTP

Comme décrit dans «Création des collaborations», à la page 131, un objet collaboration doit exister au moment de l'exécution pour qu'InterChange Server puisse savoir où recevoir et envoyer des objets métier. Lorsque vous créez l'objet collaboration pour la collaboration qui utilise le composant Adapter for HTTP pour échanger des informations avec WebSphere Partner Gateway, vous devez lier les ports de collaboration, comme suit :

- Pour le traitement des demandes, affectez au port "cible", qui envoie des demandes à WebSphere Partner Gateway, l'objet connecteur que vous avez créé pour le composant Adapter for HTTP ; à savoir, l'adaptateur de *destination*.
- Pour la notification d'événements, affectez au port "source", qui reçoit les événements à partir de WebSphere Partner Gateway, l'objet connecteur que vous avez créé pour le composant Adapter for HTTP ; à savoir, l'adaptateur *source*.

---

## Envoi de documents SOAP via HTTP/S

Les documents SOAP diffèrent des autres types de documents échangés via HTTP/S. Ils utilisent le composant standard Adapter for Web Services, qui appelle le gestionnaire de données SOAP dans le but de convertir les messages SOAP en objets métier, et inversement. Cette section explique comment échanger des documents SOAP entre WebSphere Partner Gateway et WebSphere InterChange Server via le protocole de transport HTTP.

**Remarque :** Pour échanger des documents autres que SOAP entre WebSphere Partner Gateway et WebSphere InterChange Server via le protocole de transport HTTP, voir «Utilisation du protocole de transport HTTP avec ICS», à la page 151.

Pour plus d'informations sur la structure de l'objet métier et sur WSDL Object Discovery Agent (ODA), un outil de conception qui permet de créer des objets métier SOAP contenant des informations sur les services Web du récepteur, voir la documentation du composant Adapter for Web Services.

Comme l'explique le *Guide de configuration du concentrateur WebSphere Partner Gateway*, vous devez configurer un récepteur pour recevoir les appels du service Web depuis un système dorsal (récepteur des services Web), ainsi qu'un récepteur pour recevoir les appels du service Web depuis un partenaire externe (récepteur externe des services Web).

## Composants requis pour l'envoi et la réception de documents

Pour envoyer un document SOAP depuis WebSphere Partner Gateway vers InterChange Server via le protocole HTTP, vous devez utiliser le composant contenu dans le tableau 61.

Tableau 61. Composants requis pour envoyer des documents SOAP à InterChange Server via HTTP

Composant	Description	Remarques et limitations
WebSphere Business Integration Adapter for Web Services	Cet adaptateur permet à InterChange Server d'échanger des objets métier avec des applications qui envoient ou reçoivent des données sous la forme de messages HTTP.	<ol style="list-style-type: none"><li>1. Cet adaptateur <i>ne peut pas</i> être utilisé avec des documents autres que SOAP.</li><li>2. Assurez-vous que vous utilisez le composant Adapter for Web Services version 3.4.0 (ou ultérieure). Voir le document <i>Adapter for Web Services User Guide</i> pour vous assurer que le niveau de l'adaptateur est compatible avec la version de WebSphere InterChange Server que vous utilisez.</li></ol>

**Remarque :** Si un document SOAP contient des pièces jointes, vous n'avez pas besoin d'utiliser le gestionnaire de données de pièces jointes pour les traiter.

## Fonctionnement de l'appel des services Web par les partenaires externes

Les opérations suivantes ont lieu lorsqu'un partenaire externe envoie une demande concernant une collaboration exposée en tant que service Web fourni par le partenaire interne :

1. Le partenaire externe envoie un message de requête SOAP à la destination indiquée dans le document WSDL généré pour la collaboration. Notez que le noeud final indiqué dans le document WSDL correspond au récepteur des services Web (URL) de WebSphere Partner Gateway, et non au noeud final réel.
2. WebSphere Partner Gateway reçoit et route le message vers le composant Adapter for Web Services.
3. Le composant Adapter for Web Services envoie le message SOAP au gestionnaire de données SOAP dans le but de convertir le message SOAP en objet métier. L'adaptateur appelle la collaboration affichée en tant que service Web.
4. S'il s'agit d'une opération de demande/réponse, la collaboration renvoie un objet métier SOAP de type réponse (ou erreur).

5. Si la collaboration renvoie un objet métier SOAP de type réponse (ou erreur), le composant Adapter for Web Services appelle le gestionnaire de données SOAP pour qu'il convertisse l'objet métier SOAP de type réponse (ou erreur) en message de réponse SOAP. L'adaptateur renvoie la réponse à WebSphere Partner Gateway. Si la collaboration n'a pas renvoyé d'objet métier SOAP de type réponse (ou erreur), le composant Adapter for Web Services renvoie le code d'état de réponse HTTP approprié.
6. WebSphere Partner Gateway route la réponse vers le service Web.

## Fonctionnement de l'appel des services Web par le partenaire interne

Le document WSDL public fourni par WebSphere Partner Gateway peut être utilisé pour créer des objets métier à l'aide de l'agent ODA WSDL. Il est important de noter que lorsque le service Web est fourni par un partenaire externe et utilisé par le partenaire interne, l'URL public utilisé par ce dernier pour appeler le service Web doit contenir la chaîne de requête suivante :

*?to=<External Partner Web Service Provider's business ID>*

Par exemple, l'adresse suivante informe WebSphere Partner Gateway que le fournisseur de ce service Web est le partenaire dont l'ID entreprise est 123456789 :  
[http://<Adresse\\_IP\\_concentrateur>/bcgreceiver/Receiver?to=123456789](http://<Adresse_IP_concentrateur>/bcgreceiver/Receiver?to=123456789)

Le composant WSDL ODA n'ajoutera pas la chaîne de demande à la valeur par défaut de l'attribut de l'URL de l'objet métier de niveau supérieur du service Web.

Les opérations suivantes ont lieu lorsqu'une collaboration envoie une demande (à Adapter for Web Services) pour appeler le service Web d'un partenaire externe :

1. La collaboration envoie une demande d'appel à l'adaptateur, qui appelle le gestionnaire de données SOAP pour qu'il convertisse l'objet métier en message de requête SOAP.
2. L'adaptateur appelle le service Web en envoyant le message SOAP au récepteur externe des services Web (URL) sur WebSphere Partner Gateway.
3. WebSphere Partner Gateway agit en tant que proxy en envoyant le message SOAP vers le noeud final correspondant au service Web (partenaire externe) de la destination. Cette opération appelle le service Web.
4. Le service Web appelé reçoit le message de requête SOAP et effectue le traitement demandé.
5. Le service Web appelé envoie un message de réponse (ou d'erreur) SOAP. Dans le cas d'une opération à sens unique, un code d'état HTTP est renvoyé.
6. S'il s'agit d'un service Web de demande/réponse, WebSphere Partner Gateway route le message de réponse (ou d'erreur) SOAP vers l'adaptateur, qui à son tour appelle le gestionnaire de données pour convertir ce message en objet métier de type réponse ou erreur. Le connecteur renvoie l'objet métier SOAP de type réponse ou erreur à la collaboration.



---

## Chapitre 10. Intégration à InterChange Server via JMS

Ce chapitre décrit le processus d'intégration de WebSphere Partner Gateway à WebSphere InterChange Server via le protocole de transport JMS. Il fournit des informations relatives à la configuration d'InterChange Server et des adaptateurs nécessaires à la communication via JMS.

**Remarque :** Pour plus d'informations sur la manière de configurer WebSphere Partner Gateway pour communiquer avec InterChange Server via JMS, voir «Configuration de WebSphere Partner Gateway pour InterChange Server», à la page 123. Pour obtenir des informations générales sur la configuration d'InterChange Server, voir «Configuration d'InterChange Server», à la page 127.

Ce chapitre explique comment échanger des documents entre WebSphere Partner Gateway et WebSphere InterChange Server via le protocole de transport JMS :

- «Composants requis pour les documents envoyés via JMS»
- «Configuration de l'environnement requis pour le transport JMS», à la page 175
- «Création de définitions d'objet métier pour JMS», à la page 177

---

### Composants requis pour les documents envoyés via JMS

Pour que WebSphere Partner Gateway communique avec InterChange Server à l'aide du protocole de transport JMS, ces composants doivent être configurés de sorte qu'ils puissent fonctionner avec JMS. Le tableau 62 donne un récapitulatif des étapes de configuration à suivre.

*Tableau 62. Configuration de WebSphere Partner Gateway et InterChange Server pour un fonctionnement via le protocole de transport JMS*

Composant	Version	Pour plus d'informations
WebSphere Partner Gateway	6.1	«Configuration pour l'envoi de documents à ICS via le protocole de transport JMS», à la page 124  «Configuration pour la réception de documents depuis ICS via le protocole de transport JMS», à la page 126
WebSphere InterChange Server	4.3	«Création d'artefacts ICS pour JMS», à la page 182

Pour échanger un document entre WebSphere Partner Gateway et InterChange Server via le protocole JMS, vous devez également utiliser les composants répertoriés dans le tableau 63.

Tableau 63. Composants requis pour transférer des documents depuis et vers InterChange Server via JMS

Composant	Description	Remarques et limitations
WebSphere Business Integration Adapter for JMS (Adapter for JMS)	Ce composant permet à InterChange Server d'échanger des objets métier avec des applications qui envoient ou reçoivent des données sous la forme de messages JMS. Adapter for JMS et WebSphere Partner Gateway communiquent par le biais des files d'attente JMS.	Assurez-vous que vous utilisez Adapter for JMS version 2.7.0 (ou ultérieure), qui assure la prise en charge des propriétés d'en-têtes personnalisés. Voir la documentation relative à l'adaptateur pour vérifier que sa version est compatible avec la version d'InterChange Server que vous utilisez.
Un gestionnaire de données utiles	Ce gestionnaire de données convertit les données utiles à partir du format du document en représentation d'objet métier.	Pour plus d'informations, voir «Spécification du gestionnaire de données utiles», à la page 176.
Gestionnaire de données de pièces jointes	Ce gestionnaire de données convertit les documents contenant des pièces jointes en objets métier.	Ce gestionnaire de données est nécessaire <i>uniquement</i> si votre document contient des pièces jointes. Pour plus d'informations, voir «Gestion des documents contenant des pièces jointes», à la page 132.

Les sections suivantes décrivent la façon dont les composants répertoriés dans le tableau 63 fonctionnent conjointement pour permettre l'échange de documents entre WebSphere Partner Gateway et InterChange Server via le protocole JMS.

## Envoi de documents via le transport JMS

Pour que WebSphere Partner Gateway envoie un document à InterChange Server par le biais du protocole JMS, vous devez utiliser Adapter for JMS afin de récupérer le message que WebSphere Partner Gateway a placé dans la file d'attente JMS. L'adaptateur route alors le message vers InterChange Server. La figure 35 illustre comment WebSphere Partner Gateway envoie des documents à InterChange Server via le protocole JMS.

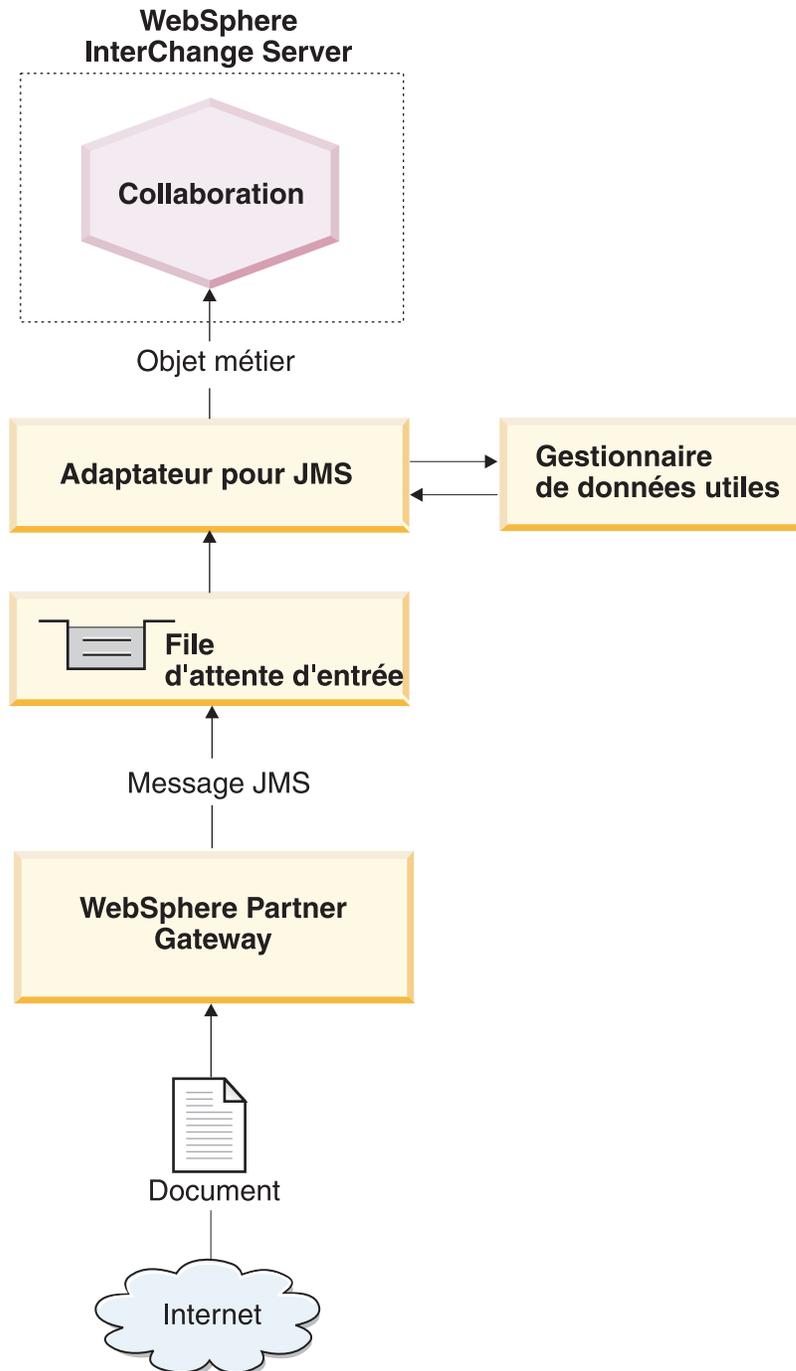


Figure 35. Flux de messages entre WebSphere Partner Gateway et une collaboration via JMS

Les étapes suivantes expliquent comment WebSphere Partner Gateway participe à la notification d'événements en envoyant un document à une collaboration dans InterChange Server via le protocole de transport JMS :

1. WebSphere Partner Gateway place un message dans sa file d'attente de sortie JMS.

Si le type d'emballage du document est Backend Integration, WebSphere Partner Gateway a intégré des propriétés personnalisées dans ce message. L'en-tête de message JMS, JMSType, est défini avec le type de contenu des données utiles.

**Remarque :** Dans WebSphere Partner Gateway, vous devez configurer une destination qui identifie la file d'attente JMS à laquelle WebSphere Partner Gateway envoie le message et dans laquelle Adapter for JMS effectue des interrogations.

2. Lorsque Adapter for JMS repère un message dans l'une de ses files d'attente d'entrée, il le récupère.

La file d'attente JMS que WebSphere Partner Gateway utilise comme file d'attente de sortie est identique à celle utilisée par Adapter for JMS comme file d'attente d'entrée. Pour plus d'informations sur la manière de configurer cette file d'attente, voir «Configuration des files d'attente JMS», à la page 175. Pour plus d'informations sur le traitement d'Adapter for JMS, voir le document *Adapter for JMS User Guide*.

3. Adapter for JMS déplace le message vers sa file d'attente des messages en cours.
4. Adapter for JMS extrait le corps du message JMS et appelle le gestionnaire de données avec le corps du message. Ce gestionnaire de données convertit le corps du message JMS en objet métier.

**Remarque :** Si vos messages contiennent des pièces jointes, vous pouvez installer le gestionnaire de données en pièce jointe puis configurer Adapter for JMS pour qu'il l'appelle afin de convertir le corps du message JMS en objet métier. Pour plus d'informations, voir «Gestion des documents contenant des pièces jointes», à la page 132.

Lorsque le type d'emballage est Backend Integration et que le document contient des pièces jointes, le gestionnaire de données configuré est responsable de la gestion des données utiles et des pièces jointes.

5. Le gestionnaire de données renvoie l'objet métier à Adapter for JMS.

**Remarque :** Si le gestionnaire de données en pièce jointe a été utilisé, cet objet métier contient les données utiles ainsi que les pièces jointes.

6. Si Adapter for JMS récupère un méta-objet enfant dynamique (spécifié à l'aide de l'attribut `cw_mo_conn` contenu dans les informations propres à l'application au niveau de l'objet métier), l'adaptateur remplit les en-têtes JMS définis par l'utilisateur qui sont présents dans l'objet métier avec les en-têtes présents dans le message JMS.
7. Adapter for JMS transmet l'objet métier à InterChange Server en même temps que l'inscription.

## Réception de documents via le transport JMS

Pour que WebSphere Partner Gateway puisse recevoir un document issu d'InterChange Server par le biais du protocole JMS, vous devez utiliser Adapter for JMS qui place le message reçu d'InterChange Server dans une file d'attente JMS destinée à WebSphere Partner Gateway. La figure 36 illustre comment WebSphere Partner Gateway reçoit des documents d'InterChange Server via le protocole JMS.

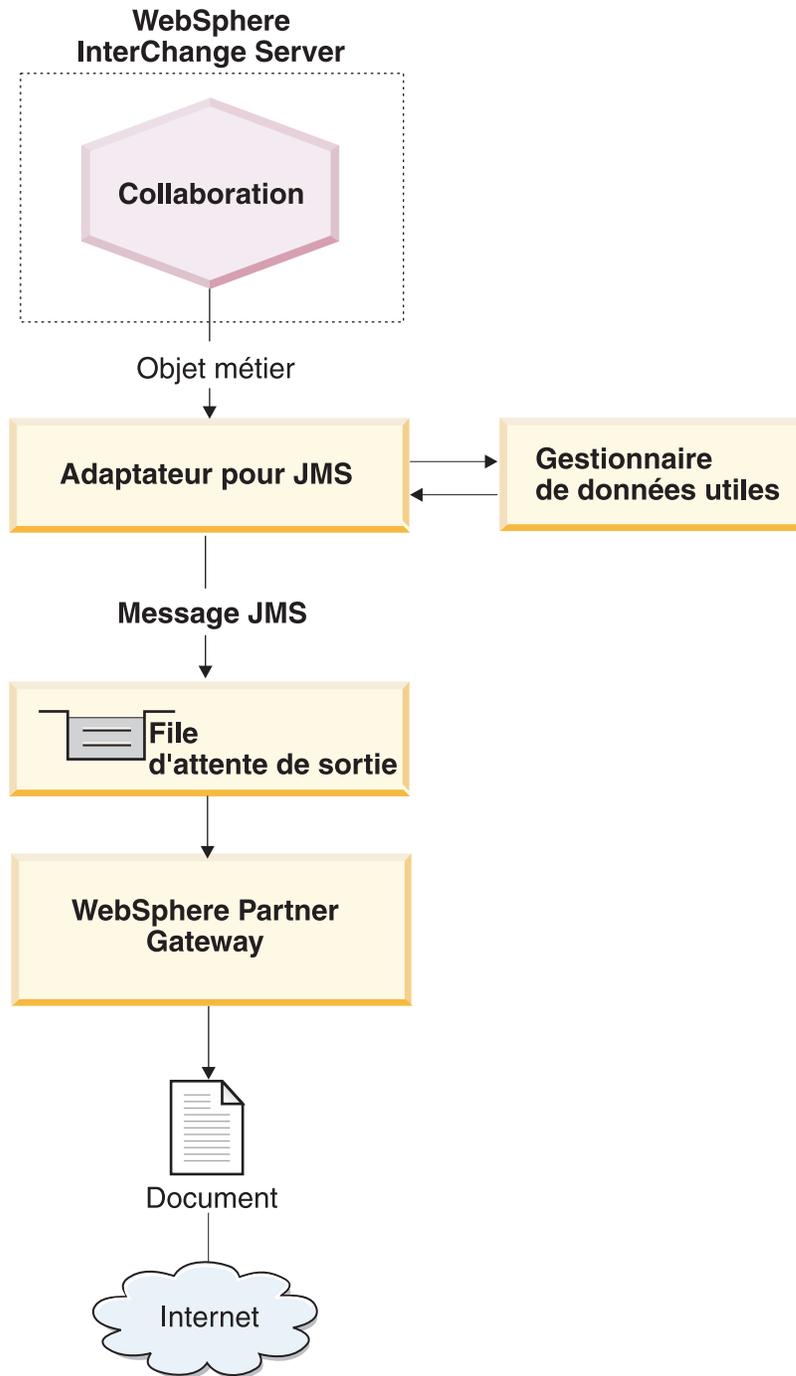


Figure 36. Flux de messages entre une collaboration et WebSphere Partner Gateway via JMS

Les étapes suivantes expliquent comment WebSphere Partner Gateway participe au traitement des requêtes en recevant un document depuis une collaboration dans InterChange Server via le protocole de transport JMS :

1. La collaboration dans InterChange Server appelle Adapter for JMS, en lui envoyant l'objet métier de type demande.

L'objet métier de type demande contient des informations propres à l'application qui désignent un méta-objet dynamique comprenant des informations d'en-tête de transport JMS attendues par WebSphere Partner Gateway.

2. Adapter for JMS utilise un gestionnaire de données pour convertir l'objet métier que la collaboration a envoyé dans un message JMS.  
L'adaptateur lit les propriétés `DataHandlerMimeType` et `DataHandlerConfigMO` afin de déterminer quel gestionnaire de données utiliser. Pour plus d'informations, voir «Spécification du gestionnaire de données utiles», à la page 176.

**Remarque :** Si vos documents contiennent des pièces jointes, installez le gestionnaire de données en pièce jointe, puis configurer Adapter for JMS en vue de convertir l'objet métier de type demande en document contenant des pièces jointes. Pour plus d'informations, voir «Gestion des documents contenant des pièces jointes», à la page 132.

3. Le gestionnaire de données convertit l'objet métier en chaîne et la renvoie à Adapter for JMS.
4. Adapter for JMS détermine, à partir de l'objet métier de type demande, le nom du méta-objet dynamique des propriétés personnalisées JMS.  
L'adaptateur recherche les informations propres à l'application relatives à l'objet métier de type demande correspondant à la balise `cw_mo_conn`, qui identifient l'attribut contenant le méta-objet dynamique. Si vous utilisez l'empaquetage Backend Integration pour votre document, vous pouvez indiquer les informations d'en-tête de transport dans ce méta-objet dynamique.
5. L'Adapter pour JMS recherche le méta-objet dynamique correspondant à l'attribut `JMSProperties`.

Si cet attribut a une valeur, l'adaptateur définit les zones d'en-tête de transport dans le document de requête. Dans l'attribut `JMSProperties`, vous pouvez également indiquer l'en-tête JMS standard du type de contenu. Pour plus d'informations, voir «Création des informations d'en-tête JMS», à la page 179.

6. Adapter for JMS crée un message JMS à l'aide de la chaîne renvoyée par le gestionnaire de données. Il définit également les propriétés personnalisées, telles qu'elles sont définies dans le méta-objet dynamique.
7. Adapter for JMS envoie le message de requête résultant à une file d'attente de sortie.  
La file d'attente peut être indiquée dans le méta-objet statique ou le méta-objet dynamique. Pour plus d'informations sur la spécification des files d'attente, voir «Identification des files d'attente JMS», à la page 177. WebSphere Partner Gateway écoute sur cette file d'attente JMS configurée comme file d'attente d'entrée dans la définition de son récepteur.
8. WebSphere Partner Gateway reçoit le message à partir de sa file d'attente d'entrée JMS, telle qu'elle est configurée dans son récepteur.

**Remarque :** WebSphere Partner Gateway prend uniquement en charge les interactions *asynchrones* avec InterChange Server via le transport JMS. Par conséquent, il se peut que vous ne souhaitiez pas attendre la réponse. La réponse du partenaire externe ou de WebSphere Partner Gateway peut arriver sur une autre file d'attente. Vous pouvez configurer Adapter for JMS pour interroger cette file d'attente. La réponse qui parvient sur la file d'attente peut être transmise à InterChange Server dans le cadre du processus de distribution des événements.

## Configuration de l'environnement requis pour le transport JMS

Etant donné que l'échange de documents sur InterChange Server nécessite l'utilisation d'Adapter for JMS et de files d'attente JMS, vous devez effectuer les tâches d'installation et de configuration décrites dans le tableau 64. Pour plus d'informations sur la manière de configurer WebSphere Partner Gateway pour une utilisation avec InterChange Server via JMS, voir «Configuration de WebSphere Partner Gateway pour InterChange Server», à la page 123.

Tableau 64. Configuration de l'environnement requis pour l'utilisation du protocole JMS

Etape de configuration	Pour plus d'informations
1. Configurez vos files d'attente JMS.	«Configuration des files d'attente JMS»
2. Configurez WebSphere Business Integration Adapter for JMS.	«Configuration d'Adapter for JMS», à la page 176

**Remarque :** Si vos documents contiennent des pièces jointes, vous devez également installer et configurer le gestionnaire de données de pièces jointes. Pour plus d'informations, voir «Gestion des documents contenant des pièces jointes», à la page 132.

## Configuration des files d'attente JMS

Pour utiliser le protocole JMS avec InterChange Server, vous devez configurer le système JMS fourni par WebSphere MQ. Les versions prises en charge d'InterChange Server utilisent comme fournisseur JMS la version 6.0 de WebSphere MQ avec dernier groupe de correctifs. Vous pouvez suivre les étapes du *Guide de configuration du concentrateur WebSphere Partner Gateway* pour configurer le mécanisme de protocole de transport JMS.

**Important :** Les opérations décrites dans le *Guide de configuration du concentrateur WebSphere Partner Gateway* doivent être effectuées sur l'ordinateur sur lequel WebSphere Partner Gateway est installé. Ce document présume que le mécanisme de transport JMS requis par Adapter for JMS et InterChange Server a déjà été configuré pendant l'installation d'InterChange Server.

Lors de la création de vos files d'attente JMS à utiliser entre WebSphere Partner Gateway et InterChange Server, prenez en considération les points suivants :

- Une partie du processus d'installation d'InterChange Server implique la création d'un gestionnaire de files d'attente WebSphere MQ. Vous pouvez utiliser ce gestionnaire de files d'attente avec WebSphere Partner Gateway.
- Lorsque vous créez vos alias de file d'attente JMS, vous pouvez les nommer de manière à indiquer la direction du flux entre WebSphere Partner Gateway et InterChange Server. Par exemple, si vous créez les files d'attente répertoriées dans la colonne Nom de file d'attente d'origine du tableau 65, vous pouvez renommer ces files d'attente pour qu'elles indiquent le flux directionnel d'InterChange Server, comme indiqué dans la colonne Nom de file d'attente directionnelle du tableau 65.

Tableau 65. Attribution de noms aux files d'attente JMS pour qu'InterChange Server donne des indications de direction

Nom de file d'attente d'origine	Nom de file d'attente directionnelle
inQ	ICS2HUB
outQ	HUB2ICS

## Configuration d'Adapter for JMS

Adapter for JMS permet à WebSphere Partner Gateway d'échanger des documents avec InterChange Server sous la forme de messages JMS. Il prend en charge les interactions suivantes avec InterChange Server :

- Pour le traitement des demandes, Adapter for JMS reçoit l'objet métier de type demande depuis InterChange Server, le convertit en message JMS (à l'aide d'un gestionnaire de données), puis le place dans une file d'attente JMS (voir la figure 36, à la page 173), d'où il peut être récupéré par WebSphere Partner Gateway.
- Pour la notification d'événements, il interroge une file d'attente JMS afin de récupérer les messages JMS à partir de WebSphere Partner Gateway. Lorsqu'il reçoit un message JMS, il le convertit en objet métier événement (à l'aide d'un gestionnaire de données) et l'envoie à InterChange Server.

**Important :** WebSphere Partner Gateway n'inclut *pas* WebSphere Business Integration Adapter for JMS. Vous devez vous procurer ce produit séparément et l'installer conformément aux instructions figurant dans le document *Guide de Adapter for JMS*. Il est important de lire attentivement les opérations décrites dans ce document afin d'installer et de configurer correctement Adapter for JMS.

Lorsque vous avez configuré Adapter for JMS pour communiquer avec InterChange Server, suivez les étapes de cette section afin de configurer cet adaptateur pour qu'il accepte les messages JMS envoyés par WebSphere Partner Gateway :

- «Spécification du gestionnaire de données utiles»
- «Identification des files d'attente JMS», à la page 177

### Spécification du gestionnaire de données utiles

Comme l'illustre la figure 36, à la page 173, Adapter for JMS utilise un gestionnaire de données pour convertir les objets métier reçus depuis InterChange Server en un message JMS.

**Remarque :** Le gestionnaire de données qu'Adapter for JMS appelle convertit les données utiles du document. Si votre document est inclus dans une enveloppe XML (car il comprend des pièces jointes ou l'indicateur d'enveloppe est sur Oui), configurer le gestionnaire de données en pièce jointe de sorte qu'il fonctionne comme un gestionnaire de données utiles. Pour plus d'informations, voir «Gestion des documents contenant des pièces jointes», à la page 132.

Pour indiquer quel gestionnaire de données utiliser pour convertir les données utiles, vous devez suivre les étapes spécifiées dans la section «Conversion d'objet métier», à la page 129. De plus, vous devez configurer Adapter for JMS pour qu'il utilise ce gestionnaire de données utiles. Dans le panneau de configuration du connecteur, suivre les étapes suivantes :

1. Définissez les propriétés de configuration du connecteur afin d'identifier le gestionnaire de données utiles à utiliser :
  - Attribuez aux propriétés `DataHandlerConfigMO` et `DataHandlerMimeType`, respectivement, le nom du méta-objet du gestionnaire de données de niveau supérieur et le type MIME pris en charge.
  - Affectez à la propriété `DataHandlerClassName` le nom de la classe du gestionnaire de données à instancier.

**Remarque :** Vous définissez *soit* les propriétés `DataHandlerConfigMO` et `DataHandlerMimeType` *soit* la propriété `DataHandlerClassName`.

2. Ajoutez le méta-objet du gestionnaire de données de niveau supérieur dans la liste des objets métier pris en charge.

Vous pouvez également indiquer quel gestionnaire de données utiliser dans le méta-objet dynamique ou statique. Les mêmes propriétés (`DataHandlerMimeType`, `DataHandlerConfigMO` et `DataHandlerClassName`) sont disponibles en tant qu'attributs dans ces méta-objets. Pour plus d'informations, voir le document *Adapter for JMS User Guide*.

## Identification des files d'attente JMS

Lorsque Adapter for JMS reçoit un document d'InterChange Server, il le place dans sa file d'attente de sortie, qui est ensuite interrogée par le récepteur de WebSphere Partner Gateway. De même, lorsque WebSphere Partner Gateway envoie un document à InterChange Server, il place le document dans sa file d'attente de sortie, qui est interrogée par Adapter for JMS.

Le tableau 66 résume comment configurer les files d'attente qu'Adapter for JMS utilise pour recevoir et envoyer des documents.

**Remarque :** Pour plus d'informations sur la configuration des files d'attente JMS, voir le document *Adapter for JMS User Guide*.

Tableau 66. Files d'attente JMS

File d'attente JMS	Jeu de configuration
File d'attente d'entrée	Définissez la propriété de configuration du connecteur <code>InputDestination</code> sur le nom de la file d'attente JMS qu'Adapter for JMS interrogera pour obtenir les messages entrants.  Vérifiez que le nom de cette file d'attente est identique à celui que WebSphere Partner Gateway utilise comme file d'attente JMS de sortie. Si cette file d'attente n'est pas spécifiée dans la propriété <code>InputDestination</code> , Adapter for JMS n'interrogera <i>pas</i> la file d'attente. <b>Remarque :</b> La propriété <code>InputDestination</code> contient une liste de files d'attente d'entrée séparée par des virgules. Si Adapter for JMS interroge plusieurs files d'attente, assurez-vous que cette liste contient le nom de la file d'attente JMS qu'utilise WebSphere Partner Gateway comme file d'attente JMS de sortie.
File d'attente de sortie	Pendant l'exécution, la collaboration peut affecter de manière dynamique à l'attribut <code>OutputQueue</code> dans le méta-objet dynamique le nom de la file d'attente JMS qu'Adapter for JMS utilise pour envoyer son message.

Vous devez vous assurer que les méta-objets statiques ou dynamiques sont configurés pour pouvoir écrire sur la file d'attente sur laquelle le récepteur de WebSphere Partner Gateway est en mode écoute.

---

## Création de définitions d'objet métier pour JMS

Adapter for JMS envoie votre document à InterChange Server et le reçoit sous la forme d'un objet métier données utiles. Adapter for JMS appelle le gestionnaire de données utiles pour qu'il gère cet objet métier lorsqu'il reçoit ou envoie un document de WebSphere Partner Gateway, comme suit :

- Pour le traitement des requêtes, le gestionnaire de données utiles convertit l'objet métier de type demande en message JMS.
- Pour la notification d'événements, le gestionnaire de données convertit le message JMS en objet métier événement.

Par conséquent, vous devez créer des définitions d'objet métier comme illustré dans le tableau 67 pour représenter la structure d'objet métier données utiles attendue par Adapter for JMS.

Tableau 67. Définitions d'objets métier pour Adapter for JMS

Condition	Définition d'objet métier	Pour plus d'informations
Si vous utilisez l'emballage None ou Backend Integration pour votre message <i>et</i> si vos documents ne possèdent <i>pas</i> de pièces jointes	Objet métier données utiles	«Création d'une structure objet métier données utiles pour JMS».
Si vous utilisez l'emballage Backend Integration pour votre document	Objets métier destinés à contenir les informations d'en-tête : <ul style="list-style-type: none"> <li>• Méta-objet dynamique</li> <li>• Objet métier Propriétés JMS</li> </ul>	«Création des informations d'en-tête JMS», à la page 179.
Si le document contient des pièces jointes	Vous devez également créer des objets métier supplémentaires pour représenter les pièces jointes.	«Création de définitions d'objets métier liées aux pièces jointes», à la page 144

## Création d'une structure objet métier données utiles pour JMS

La structure de l'objet métier données utiles pour le protocole JMS est fonction du type d'emballage sélectionné, comme suit :

- Si votre document n'utilise aucun emballage, il n'existe pas de conditions particulières pour créer un objet métier données utiles pour un document envoyé via le protocole JMS.

Comme l'explique la section «Objet métier défini pour un document», à la page 128, vous devez créer un attribut pour chaque donnée utile à transférer.

- Si votre document utilise l'emballage Backend Integration, vous devez procéder aux étapes suivantes :
  1. Ajoutez à la définition de l'objet métier données utiles un attribut spécifique permettant d'identifier le méta-objet dynamique. Cet attribut contient des informations de configuration pour les en-têtes de transport du message.
  2. Ajoutez la balise `cw_mo_conn` aux informations propres à l'application au niveau de l'objet métier afin d'identifier l'attribut qui contient le méta-objet dynamique.

Pour plus d'informations sur ces étapes, voir «Création des informations d'en-tête JMS», à la page 179.

**Remarque :** Pour le traitement des demandes, le protocole JMS peut *uniquement* prendre en charge les interactions asynchrones. Vous pouvez envoyer un objet métier de type demande, mais vous ne pouvez *pas* obtenir de réponse. Par conséquent, vous devez créer une définition d'objet métier de type demande mais pas une définition d'objet métier de type réponse.

## Création des informations d'en-tête JMS

Si vous échangez des documents avec l'empaquetage Backend Integration via le protocole de transport JMS, votre objet métier de type demande doit contenir des informations d'en-tête de niveau transport personnalisé. Adapter for JMS s'attend à ce que les informations d'en-tête personnalisé soient contenues dans son *méta-objet dynamique*.

La figure 37 illustre la structure de l'objet métier qu'Adapter for JMS utilise pour un objet métier de type demande qui représente un document WebSphere Partner Gateway utilisant l'empaquetage Backend Integration.

**Remarque :** Pour plus d'informations sur cette structure d'objet métier, voir le document *Adapter for JMS User Guide*. Consultez ce document lorsque vous définissez vos objets métier.

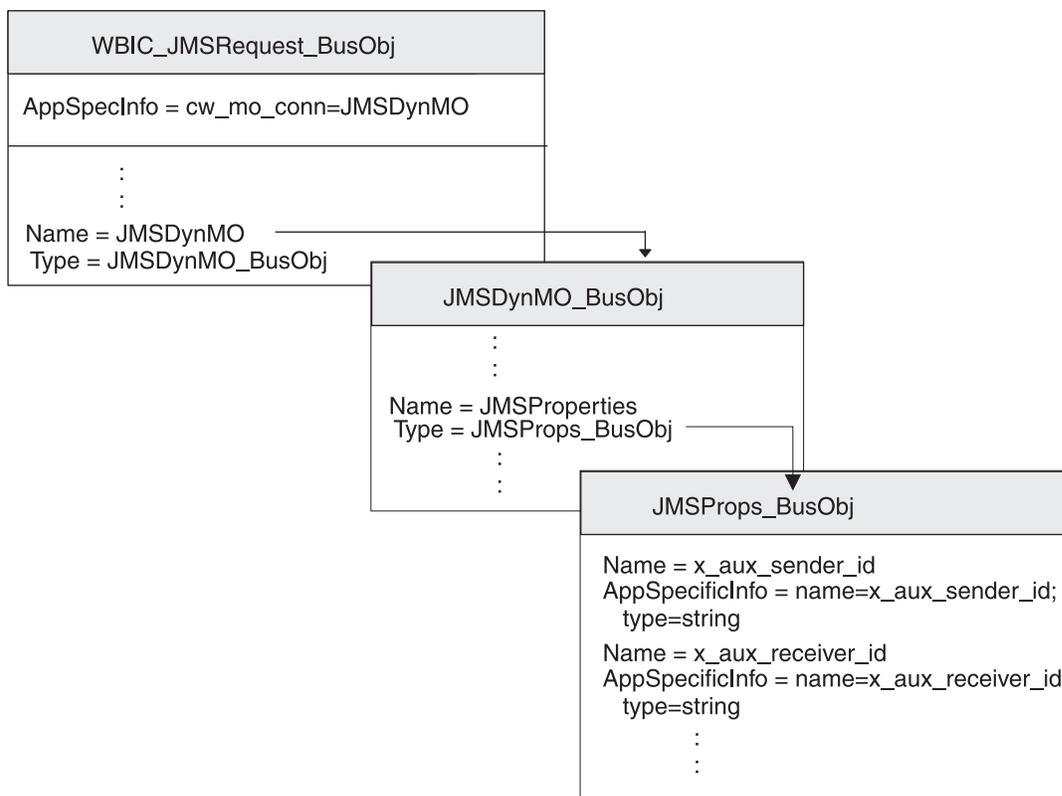


Figure 37. Relation entre l'objet métier de type demande et le méta-objet dynamique JMS

Assurez-vous que la structure de l'objet métier contient un méta-objet dynamique en procédant comme suit :

1. Créez une définition d'objet métier destinée à contenir les propriétés JMS requises par l'empaquetage Backend Integration.
2. Créez une définition d'objet métier pour l'objet métier dynamique.
3. Modifiez la définition de l'objet métier de type demande afin d'inclure un attribut pour le méta-objet dynamique.

Chaque étape est décrite dans les sections ci-dessous.

## Création de l'objet métier Propriétés JMS

Un *objet métier Propriétés JMS* contient les propriétés JMS nécessaires pour les en-têtes de transport requises par l'empaquetage Backend Integration. Il peut également contenir l'attribut du type de contenu, qui indique l'en-tête du type de contenu à définir dans le message de requête, et l'attribut de longueur du contenu, qui indique la longueur du message, en octets. Le tableau 8, à la page 27 décrit chaque zone valide de l'en-tête de transport.

Pour créer une définition d'objet métier Propriétés JMS, procédez comme suit :

1. Créez un attribut au sein de la définition de l'objet métier pour chaque zone d'en-tête de transport.

Tous les attributs doivent comporter un type d'attribut Chaîne. Pour les messages utilisant le protocole JMS, les noms des zones d'en-tête de transport utilisent des traits de soulignement au lieu des traits d'union, comme indiqué dans le tableau 68.

2. Pour chacun des attributs contenus dans la définition d'objet métier Propriétés JMS, ajoutez les informations propres à l'application qui permettent d'identifier l'objectif de l'attribut associé.

Ces informations propres à l'application au niveau de l'attribut doivent avoir le format suivant :

```
name=JMSproperty;type=string
```

où *JMSproperty* est l'une des valeurs contenues dans la colonne Nom de propriété JMS du tableau 68.

3. Pour tous les attributs contenus dans la définition d'objet métier Propriétés JMS, vous pouvez ajouter une valeur par défaut afin d'indiquer la valeur commune (ou uniquement valide) pour la zone d'en-tête de transport.

Tableau 68. Attributs de la définition de l'objet métier Propriétés JMS

Zone d'en-tête de transport	Nom de propriété JMS
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
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

**Remarque :** Le tableau 68 ne fournit *pas* de liste exhaustive des en-têtes requis pour l'intégration dorsale. Pour obtenir la liste complète et la description des en-têtes, voir «Contenu de l'en-tête de niveau transport», à la page 27. Veillez à remplacer les traits d'union par les caractères de soulignement dans les noms de zone d'en-tête de transport.

Dans la figure 37, à la page 179, la définition de l'objet métier JMSProps\_BusObj contient des attributs pour les différentes zones d'en-tête de transport. Ces attributs possèdent tous des informations propres à l'application au niveau des attributs afin de spécifier le nom de l'en-tête du protocole associé. Par exemple, les informations propres à l'application de l'attribut `x_aux_sender_id` sont définies comme suit :

```
name=x_aux_sender_id;type=string
```

## Création du méta-objet dynamique JMS

Ce méta-objet dynamique contient un objet métier enfant associé à des informations de configuration destinées à Adapter for JMS. Pour créer une définition d'objet métier pour un méta-objet dynamique, procédez comme suit :

1. Créez un attribut nommé `JMSProperties` dont le type correspond à la définition de l'objet métier propriétés JMS (voir «Création de l'objet métier Propriétés JMS», à la page 180).
2. Ajoutez d'autres propriétés de configuration, le cas échéant. Pour obtenir la liste des attributs valides contenus dans le méta-objet dynamique, voir le document *Adapter for JMS User Guide*. Consultez ce document pour plus d'informations sur la création d'attributs afin de configurer le méta-objet dynamique.

Pour qu'Adapter for JMS utilise WebSphere Partner Gateway, la définition de l'objet métier pour le méta-objet dynamique *doit* inclure l'attribut nommé `JMSProperties`, dont le type correspond à la définition de l'objet métier propriétés JMS (voir «Création de l'objet métier Propriétés JMS», à la page 180). Par exemple, dans la figure 37, à la page 179, la définition de l'objet métier `JMSDynMO_BusObj` contient les attributs de plusieurs propriétés de configuration (non présentées) et inclut l'attribut `JMSProperties`.

## Modification de la définition de l'objet métier de type demande

Pour intégrer la structure de l'objet métier à votre objet métier de type demande, vous devez effectuer les modifications suivantes dans la définition de l'objet métier de type demande :

1. Ajoutez un attribut à votre définition d'objet métier de type demande afin qu'il contienne le méta-objet enfant dynamique.

Le type de cet attribut est la définition d'objet métier du méta-objet dynamique (voir «Création du méta-objet dynamique JMS»).

2. Ajoutez la balise `cw_mo_conn` aux informations propres à l'application au niveau de l'objet métier de votre définition d'objet métier afin d'identifier l'attribut qui contient le méta-objet dynamique.

La balise `cw_mo_conn` a le format suivant :

```
cw_mo_conn=attrMétaObjDynamique
```

où `attrMétaObjDynamique` est le nom de l'attribut dans l'objet métier de type demande qui contient le méta-objet dynamique.

Par exemple, dans la figure 37, à la page 179, un attribut nommé JMSDynMO a été ajouté à la définition de l'objet métier de type demande, HUB\_JMSRequest\_BusObj. Cet attribut contient le méta-objet dynamique, qui est un objet métier enfant de type JMSDynMO\_BusObj. Par ailleurs, les informations propres à l'application de l'objet métier de type demande ont été modifiées pour inclure la balise cw\_mo\_conn suivante afin d'identifier ce méta-objet dynamique :

```
cw_mo_conn=JMSDynMO
```

## Création d'artefacts ICS pour JMS

Pour configurer InterChange Server pour qu'il communique avec WebSphere Partner Gateway via le protocole JMS, vous devez créer les artefacts InterChange Server indiqués dans le tableau 69.

Tableau 69. Artefacts ICS pour l'établissement de communication sur protocole JMS

Artefact ICS	Objet	Pour plus d'informations
Définitions d'objet métier	Représente le document	«Création de définitions d'objet métier pour JMS», à la page 177
Objet connecteur	Représente Adapter for JMS au moment de l'exécution	«Création de l'objet connecteur JMS»
Modèle et objet collaboration	Représente le processus métier qu'InterChange Server utilise pour traiter le document	«Mise en place de liens de collaboration en vue d'une communication avec Adapter for JMS», à la page 183

## Création de l'objet connecteur JMS

Pour obtenir une instance d'Adapter for JMS au moment de l'exécution, procéder selon les étapes suivantes dans System Manager :

1. Créez les objets de connecteur :
  - Créez un objet connecteur afin de représenter une instance d'Adapter for JMS.

**Remarque :** Dans l'onglet des objets métier pris en charge de Connector Configurator, vérifiez que vous avez indiqué toutes les définitions d'objet métier que vous avez créées en vue de les utiliser avec Adapter for JMS. Pour obtenir la description de ces définitions d'objet métier, voir «Création de définitions d'objet métier pour JMS», à la page 177.

- Si cela est requis par votre collaboration, créez un objet connecteur pour le connecteur de port.
2. Configurez les objets de connecteur.

Pour plus d'informations sur la manière de configurer Adapter for JMS afin de l'utiliser avec WebSphere Partner Gateway, voir «Configuration d'Adapter for JMS», à la page 176.

## Mise en place de liens de collaboration en vue d'une communication avec Adapter for JMS

Comme décrit dans «Création des collaborations», à la page 131, un objet collaboration doit exister au moment de l'exécution pour qu'InterChange Server puisse savoir où recevoir et envoyer des objets métier. Lorsque vous créez l'objet collaboration pour la collaboration qui utilise Adapter for JMS pour échanger des informations avec WebSphere Partner Gateway, vous devez lier les ports de collaboration, comme suit :

- Pour le traitement des requêtes, affectez au port "cible", qui envoie des requêtes à WebSphere Partner Gateway, l'objet connecteur que vous avez créé pour Adapter for JMS, à savoir, l'adaptateur de *destination*.
- Pour la notification d'événements, affectez au port "source", qui reçoit les événements à partir de WebSphere Partner Gateway, l'objet connecteur que vous avez créé pour Adapter for JMS, à savoir, l'adaptateur *source*.



---

## **Partie 4. Intégration à d'autres systèmes dorsaux**

Les chapitres suivants décrivent la procédure permettant d'intégrer WebSphere Partner Gateway à un système dorsal tel que WebSphere Message Broker ou WebSphere Data Interchange.



---

## Chapitre 11. Intégration à WebSphere Message Broker

Le présent chapitre décrit le processus d'intégration de WebSphere Partner Gateway à WebSphere Message Broker.

### Remarques :

1. Pour obtenir une description du processus général utilisé pour intégrer WebSphere Partner Gateway à un système dorsal, voir Chapitre 2, «Planification de l'intégration dorsale», à la page 9.
2. Ce chapitre suppose que vous connaissiez WebSphere Message Broker et les composants qui lui sont associés, tels que les projets et les flux de messages.

En général, l'intégration de WebSphere Partner Gateway à un système dorsal est effectuée par deux personnes ou rôles distincts. Chaque rôle configure un composant particulier pour lequel ce rôle est compétent. Par conséquent, le présent chapitre répartit l'intégration à WebSphere Message Broker en deux tâches : configuration de WebSphere Partner Gateway et configuration de Message Broker. Le tableau 70 répertorie ces rôles de configuration et indique les sections de ce chapitre qui contiennent les informations de configuration associées.

Tableau 70. Rôles dans le processus d'intégration à WebSphere Message Broker

Rôle de configuration	Pour plus d'informations
Configuration de WebSphere Partner Gateway	<ol style="list-style-type: none"><li>1. «Planification de l'intégration à WebSphere Message Broker», à la page 188</li><li>2. «Configuration de WebSphere Partner Gateway pour WebSphere Message Broker», à la page 189</li></ol>
Configuration de WebSphere Message Broker	<ol style="list-style-type: none"><li>1. «Planification de l'intégration à WebSphere Message Broker», à la page 188</li><li>2. «Configuration de WebSphere Message Broker», à la page 192</li></ol>

**Remarque :** Chacun de ces rôles de configuration peut être effectué séparément ; cependant des informations communes sont requises pour chaque rôle de sorte que les deux composants puissent communiquer.

Ce chapitre contient les informations suivantes :

- «Planification de l'intégration à WebSphere Message Broker», à la page 188
- «Configuration de WebSphere Partner Gateway pour WebSphere Message Broker», à la page 189
- «Configuration de WebSphere Message Broker», à la page 192
- «Utilisation du protocole de transport HTTP avec WebSphere Message Broker», à la page 193
- «Envoi de documents SOAP», à la page 197
- «Utilisation du protocole de transport JMS avec WebSphere Message Broker», à la page 198

## Planification de l'intégration à WebSphere Message Broker

Pour planifier votre intégration à WebSphere Message Broker, procédez aux étapes indiquées à la section Chapitre 2, «Planification de l'intégration dorsale», à la page 9. Le tableau 71 récapitule les étapes d'intégration de WebSphere Partner Gateway à WebSphere Message Broker.

Tableau 71. Planification de l'intégration à WebSphere Message Broker

Etape d'intégration	Pour plus d'informations
1. Confirmez qu'une version prise en charge de WebSphere Message Broker est installée et accessible à WebSphere Partner Gateway.	Chapitre 11: «Versions de WebSphere Message Broker prises en charge par WebSphere Partner Gateway»
2. Déterminez le protocole de gestion du document de WebSphere Message Broker.	Chapitre 2: «Quel est le protocole de gestion utilisé ?», à la page 9
3. Déterminez le type d'emballage pour le document : None ou Backend Integration.	Chapitre 2: «Quel emballage allez-vous utiliser ?», à la page 26
4. Déterminez le mode de transport de message à utiliser entre WebSphere Partner Gateway et WebSphere Message Broker.	Chapitre 11: «Transports de message pris en charge par WebSphere Message Broker»
5. Configurez WebSphere Partner Gateway.	Chapitre 11: «Configuration de WebSphere Partner Gateway pour WebSphere Message Broker», à la page 189

## Versions de WebSphere Message Broker prises en charge par WebSphere Partner Gateway

WebSphere Partner Gateway prend en charge l'intégration à la version 6.0 de Message Broker incluant le dernier groupe de correctifs. Message Broker est disponible sur de nombreuses plateformes, y compris Windows 2000 et plusieurs plateformes UNIX. Pour plus d'informations, consultez le guide d'installation de Message Broker dans la documentation WebSphere Message Broker.

## Transports de message pris en charge par WebSphere Message Broker

Les deux protocoles de transport de message suivants sont pris en charge par WebSphere Message Broker :

- Protocole de transport HTTP (y compris services Web)
- Protocole de transport JMS

La prise en charge de ces protocoles de transport de message requiert l'installation et la configuration d'IBM WebSphere MQ.

### HTTP

Message Broker utilise le protocole de transport HTTP pour ses transactions de services Web. Vous n'avez besoin d'aucun logiciel supplémentaire pour échanger des documents entre WebSphere Partner Gateway et Message Broker via le protocole HTTP. Cependant, l'envoi de documents hors de Message Broker à une autre destination requiert WebSphere MQ.

**Remarque :** Avec Message Broker via HTTP, WebSphere Partner Gateway prend en charge tant les interactions synchrones que les interactions asynchrones.

## JMS

Message Broker utilise le protocole de transport JMS pour la plupart de ses transactions. Pour échanger des documents entre WebSphere Partner Gateway et Message Broker via le protocole de transport JMS, vous devez utiliser des files d'attente JMS. Si ces deux composants résident sur des ordinateurs différents, créez les files d'attente JMS sur chaque ordinateur. En règle générale, la prise en charge du protocole JMS nécessite l'utilisation d'un flux de messages dans Message Broker et les files d'attente JMS sous-jacentes. Pour plus d'informations sur la configuration du protocole JMS, voir «Utilisation du protocole de transport JMS avec WebSphere Message Broker», à la page 198.

**Remarque :** Avec Message Broker via JMS, WebSphere Partner Gateway ne prend en charge que les interactions asynchrones.

## Prise en charge de l'intégration à WebSphere Message Broker

WebSphere Partner Gateway propose des exemples pour faciliter le processus d'intégration à Message Broker. Ces exemples figurent dans le sous-répertoire suivant du répertoire de produit de WebSphere Partner Gateway :

Integration/WBI/WBIMB/samples

---

## Configuration de WebSphere Partner Gateway pour WebSphere Message Broker

La section «Configuration de WebSphere Partner Gateway», à la page 47 donne un aperçu général sur la manière dont WebSphere Partner Gateway est configuré pour communiquer avec un système dorsal. Cette section résume les étapes nécessaires pour configurer WebSphere Partner Gateway afin de communiquer avec Message Broker.

La configuration de WebSphere Partner Gateway implique les étapes suivantes :

- Configuration de la prise en charge des documents sortants

Pour plus d'informations sur l'envoi de documents à partir de WebSphere Partner Gateway à Message Broker, voir «Prise en charge des documents sortants».

- Configuration des documents entrants

Pour plus d'informations sur l'envoi de documents à partir de Message Broker à WebSphere Partner Gateway, voir «Prise en charge des documents entrants», à la page 191.

## Prise en charge des documents sortants

Pour que WebSphere Partner Gateway puisse envoyer des documents à un système dorsal, vous devez effectuer les opérations expliquées à la section «Définition de la cible d'envoi du document du partenaire», à la page 49. Lorsque votre système dorsal est Message Broker, vous devez créer une destination dont le type de transport correspond au protocole de transport utilisé pour l'échange de messages entre WebSphere Partner Gateway et Message Broker. Lorsque le concentrateur envoie un document à Message Broker, il doit savoir vers où router le document. Cet emplacement doit être conforme au protocole de transport utilisé. Le protocole de transport doit être pris en charge par Message Broker (voir «Transports de message pris en charge par WebSphere Message Broker», à la page 188).

Les sections ci-dessous résument comment créer des destinations pour les protocoles de transport suivants pris en charge par Message Broker :

- «Configuration pour l'envoi de documents via le protocole de transport HTTP»
- «Configuration pour la réception de documents via le protocole de transport JMS»

## Configuration pour l'envoi de documents via le protocole de transport HTTP

Lorsque le concentrateur envoie un document à Message Broker via le protocole HTTP, il transmet le message par le biais de la destination définie. Cette destination identifie l'adresse URL à laquelle Message Broker peut recevoir le document. Lorsque Message Broker utilise le protocole HTTP, il route le document vers le noeud HTTPInput du flux de messages associé à l'adresse URL indiquée.

Pour que le concentrateur puisse envoyer des documents via une destination via le protocole de transport HTTP, créez une destination sur la page Caractéristiques de la destination de la console de communauté. Cette destination doit être configurée pour utiliser le protocole de transport HTTP 1.1 et écrire à l'adresse URL du noeud HTTPInput situé à l'extrémité. Comme le montre le tableau 72, cette adresse URL est indiquée dans la zone URI du récepteur de la définition de destination.

**Remarque :** La section «Définition de la cible d'envoi du document du partenaire», à la page 49 décrit comment créer une destination.

Tableau 72. Valeurs HTTP de la page Caractéristiques de la destination pour la communication avec WebSphere Message Broker

Zone Caractéristiques du récepteur	Valeur	Remarques et limitations
URI du récepteur	L'adresse URL doit être identique à celle configurée pour le noeud HTTPInput dans le flux de messages Message Broker.	Extrayez cette adresse URL de la configuration du flux de messages dans l'intégration de WebSphere Message Broker.

## Configuration pour la réception de documents via le protocole de transport JMS

Lorsque le concentrateur envoie un document à Message Broker via le protocole JMS, il transmet le document à la file d'attente JMS correspondante, d'où il peut être ensuite transféré à la file d'attente JMS à partir de laquelle il peut être récupéré par Message Broker. Pour que le concentrateur puisse obtenir cet emplacement JMS, vous devez créer dans WebSphere Partner Gateway une destination qui utilise le protocole de transport JMS. Cette destination doit être configurée pour écrire dans la file d'attente dont le contenu est transféré à la file d'attente sur laquelle Message Broker reçoit des messages.

**Remarque :** Pour plus d'informations sur la création d'une destination, voir «Définition de la cible d'envoi du document du partenaire», à la page 49.

Pour que le concentrateur puisse envoyer des documents via une destination via le protocole de transport JMS, créez une destination sur la page Caractéristiques de la destination de la console de communauté. Si vous utilisez comme fournisseur JMS la version 6.0 de WebSphere MQ incluant le dernier groupe de correctifs, reportez-vous au *Guide de configuration du concentrateur WebSphere Partner Gateway* pour obtenir les étapes détaillées. Par ailleurs, utilisez les informations contenues dans le tableau 73 concernant le protocole JMS sur la page Caractéristiques de la destination.

Tableau 73. Valeurs JMS de la page Caractéristiques de la destination pour la communication avec WebSphere Message Broker

Zone Caractéristiques de la destination	Valeur	Remarques et limitations
Nom de file d'attente JMS	Nom de la file d'attente JMS sur l'ordinateur sur lequel réside WebSphere Partner Gateway	Les documents reçus sur cette file d'attente sont transférés vers la file d'attente JMS de l'ordinateur sur lequel réside Message Broker.

## Prise en charge des documents entrants

Pour que WebSphere Partner Gateway puisse recevoir des messages à partir d'un système dorsal, vous devez effectuer les opérations indiquées à la section «Définition de l'emplacement à partir duquel le document dorsal sera récupéré», à la page 53. Si Message Broker est votre système dorsal, vous devez effectuer les opérations suivantes :

1. Dans le cadre de la définition du profil de votre partenaire interne, spécifiez le type de destination et indiquez l'adresse IP associée à laquelle le récepteur écoute.
2. Créez un récepteur dont le type de transport correspond au protocole de transport utilisé pour l'échange de documents entre WebSphere Partner Gateway et Message Broker.

Pour que le concentrateur reçoive un document depuis Message Broker, il doit connaître l'emplacement où récupérer les messages. Cet emplacement doit être compatible avec le protocole de transport utilisé.

Les sections suivantes résument comment créer des récepteurs pour les protocoles de transport pris en charge par Message Broker.

### Configuration de documents entrants via le protocole de transport HTTP

Lorsque le concentrateur reçoit un document via le protocole de transport HTTP, son récepteur extrait le document du récepteur défini. Ce récepteur défini identifie l'adresse URL sur laquelle le récepteur du concentrateur récupère les documents depuis Message Broker. Lorsque Message Broker utilise le protocole HTTP, le noeud HTTPRequest envoie le document à une adresse URL, où il est récupéré par le concentrateur.

Pour que le concentrateur puisse recevoir les documents via un récepteur via le protocole de transport HTTP, vous devez créer un récepteur sur la page Liste des récepteurs de la console de communauté. Ce récepteur doit utiliser le protocole de transport HTTP 1.1. Le concentrateur détermine cette URL sous la forme d'une combinaison des informations suivantes :

- l'adresse IP de l'ordinateur hôte, obtenue auprès du profil du partenaire interne ;
- l'adresse URL du récepteur, obtenue dans la zone URL de la définition du récepteur.

**Remarque :** La section «Définition de l'emplacement à partir duquel le document dorsal sera récupéré», à la page 53 explique comment créer un récepteur.

Pour que Message Broker puisse transmettre des documents à ce récepteur, le noeud HTTPRequest du flux de messages doit être configuré pour transmettre des documents à cette URL. Par conséquent, vous devez vous assurer que l'URL de ce récepteur est disponible pour la configuration de Message Broker.

## Configuration de documents entrants via le protocole de transport JMS

Lorsque le concentrateur reçoit un document à partir de Message Broker via le protocole JMS, il récupère le document dans la file d'attente d'entrée JMS appropriée, où il a été transféré à partir de la file d'attente de sortie JMS dans laquelle Message Broker l'a envoyé. Pour que le concentrateur puisse obtenir cet emplacement JMS, vous devez créer dans WebSphere Partner Gateway un récepteur qui utilise le protocole de transport JMS. Par le biais de ce récepteur, le concentrateur interroge la file d'attente d'entrée pour savoir si des documents ont été transmis, puis les récupère.

**Remarque :** Pour plus d'informations sur la création d'un récepteur, voir «Définition de l'emplacement à partir duquel le document dorsal sera récupéré», à la page 53.

Pour que le concentrateur puisse recevoir les documents via un récepteur via le protocole de transport JMS, vous devez créer un récepteur sur la page Liste des récepteurs de la console de communauté. Si vous utilisez comme fournisseur JMS la version 6.0 de WebSphere MQ incluant le dernier groupe de correctifs, reportez-vous au *Guide de configuration du concentrateur WebSphere Partner Gateway* pour obtenir les étapes détaillées. Par ailleurs, utilisez les informations contenues dans le tableau 74 concernant le protocole JMS sur la page Caractéristiques du récepteur.

Tableau 74. Valeurs JMS de la page Caractéristiques du récepteur pour la communication avec WebSphere Message Broker

Zone Caractéristiques du récepteur	Valeur	Remarques et limitations
Nom de file d'attente JMS	Nom de la file d'attente d'entrée JMS qui reçoit des documents de la file d'attente de sortie de Message Broker	Les documents de cette file d'attente d'entrée sont transférés à partir de la file d'attente de sortie JMS sur l'ordinateur sur lequel réside Message Broker.

---

## Configuration de WebSphere Message Broker

Pour vos interactions entre WebSphere Partner Gateway et Message Broker, vous devez créer un projet de flux de messages dans Broker Application Development Perspective du Toolkit de Message Broker. Ce projet contiendra les artefacts suivants :

- Flux de messages
- Fichiers PIP (RosettaNet uniquement) ou fichiers de définition de message

**Remarque :** Pour plus d'informations sur la création de projets de flux de messages, consultez la documentation WebSphere Message Broker.

## Création du flux de messages

C'est le *flux de messages*, dans Message Broker, qui exécute la logique applicative réelle dont vous avez besoin pour traiter les informations. Par conséquent, vous devez créer les flux de messages appropriés pour Message Broker afin de traiter correctement les documents de WebSphere Partner Gateway. Assurez-vous qu'un flux de messages fournissant la logique applicative dont vous avez besoin est disponible :

- Si un flux de messages de ce type n'existe *PAS*, vous devez le créer et l'importer.
- Si un flux de messages existe, vous devez comprendre comment l'utiliser.

Pour que Message Broker traite les documents entrants et sortants, son flux de messages utilise des noeuds de transport spéciaux. Le type de noeud de transport à utiliser est fonction du protocole de transport particulier, comme indiqué.

Tableau 75. Création de flux de messages pour des protocoles de transport différents

Protocole de transport	Pour plus d'informations
HTTP HTTP (documents SOAP)	«Création du flux de messages pour le transport HTTP», à la page 195
JMS	«Création du flux de messages pour le transport JMS», à la page 203

## Déploiement du projet

Une fois que le projet de flux de messages contient les artefacts corrects, il doit être déployé sur Message Broker. Pour déployer un projet de flux de messages, utilisez Broker Administrator Perspective du Toolkit de Message Broker.

---

## Utilisation du protocole de transport HTTP avec WebSphere Message Broker

Cette section explique comment échanger des documents entre WebSphere Partner Gateway et WebSphere Message Broker via le protocole de transport HTTP.

**Remarque :** Toutes les informations relatives au protocole de transport HTTP concernent également le protocole HTTPS.

## Composants requis pour les documents envoyés via HTTP

Vous n'avez besoin d'aucun logiciel supplémentaire pour échanger un document entre WebSphere Partner Gateway et Message Broker via le protocole de transport HTTP. Seuls WebSphere Partner Gateway et Message Broker sont requis. Pour que WebSphere Partner Gateway communique avec la version 5.0 de Message Broker via le protocole de transport HTTP, ces deux composants doivent être configurés. Le tableau 76 récapitule ces étapes de configuration.

Tableau 76. Configuration de WebSphere Partner Gateway et de WebSphere Message Broker

Composant	Version	Pour plus d'informations
WebSphere Partner Gateway	6.1	«Configuration pour l'envoi de documents via le protocole de transport HTTP», à la page 190  «Configuration de documents entrants via le protocole de transport HTTP», à la page 191
WebSphere Message Broker	6.0 avec dernier groupe de correctifs	«Configuration de WebSphere Message Broker», à la page 192

En outre, pour échanger un document entre WebSphere Partner Gateway et Message Broker via le protocole de transport HTTP, vous devez utiliser comme fournisseur JMS la version 6.0 d'IBM WebSphere MQ incluant le dernier groupe de correctifs.

### Envoi de documents via HTTP

Pour que WebSphere Partner Gateway envoie un document à Message Broker via le protocole de transport HTTP, utilisez des noeuds de transport HTTP spéciaux dans le flux de messages Message Broker afin de récupérer le document qui a été envoyé par WebSphere Partner Gateway sous la forme d'un flux HTTP. Les noeuds du flux de messages effectuent les calculs requis et routent ensuite le document vers une certaine destination (file d'attente de sortie JMS).

Les étapes suivantes expliquent comment WebSphere Partner Gateway envoie un document à un flux de messages dans Message Broker via le protocole de transport HTTP :

1. WebSphere Partner Gateway envoie un message HTTP à Message Broker.

Si le type d'emballage du document était Backend Integration, WebSphere Partner Gateway a intégré des propriétés personnalisées dans ce message.

**Remarque :** Dans WebSphere Partner Gateway, vous devez configurer une destination qui identifie l'adresse URL à laquelle WebSphere Partner Gateway envoie le message et dans laquelle Message Broker effectue des interrogations. Pour plus d'informations, voir «Configuration pour l'envoi de documents via le protocole de transport HTTP», à la page 190.

2. Le noeud HTTPInput du flux de messages récupère le document et l'envoie au noeud suivant du flux de messages. Ce noeud est généralement un noeud de traitement.
3. Les noeuds du flux de messages exécutent la logique applicative.  
Une fois la logique applicative terminée, le flux de messages envoie le document résultant à son noeud HTTPReply.
4. Par défaut, le noeud HTTPReply renvoie le message de sortie au client (WebSphere Partner Gateway).

Il se peut également que le flux de messages place le message dans un noeud MQOutput. Le noeud MQOutput reçoit le document et l'envoie à la file d'attente JMS appropriée ou à une autre application.

## Réception de documents via HTTP

Pour que WebSphere Partner Gateway reçoive un document de Message Broker via le protocole de transport HTTP, utilisez des noeuds de transport HTTP spéciaux dans le flux de messages Message Broker afin d'envoyer le document que WebSphere Partner Gateway doit recevoir sous la forme d'un flux HTTP. Les noeuds du flux de messages effectuent les calculs requis et traitent les interactions demande et réponse (s'il s'agit d'interactions synchrones) avec WebSphere Partner Gateway.

Les étapes suivantes expliquent comment WebSphere Partner Gateway reçoit un document à partir d'un flux de messages dans Message Broker via le protocole de transport HTTP :

1. Le flux de messages dans Message Broker reçoit un document dans son noeud MQInput (file d'attente d'entrée JMS).
2. Le noeud MQInput du flux de messages reçoit le document et l'envoie au noeud HTTPRequest.
3. Le noeud HTTPRequest traite les interactions demande et réponse avec le client (WebSphere Partner Gateway), à l'aide d'une adresse URL spécifiée.
4. WebSphere Partner Gateway reçoit le message à partir de son adresse URL, telle qu'elle est configurée dans son récepteur.

Pour plus d'informations sur le récepteur, voir «Configuration de documents entrants via le protocole de transport HTTP», à la page 191.

## Création du flux de messages pour le transport HTTP

Pour qu'un flux de messages Message Broker traite les documents via le protocole de transport HTTP, il utilise les noeuds de transport suivants :

- HTTPInput
- HTTPReply
- HTTPRequest

L'ordre dans lequel ces noeuds de transport sont utilisés est fonction du sens de la communication :

- Lorsque WebSphere Partner Gateway *envoie* un document à Message Broker, le flux de messages inclut les types de noeuds du tableau 77 (dans l'ordre indiqué) pour décrire la logique applicative.
- Lorsque WebSphere Partner Gateway *reçoit* un document de Message Broker, le flux de messages inclut les types de noeuds du tableau 78 (dans l'ordre indiqué) pour décrire la logique applicative.

Tableau 77. Noeuds permettant l'envoi de documents à WebSphere Message Broker via HTTP

Type de noeud	Objet	Remarques
HTTPInput	Reçoit le document WebSphere Partner Gateway demandé dans le flux de messages	<p>Paramétrez la zone de sélection d'URL de ce noeud de transport (dans les propriétés de base) sur l'adresse URL à laquelle WebSphere Partner Gateway envoie ses documents (adresse URL configurée dans le récepteur de WebSphere Partner Gateway). L'adresse URL doit être au format suivant :</p> <p><code>http://nomHôte:port/chemin</code></p> <p>où <i>nomHôte</i> représente le nom de l'ordinateur sur lequel réside Message Broker, <i>port</i> le numéro du port HTTP sur lequel Message Broker est en mode écoute et <i>chemin</i> l'emplacement de cet ordinateur.</p> <p>Pour plus d'informations, voir «Configuration pour l'envoi de documents via le protocole de transport HTTP», à la page 190.</p>
Calcul	Effectue des tâches de logique applicative, telles que la mise à jour des informations d'en-tête	Utilisez ESQL pour effectuer la logique applicative. Le noeud de traitement envoie le message résultant au noeud HTTPReply.
HTTPReply	Renvoie une réponse à WebSphere Partner Gateway	Par défaut, ce noeud envoie le message de sortie au client. Cependant, vous pouvez le configurer pour l'envoyer à un noeud MQOutput.
MQOutput	Reçoit le document du noeud HTTPReply et l'envoie à WebSphere Partner Gateway	Ce noeud de transport envoie le document résultant à une file d'attente de sortie JMS qui le route vers sa prochaine destination.

Tableau 78. Noeuds permettant la réception de documents de WebSphere Message Broker via HTTP

Type de noeud	Objet	Remarques
MQInput	Reçoit le document de WebSphere Partner Gateway	Ce noeud de transport reçoit le document entrant d'une file d'attente d'entrée JMS.
HTTPRequest	Traite les interactions demande/réponse avec WebSphere Partner Gateway	<p>Ce noeud de transport doit paramétrer sa zone d'URL de services Web (dans les propriétés de base) sur l'adresse URL que WebSphere Partner Gateway interroge pour extraire les documents (adresse URL configurée dans le récepteur de WebSphere Partner Gateway). L'adresse URL doit être au format suivant :</p> <p><code>http://nomHôte:port/bcgreceiver/chemin</code></p> <p>où <i>nomHôte</i> représente le nom de l'ordinateur sur lequel réside WebSphere Partner Gateway, <i>port</i> le numéro du port HTTP sur lequel le récepteur de WebSphere Partner Gateway est en mode écoute et <i>chemin</i> l'emplacement de cet ordinateur.</p> <p>Pour plus d'informations, voir «Configuration de documents entrants via le protocole de transport HTTP», à la page 191.</p>

Pour plus de détails sur la création et la configuration des noeuds de flux de messages, consultez la documentation WebSphere Message Broker.

## Envoi de documents SOAP

Les documents SOAP diffèrent des autres types de documents échangés via HTTP/S. Cette section explique comment échanger des documents SOAP entre WebSphere Partner Gateway et WebSphere Message Broker via le protocole de transport HTTP.

La procédure de configuration de WebSphere Partner Gateway et Message Broker pour le transport de documents SOAP est très similaire à celle du transport de documents non-SOAP via le protocole HTTP. Le tableau 79 indique où vous pouvez trouver des informations sur la configuration de ces deux composants d'intégration.

Tableau 79. Configuration de WebSphere Partner Gateway et de WebSphere Message Broker pour le transfert de documents SOAP

Composant d'intégration	Etape de configuration	Pour plus d'informations
WebSphere Partner Gateway	Le récepteur et la destination sont configurés de la même façon pour les documents SOAP que pour les documents non-SOAP via HTTP.	«Configuration pour l'envoi de documents via le protocole de transport HTTP», à la page 190  «Configuration de documents entrants via le protocole de transport HTTP», à la page 191
WebSphere Message Broker	Les flux de messages permettant de traiter les documents SOAP sont très similaires à ceux destinés aux documents non-SOAP via HTTP. Un seul noeud de transport supplémentaire est requis pour le traitement des documents SOAP.	Pour envoyer un document SOAP à Message Broker, voir tableau 80, à la page 198.  Pour recevoir un document SOAP de Message Broker, voir «Création du flux de messages pour le transport HTTP», à la page 195.

Pour que Message Broker traite correctement un document SOAP envoyé par WebSphere Partner Gateway, le flux de messages doit contenir un noeud HTTPRequest permettant de traiter la communication avec le client de services Web. Le tableau 80 répertorie les noeuds d'un flux de messages Message Broker qui sont requis pour le traitement d'un document SOAP envoyé par WebSphere Partner Gateway.

Tableau 80. Noeuds permettant l'envoi de documents SOAP à WebSphere Message Broker

Type de noeud	Objet	Remarques
HTTPInput	Reçoit le document WebSphere Partner Gateway demandé dans le flux de messages	<p>Paramétrez la zone de sélection d'URL de ce noeud de transport (dans les propriétés de base) sur l'adresse URL à laquelle WebSphere Partner Gateway envoie ses documents (adresse URL configurée dans la destination de WebSphere Partner Gateway). L'adresse URL doit être au format suivant :</p> <p><code>http://nomHôte:port/chemin</code></p> <p>où <i>nomHôte</i> représente le nom de l'ordinateur sur lequel réside WebSphere Partner Gateway, <i>port</i> le numéro du port HTTP sur lequel le récepteur de WebSphere Partner Gateway est en mode écoute et <i>chemin</i> l'emplacement de cet ordinateur.</p> <p>Pour plus d'informations, voir «Configuration pour l'envoi de documents via le protocole de transport HTTP», à la page 190.</p>
Calcul	Effectue des tâches de logique applicative, telles que la mise à jour des informations d'en-tête	Utilisez ESQL pour effectuer la logique applicative. Le noeud de traitement envoie le message résultant au noeud HTTPReply.
HTTPRequest	Envoie la requête SOAP au fournisseur de service Web externe (WebServices) et reçoit une réponse de ce service Web.	Aucun
HTTPReply	Renvoie une réponse à WebSphere Partner Gateway	Par défaut, ce noeud envoie le message de sortie au client.

## Utilisation du protocole de transport JMS avec WebSphere Message Broker

Cette section explique comment configurer les composants pour échanger des documents entre WebSphere Partner Gateway et WebSphere Message Broker via le protocole de transport JMS. Elle contient les informations suivantes qui expliquent comment envoyer et recevoir des documents :

- «Composants requis pour les documents envoyés via JMS»
- «Envoi de documents via le transport JMS», à la page 170
- «Réception de documents via le transport JMS», à la page 172

### Composants requis pour les documents envoyés via JMS

Pour envoyer des recevoir des documents entre WebSphere Partner Gateway et la version 6.0 de Message Broker via le protocole de transport JMS, vous devez utiliser WebSphere MQ comme fournisseur JMS. Les sections qui suivent expliquent comment WebSphere Partner Gateway, Message Broker et WebSphere MQ collaborent pour échanger des documents via le protocole de transport HTTP.

#### Envoi de documents via le transport JMS

Pour que WebSphere Partner Gateway envoie un document à Message Broker par le biais du protocole JMS, vous devez utiliser les files d'attente JMS. WebSphere Partner Gateway envoie un document à sa file d'attente de sortie JMS d'où il est transféré, vers la file d'attente d'entrée JMS sur laquelle Message Broker est en mode écoute. Lorsque Message Broker reçoit un document, il l'extrait de sa file

d'attente d'entrée. Le flux de messages de Message Broker contient des noeuds de transport WebSphere MQ (JMS) spéciaux qui traitent l'accès aux files d'attente JMS. La figure 38 illustre comment WebSphere Partner Gateway envoie des documents à Message Broker via le protocole JMS.

### WebSphere Business Integration Message Broker

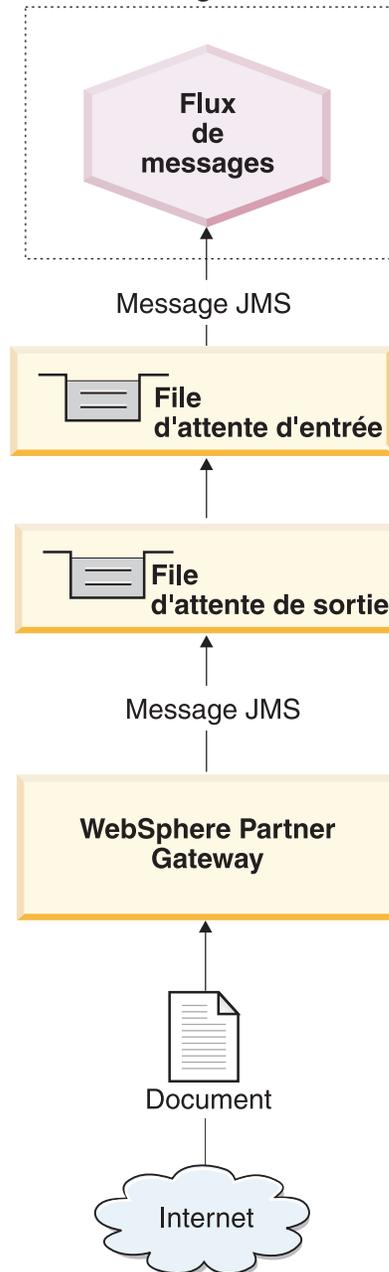


Figure 38. Flux de messages entre WebSphere Partner Gateway et un flux de messages via le protocole de transport JMS

Les étapes suivantes expliquent comment WebSphere Partner Gateway envoie un document à un flux de messages dans Message Broker via le protocole de transport JMS :

1. WebSphere Partner Gateway place un message dans sa file d'attente de sortie JMS.

Si le type d'emballage du document était Backend Integration, WebSphere Partner Gateway a intégré des propriétés personnalisées dans ce message. L'en-tête de message JMS, JMSType, est défini avec le type de contenu des données utiles.

**Remarque :** Dans WebSphere Partner Gateway, vous devez configurer une destination qui identifie la file d'attente de sortie JMS à laquelle WebSphere Partner Gateway envoie le message et dans laquelle Message Broker effectue des interrogations. Pour plus d'informations, voir «Configuration pour la réception de documents via le protocole de transport JMS», à la page 190.

2. WebSphere MQ transfère le document de la file d'attente de sortie de l'ordinateur sur lequel réside WebSphere Partner Gateway vers la file d'attente d'entrée dans laquelle Message Broker effectue des interrogations.
3. Lorsque Message Broker détecte un message dans la file d'attente d'entrée, il extrait le message et l'envoie au flux de messages approprié.  
Pour plus d'informations sur la procédure de configuration de cette file d'attente, voir «Configuration de l'environnement requis pour le transport JMS», à la page 202.
4. Le noeud MQInput envoie le document au noeud suivant du flux de messages. Ce noeud est généralement un noeud de traitement.
5. Les noeuds du flux de messages exécutent la logique applicative.  
Une fois la logique applicative est terminée, le flux de messages envoie le document résultant à son noeud MQOutput.
6. Le noeud MQOutput envoie le document à la file d'attente appropriée.

### **Réception de documents via le transport JMS**

Pour que WebSphere Partner Gateway reçoive un document de Message Broker par le biais du protocole JMS, vous devez utiliser des files d'attente JMS. Message Broker envoie un document à sa file d'attente de sortie JMS d'où il est transféré, vers la file d'attente d'entrée JMS sur laquelle WebSphere Partner Gateway est en mode écoute. Lorsque WebSphere Partner Gateway reçoit un document, il l'extrait de sa file d'attente d'entrée. Le flux de messages de Message Broker contient des noeuds de transport WebSphere MQ (JMS) spéciaux qui traitent l'accès aux files d'attente JMS. La figure 39 illustre la façon dont les documents sont envoyés de Message Broker à WebSphere Partner Gateway.

## WebSphere Business Integration Message Broker

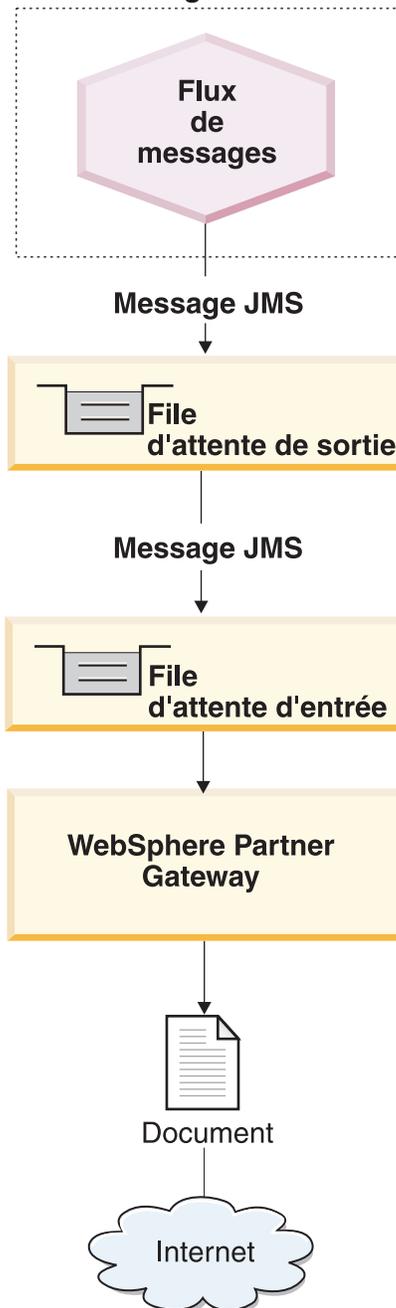


Figure 39. Flux de messages entre un flux de messages et WebSphere Partner Gateway via le protocole de transport JMS

Les étapes suivantes expliquent comment WebSphere Partner Gateway reçoit un document à partir d'un flux de messages dans Message Broker via le protocole de transport JMS :

1. Le flux de messages dans Message Broker reçoit un document dans son noeud MQInput.  
Le flux de messages reçoit sont message entrant d'une file d'attente d'entrée JMS.

2. La file d'attente MQInput du flux de messages reçoit le document et l'envoie au noeud suivant du flux de messages. Ce noeud est généralement un noeud de traitement.
3. Les noeuds du flux de messages exécutent la logique applicative.  
Une fois la logique applicative est terminée, le flux de messages envoie le document résultant à son noeud MQOutput.
4. Le noeud MQOutput envoie le document à la file d'attente de sortie JMS appropriée.
5. WebSphere MQ transfère le document de la file d'attente de l'ordinateur sur lequel réside Message Broker vers la file d'attente dans laquelle WebSphere Partner Gateway effectue des interrogations.
6. WebSphere Partner Gateway reçoit le message à partir de sa file d'attente d'entrée JMS, telle qu'elle est configurée dans son récepteur.  
Pour plus d'informations sur le récepteur, voir «Configuration de documents entrants via le protocole de transport JMS», à la page 192. Pour plus d'informations sur la procédure de configuration de cette file d'attente, voir «Configuration de l'environnement requis pour le transport JMS».

## Configuration de l'environnement requis pour le transport JMS

L'échange de documents sur Message Broker nécessite des files d'attente JMS (éloignées et locales). Pour plus d'informations sur la manière de configurer WebSphere Partner Gateway pour une utilisation avec Message Broker via JMS, voir «Configuration de WebSphere Partner Gateway pour WebSphere Message Broker», à la page 189. Pour utiliser le protocole JMS avec Message Broker, vous pouvez configurer le système JMS fourni par WebSphere MQ. La version 6.0 de Message Broker utilise la version 6.0 de WebSphere MQ comme fournisseur JMS. Vous pouvez suivre les étapes du *Guide de configuration du concentrateur WebSphere Partner Gateway* pour configurer le mécanisme de protocole de transport JMS.

**Important :** Les opérations décrites dans le *Guide de configuration du concentrateur WebSphere Partner Gateway* doivent être effectuées sur l'ordinateur sur lequel WebSphere Partner Gateway est installé. Ce document présume que le mécanisme de transport JMS requis par Message Broker a déjà été configuré pendant l'installation de Message Broker.

Lors de la création de vos files d'attente JMS à utiliser entre WebSphere Partner Gateway et Message Broker, prenez en considération les points suivants :

- Une partie du processus d'installation de Message Broker nécessite la création des gestionnaires de files d'attente suivants :
  - Gestionnaire de files d'attente WebSphere MQ associé au domaine de courtier  
Vous pouvez utiliser la commande suivante pour créer ce gestionnaire de files d'attente et un ensemble de files d'attente nommées :  
`mqsicreatebroker`
  - Gestionnaire de files d'attente WebSphere MQ pour Message Broker  
Dans la mesure où Message Broker utilise un ensemble de noms de files d'attente prédéterminés, il nécessite un gestionnaire de files d'attente WebSphere MQ distinct pour chaque courtier. Message Broker peut partager cet hébergement de gestionnaire de files d'attente à l'aide de son gestionnaire de configuration et/ou du serveur de noms d'utilisateur facultatif.

Pour plus d'informations, voir *WebSphere Message Broker Installation and Configuration Guide*.

- Lorsque vous créez vos alias de file d'attente JMS, vous pouvez les nommer de manière à indiquer la direction du flux entre WebSphere Partner Gateway et Message Broker.

Par exemple, si vous créez les files d'attente répertoriées dans la colonne Nom de file d'attente d'origine du tableau 65, vous pouvez renommer ces files d'attente pour qu'elles indiquent la direction du flux, comme indiqué dans la colonne Nom de file d'attente directionnelle du tableau 81.

Tableau 81. Attribution de noms aux files d'attente JMS pour que WebSphere Message Broker donne des indications de direction

Nom de file d'attente d'origine	Nom de file d'attente directionnelle
inQ	MB2HUB
outQ	HUB2MB

## Création du flux de messages pour le transport JMS

Pour qu'un flux de messages Message Broker traite les documents via le protocole de transport JMS, il utilise les noeuds de transport suivants :

- MQInput
- MQOutput

L'ordre dans lequel ces noeuds de transport sont utilisés est fonction du sens de la communication :

- Lorsque WebSphere Partner Gateway *envoie* un document à Message Broker, le flux de messages inclut les types de noeuds du tableau 82 (dans l'ordre indiqué) pour décrire la logique applicative.
- Lorsque WebSphere Partner Gateway *reçoit* un document de Message Broker, le flux de messages inclut les types de noeuds du tableau 83 (dans l'ordre indiqué) pour décrire la logique applicative.

Tableau 82. Noeuds permettant l'envoi de documents à WebSphere Message Broker via JMS

Type de noeud	Objet	Remarques et limitations
MQInput	Reçoit le document de WebSphere Partner Gateway	La valeur de la zone Nom de file d'attente (dans les propriétés de base) de ce noeud de transport correspond à la file d'attente d'entrée du flux de messages. WebSphere MQ doit être configuré de sorte que cette file d'attente JMS reçoive des documents de la file d'attente de sortie de WebSphere Partner Gateway. Pour plus d'informations, voir «Configuration pour la réception de documents via le protocole de transport JMS», à la page 190.
Calcul	Effectue des tâches de logique applicative, telles que la suppression des informations d'en-tête	Aucun
MQOutput	Reçoit le document du noeud de traitement et l'envoie sous forme de sortie de flux de messages	Ce noeud de transport envoie le document résultant à une file d'attente de sortie JMS qui le route vers sa prochaine destination.

Tableau 83. Noeuds permettant la réception de documents à partir de WebSphere Message Broker

Type de noeud	Objet	Remarques et limitations
MQInput	Reçoit le document dans le flux de messages	Ce noeud de transport reçoit le document entrant d'une file d'attente d'entrée JMS.
Calcul	Effectue des tâches de logique applicative, telles que la mise à jour des informations d'en-tête	Aucun
MQOutput	Reçoit le document du noeud de traitement et l'envoie à WebSphere Partner Gateway	La valeur de la zone Nom de file d'attente (dans les propriétés de base) de ce noeud de transport correspond à la file d'attente de sortie du flux de messages. WebSphere MQ doit être configuré de sorte que cette file d'attente JMS envoie des documents à la file d'attente d'entrée de WebSphere Partner Gateway. Pour plus d'informations, voir «Configuration de documents entrants via le protocole de transport JMS», à la page 192.

Pour plus de détails sur la création et la configuration des noeuds de flux de messages, consultez la documentation WebSphere Message Broker.

---

## Chapitre 12. Intégration à WebSphere Data Interchange

Le présent chapitre décrit le processus d'intégration de WebSphere Partner Gateway à WebSphere Data Interchange.

**Remarque :** Pour obtenir une description du processus général utilisé pour intégrer WebSphere Partner Gateway à un système dorsal, voir Chapitre 2, «Planification de l'intégration dorsale», à la page 9.

---

### A qui s'adresse ce chapitre ?

WebSphere Partner Gateway inclut maintenant des fonctionnalités de traitement EDI (telles que le développement et la transformation) similaires à celles de WebSphere Data Interchange. Vous pourrez peut-être utiliser ces fonctionnalités EDI à la place de WebSphere Data Interchange. Dans ce cas, vous n'aurez pas besoin des informations fournies dans ce chapitre. Voir le *Guide de configuration du concentrateur WebSphere Partner Gateway* pour plus d'informations sur la configuration du concentrateur pour traitement des documents EDI.

Ainsi, ce chapitre s'adresse aux personnes qui :

- utilisaient une version précédente de WebSphere Partner Gateway (précédemment appelé WebSphere Business Integration Connect) et souhaitent utiliser WebSphere Data Interchange pour l'intégration ;
- requièrent les fonctionnalités avancées de WebSphere Data Interchange qui ne sont pas incluses dans la prise en charge EDI de WebSphere Partner Gateway.

Si vous souhaitez utiliser WDI, suivez les étapes décrites dans le présent chapitre. Tenez compte du fait que les étapes de la configuration EDI ont été modifiées dans la version 6.0 de WebSphere Partner Gateway. Désormais, vous sélectionnez un flux de document spécifique pour la version EDI utilisée. Par exemple, si vous échangez des documents EDI-X12, sélectionnez ISA (au lieu de ALL) pour le flux de document.

---

### Ressources pouvant être utilisées dans ce chapitre

Ce chapitre fournit une explication du processus par lequel les documents sont échangés puis répertorie les étapes de configuration d'un exemple d'environnement pour de tels échanges. Le scénario utilisé tout au long de ce chapitre est identique à celui présenté dans le référentiel *Intégration de WebSphere Data Interchange V3.2 à WebSphere Business Integration Connect V4.2*, disponible sur le site Web suivant :

[www.ibm.com/developerworks/websphere/](http://www.ibm.com/developerworks/websphere/)

Le référentiel fournit des scripts supplémentaires (dans la section relative à la configuration de WebSphere MQ) ainsi que des exemples de mappes de transformation. Grâce à ce référentiel, vous pouvez définir l'environnement décrit dans ce chapitre.

**Remarque :** Le tutoriel décrit l'intégration à WebSphere Business Integration Connect version 4.2. Les mêmes étapes, légèrement modifiées, s'appliquent également à WebSphere Partner Gateway.

Vous êtes supposé connaître WebSphere Data Interchange et son utilisation. Voir la documentation de WebSphere Data Interchange pour plus d'informations au cours de la lecture de ce chapitre.

---

## Introduction

WebSphere Data Interchange intègre l'échange de données informatisé (EDI) au processus métier, à la messagerie et aux fonctions business-to-business basées sur Internet de WebSphere. Vous pouvez alors échanger des documents et des messages entre WebSphere Partner Gateway et WebSphere Data Interchange via le protocole de transport JMS. Vous ne devez spécifier aucun emballage lors de l'envoi d'un document à WebSphere Data Interchange.

**Remarque :** WebSphere Data Interchange fournit d'autres types d'option d'intégration, tels que l'intégration basée sur les fichiers. Pour plus d'informations sur l'activation de l'échange de documents via l'intégration basée sur les fichiers, voir la documentation de WebSphere Data Interchange.

## Envoi de documents à WebSphere Data Interchange

Les étapes suivantes déterminent le processus d'envoi de documents EDI par WebSphere Partner Gateway à WebSphere Data Interchange :

1. Un partenaire externe envoie un document EDI à WebSphere Partner Gateway. Le document est envoyé avec un emballage spécifique via un protocole de transport (dans cet exemple, emballage AS2 via HTTP). WebSphere Partner Gateway dépouille le document EDI de son emballage AS2.
2. WebSphere Partner Gateway place le document EDI dans une file d'attente.

**Remarque :** WebSphere Partner Gateway détermine le protocole utilisé dans le document en étudiant les trois premiers caractères du document EDI. Il détermine ensuite, à partir du type de protocole, les informations relatives à l'expéditeur et au récepteur. Pour plus d'informations, voir «Présentation du routage EDI», à la page 249.

3. WebSphere Data Interchange lit le document EDI dans la file d'attente. Il procède aux opérations de désencapsulation, de validation et de conversion du document EDI.

**Remarque :** WebSphere Data Interchange doit être configuré avec les mappages nécessaires, les profils de partenaires d'échanges et d'autres informations. Pour plus d'informations, voir la documentation de WebSphere Data Interchange.

4. WebSphere Data Interchange transfère le document à un système dorsal. Si ce système dorsal est WebSphere InterChange Server, WebSphere Data Interchange envoie le document à WebSphere Business Integration Adapter for MQ de sorte qu'un objet métier puisse être créé et qu'une collaboration soit appelée au sein d'InterChange Server.

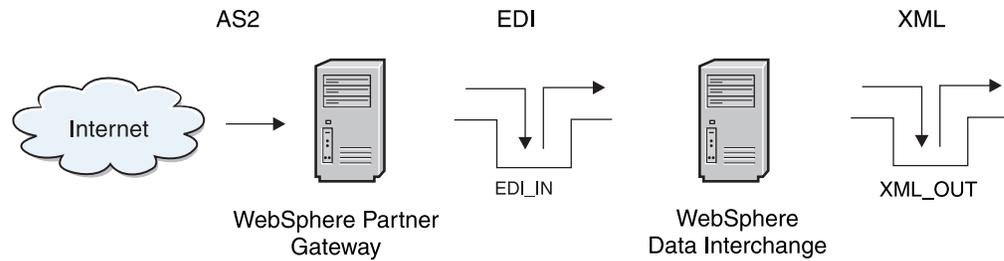


Figure 40. Document EDI provenant de WebSphere Partner Gateway

Dans la figure 40, un partenaire externe envoie un document EDI dans un empaquetage AS à WebSphere Partner Gateway, qui, à son tour, envoie le document dans la file d'attente EDI\_IN résidant sur WebSphere Data Interchange. Notez que la file d'attente éloignée, la file d'attente de transmission, la file d'attente de réception (dans l'exemple, EDI\_IN) et les canaux émetteur et récepteur doivent être configurés de sorte que les messages envoyés à WebSphere Partner Gateway soient transmis dans la file d'attente EDI\_IN. WebSphere Data Interchange Server extrait le document EDI, recherche les profils utilisateur, les mappages et autres, convertit le document au format XML puis le place dans la file d'attente XML\_OUT.

## Réception de documents de WebSphere Data Interchange

Les étapes suivantes déterminent le processus de réception par WebSphere Partner Gateway de documents EDI provenant de WebSphere Data Interchange :

1. WebSphere Data Interchange place le document EDI dans une file d'attente.
2. WebSphere Partner Gateway lit le message depuis la file d'attente.

**Remarque :** WebSphere Partner Gateway détermine le routage du document comme indiqué dans «Présentation du routage EDI», à la page 249.

3. WebSphere Partner Gateway route le document vers le partenaire externe approprié.

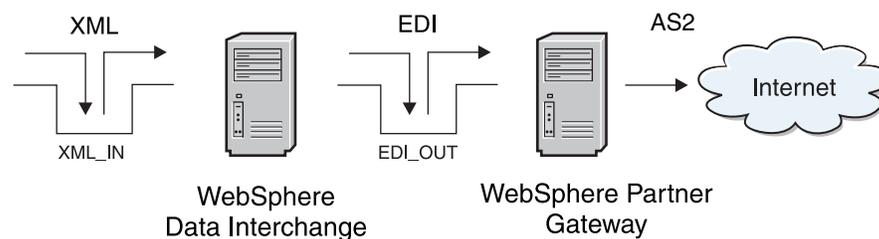


Figure 41. Envoi d'un document EDI à WebSphere Partner Gateway

Dans la figure 41, un document XML est placé dans la file d'attente XML\_IN pour que WebSphere Data Interchange le convertisse. Les profils utilisateurs, les mappages et autres sont supposés avoir déjà été réalisés. Dès réception d'un document XML valide, WebSphere Data Interchange le convertit au format EDI et place le document résultant dans la file d'attente EDI\_OUT (file d'attente éloignée). La file d'attente de transmission, les canaux émetteur et récepteur et la file d'attente de réception sont également supposés être configurés du côté de WebSphere Partner Gateway. Dès réception du document, WebSphere Partner Gateway le route vers le partenaire externe.

## Exemple de scénario utilisé dans ce chapitre

Tout au long de ce chapitre, vous allez étudier les étapes à suivre pour la configuration de l'échange de documents EDI entre deux partenaires d'échanges. Les documents EDI sont envoyés sur Internet et le protocole AS2 (via HTTP) est utilisé en tant que protocole de communication.

Dans cet exemple, les partenaires d'échanges sont désignés sous les noms de Partenaire numéro un et Partenaire numéro deux. La figure 42 illustre la configuration des deux partenaires.

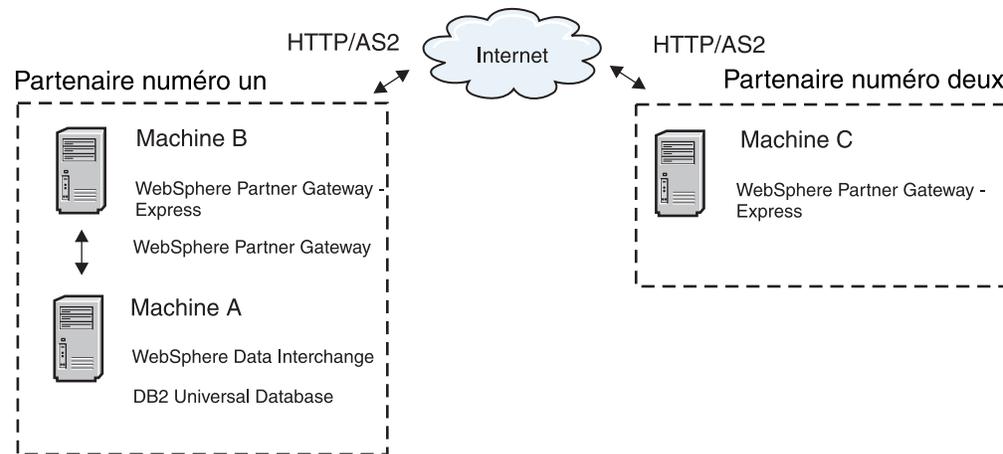


Figure 42. Configuration de deux partenaires dans un exemple de scénario

Les trois ordinateurs sont dotés des logiciels suivants :

- L'ordinateur A contient WebSphere Data Interchange Server 3.2 et WebSphere Data Interchange Client 3.2 avec les logiciels prérequis.
- L'ordinateur B contient WebSphere Partner Gateway Enterprise Edition avec ses logiciels prérequis.
- L'ordinateur C contient WebSphere Partner Gateway - Express.

Voir le *Guide d'installation de WebSphere Partner Gateway* et la documentation de WebSphere Data Interchange pour obtenir une liste complète des logiciels prérequis.

Dans cet exemple, Partenaire numéro un fait fonctionner deux ordinateurs. WebSphere MQ et WebSphere Data Interchange Server sont tous les deux installés sur l'ordinateur A. Sur l'ordinateur B, ce sont WebSphere MQ ainsi que WebSphere Partner Gateway Enterprise Edition qui sont installés. L'ordinateur B prend en charge les communications entre les deux partenaires de l'échange.

WebSphere Data Interchange prend en charge l'intégration à WebSphere MQ par l'activation de l'interopération avec une grande quantité d'applications d'entreprise et de moteurs de processus métier. WebSphere Partner Gateway utilise WebSphere MQ en tant que fournisseur JMS. Ainsi, l'intégration entre WebSphere Data Interchange et WebSphere Partner Gateway s'effectue via des messages MQ destinés aux clients de l'API JMS.

WebSphere Partner Gateway est utilisé pour communiquer les transactions EDI sur Internet à l'aide du protocole AS2.

Notez que, dans cet exemple, Partenaire numéro deux utilise WebSphere Partner Gateway - Express pour accepter les transactions via AS2 et dispose de son propre environnement WebSphere Data Interchange pour la gestion des conversions et des accusés réception.

Tout au long de ce chapitre, vous allez étudier les détails de la configuration des ordinateurs utilisés dans cet exemple de scénario. Le flux de messages est bidirectionnel : les artefacts de réception et d'envoi sont donc inclus.

---

## Planification de l'intégration à WebSphere Data Interchange

### Versions de WebSphere Data InterChange prises en charge par WebSphere Partner Gateway

La version prise en charge de WebSphere Data Interchange est la version 3.2 incluant le correctif CSD 22. WebSphere MQ est utilisé pour l'interface avec WebSphere Data Interchange. La version prise en charge de WebSphere MQ est la version 6.0 incluant le dernier groupe de correctifs.

---

## Configuration de votre environnement pour l'échange de messages

Effectuer les configurations et paramétrages suivants pour établir une communication entre WebSphere Data Interchange et WebSphere Partner Gateway :

- «Configuration de l'intercommunication WebSphere MQ»
- «Configuration de WebSphere Data Interchange», à la page 211
- «Configuration de l'environnement JMS», à la page 216
- «Configuration de WebSphere Partner Gateway Enterprise Edition», à la page 217

### Configuration de l'intercommunication WebSphere MQ

La première étape de la configuration de l'environnement consiste à configurer l'intercommunication WebSphere MQ. Le terme intercommunication signifie envoyer des messages d'un gestionnaire de files d'attente à un autre. La première étape consiste à définir un gestionnaire de files d'attente (et ses objets associés) pour les systèmes WebSphere Data Interchange et WebSphere Partner Gateway. Si vous avez l'intention d'envoyer des messages dans les deux directions, vous devez définir un gestionnaire de files d'attente source et un gestionnaire de files d'attente de réception sur les deux systèmes. Sur le gestionnaire de files d'attente source, définissez un canal émetteur, une définition de file d'attente éloignée et une file d'attente de transmission. Sur le gestionnaire de files d'attente de réception, vous devez définir un canal récepteur et une file d'attente de réception.

**Remarque :** Voir la documentation de WebSphere MQ pour plus d'informations sur la définition de gestionnaires de files d'attente.

Cette section indique les valeurs à utiliser pour définir les gestionnaires de files d'attente et les objets associés nécessaires pour l'exemple de scénario. Dans ce scénario, WebSphere MQ V5.3 est installé sur les ordinateurs A et B. La première étape consiste ici à créer un gestionnaire de files d'attente sur chacun des ordinateurs A et B pour l'utiliser avec WebSphere Data Interchange et WebSphere Partner Gateway Enterprise Edition respectivement.

**Remarque :** Votre gestionnaire de files d'attente WebSphere Data Interchange doit être configuré pour lancer WebSphere Data Interchange Server à l'aide de l'application WDI Adapter.

- Sur l'ordinateur A, vous devez utiliser le gestionnaire de files d'attente défini pour être utilisé avec WebSphere Data Interchange. Tout au long de ce chapitre, ce gestionnaire de files d'attente est appelé WDI32\_QM.
- Sur l'ordinateur B, vous devez utiliser le gestionnaire de files d'attente créé lors de l'installation et la configuration initiales de WebSphere Partner Gateway Enterprise Edition. Tout au long de ce chapitre, ce gestionnaire de files d'attente est appelé HUB\_QM.

Pour envoyer des messages depuis un gestionnaire de files d'attente vers un autre à l'aide de WebSphere MQ, vous devez définir les objets suivants :

- Sur le gestionnaire de files d'attente source :
  - un canal émetteur ;
  - une définition de file d'attente éloignée ;
  - une file d'attente de transmission ;
- Sur le gestionnaire de files d'attente de réception :
  - un canal récepteur ;
  - une file d'attente de réception ;

Dans l'exemple de scénario, les ordinateurs A et B jouent le rôle d'émetteur et de récepteur. Par conséquent, vous allez devoir définir un certain nombre d'objets sur chaque ordinateur.

Le tableau 84 répertorie les objets que vous devez créer pour configurer les ordinateurs A et B en tant que récepteur et émetteur.

*Tableau 84. Objets WebSphere MQ à créer*

Objet MQ WebSphere	Ordinateur A	Ordinateur B
Gestionnaire de files d'attente	WDI32_QM	HUB_QM
Canal émetteur	TO.HUB60	TO.WDI32
Canal récepteur	TO.WDI32	TO.HUB60
File d'attente éloignée	EDI_OUT_A	EDI_OUT_B
File d'attente de transmission	XMITQ_A	XMITQ_B
File d'attente locale	EDI_IN_A	EDI_IN_B
File d'attente locale	XML_IN_A	XML_IN_B
File d'attente locale	XML_OUT_A	XML_OUT_B

La figure 43 illustre le flux de messages entre l'ordinateur A et l'ordinateur B en indiquant le rôle des objets WebSphere MQ répertoriés dans le tableau 84.

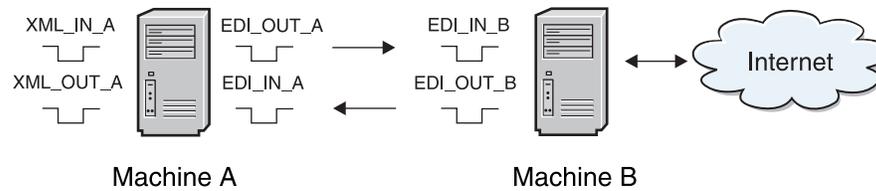


Figure 43. Flux de messages entre l'ordinateur A et l'ordinateur B

Vous pouvez utiliser plusieurs méthodes différentes pour définir ces objets, en fonction de votre plateforme WebSphere MQ. Par exemple, vous pouvez utiliser WebSphere MQ Explorer sur Windows pour définir les objets.

## Configuration de WebSphere Data Interchange

Pour permettre à WebSphere Data Interchange de recevoir des messages depuis la file d'attente WebSphere MQ et écrire des messages EDI sur une file d'attente, vous devez configurer des profils dans WebSphere Data Interchange Client. Grâce à WebSphere Data Interchange Client, vous pouvez créer les profils suivants, décrits dans les prochaines sections :

- profil de file d'attente MQ Series ;
- profil de réseau ;
- profil de boîte aux lettres ;
- profil de service.

Dans cet exemple de scénario, WebSphere Data Interchange reçoit des messages XML provenant de la file d'attente WebSphere MQ XML\_IN\_A et écrit le résultat de la conversion dans la file d'attente WebSphere MQ EDI\_OUT\_A. Cette procédure est connue sous le nom de "conversion XML en EDI". WebSphere Data Interchange reçoit également un message EDI provenant de WebSphere Partner Gateway Enterprise Edition dans la file d'attente WebSphere MQ EDI\_IN\_A et écrit le résultat de la conversion dans la file d'attente XML\_OUT\_A.

### Profil de file d'attente MQSeries

Un profil de file d'attente MQSeries contient des informations sur une file d'attente de messages WebSphere MQ. Le tableau 85 indique les propriétés à configurer pour chaque profil.

Tableau 85. Propriétés dans un profil de file d'attente MQSeries

Propriété MQ	Description
ID profil de file d'attente	Identificateur unique pour nommer le profil (nom logique)
Nom complet de la file d'attente	Nom réel de la file d'attente WebSphere MQ
Nom du gestionnaire de files d'attente	Nom réel du gestionnaire de files d'attente WebSphere MQ
Description	Toute chaîne identifiant l'objectif du profil
Longueur maximale	Longueur maximale autorisée du message pour la file d'attente en fonction de la configuration dans WebSphere MQ
Lectures destructives	Si cette propriété est sélectionnée, elle peut obliger WebSphere Data Interchange à supprimer le message de la file d'attente WebSphere MQ à la lecture.

Tableau 85. Propriétés dans un profil de file d'attente MQSeries (suite)

Propriété MQ	Description
Contrôle du point de synchronisation	Lorsque cette propriété est sélectionnée, la lecture et l'écriture des messages en file d'attente est sous contrôle du point de synchronisation. Si le contrôle du point de synchronisation est activé, les modifications apportées à une file d'attente de messages ne s'appliquent pas tant que WebSphere Data Interchange n'émet aucun point de synchronisation.

Dans la mesure où vous utilisez les files d'attente WebSphere MQ, vous avez besoin d'un profil de file d'attente MQSeries dans WebSphere Data Interchange pour *chaque* file d'attente. Dans tous les cas, vous devez créer quatre profils de file d'attente MQSeries, un pour chaque file d'attente WebSphere MQ utilisée dans le flux de messages. A partir de la zone d'installation de WebSphere Data Interchange Client, procédez comme suit :

1. Créez un profil de file d'attente MQSeries pour XML\_IN\_A et EDI\_OU\_A.

Le tableau 86 répertorie les paramètres réels spécifiés dans chaque profil de file d'attente MQSeries créé. Les files d'attente représentées ici sont utilisées avec la conversion XML en EDI.

Tableau 86. Profil de file d'attente MQSeries pour XML\_IN\_A et EDI\_OU\_A

Propriété de file d'attente	Valeur pour XML_IN_A	Valeur pour EDI_OU_A
ID profil de file d'attente	XML_IN_A	EDI_OU_A
Nom complet de la file d'attente	XML_IN_A	EDI_OUT_A
Nom du gestionnaire de files d'attente	WDI32_QM	WDI32_QM
Lectures destructives	Sélectionné	Sélectionné
Contrôle du point de synchronisation	Sélectionné	Sélectionné

**Remarque :** La propriété ID profil de file d'attente est limitée à huit caractères uniquement. Il faut donc renommer l'ID profil de la file d'attente EDI\_OUT\_A par EDI\_OU\_A. Toutes les références à la file d'attente WebSphere MQ EDI\_OUT\_A dans WebSphere Data Interchange utilisent le nom EDI\_OU\_A.

2. Créez un profil de file d'attente MQSeries pour EDI\_IN\_A et XML\_OU\_A. Le tableau 87 définit les propriétés de chaque file d'attente utilisée lors de la conversion EDI en XML.

Tableau 87. Profil de file d'attente MQSeries pour EDI\_IN\_A et XML\_OU\_A

Propriété de file d'attente	Valeur pour EDI_IN_A	Valeur pour XML_OU_A
ID profil de file d'attente	EDI_IN_A	XML_OU_A
Nom complet de la file d'attente	EDI_IN_A	XML_OUT_A
Nom du gestionnaire de files d'attente	WDI32_QM	WDI32_QM
Lectures destructives	Sélectionné	Sélectionné
Contrôle du point de synchronisation	Sélectionné	Sélectionné

## Profil de réseau

Les profils de réseau définissent pour WebSphere Data Interchange les caractéristiques des réseaux utilisés pour les communications avec les partenaires d'échanges. Pour ce scénario, vous devez créer et configurer un profil de réseau qui communique avec les files d'attente WebSphere MQ créées précédemment.

Le tableau 88 indique les propriétés à configurer pour chaque profil de réseau.

Tableau 88. Propriétés dans un profil de réseau

Propriété de réseau	Description
ID réseau	Identificateur unique pour nommer le profil
Routine de communication	Nom du programme qui génère les commandes réseau et appelle le programme réseau afin qu'il traite les commandes
Programme réseau	Programme appelé par la routine de communication pour le traitement des requêtes
Paramètres réseau	Paramètres requis par le programme réseau

Pour cet exemple de scénario, vous devez créer et configurer un profil de réseau qui communique avec les files d'attente WebSphere MQ créées précédemment (voir «Profil de file d'attente MQSeries», à la page 211), comme suit :

1. Créez un nouveau profil de réseau appelé HUB\_IN.

Ce profil de réseau est utilisé dans le scénario de conversion XML en EDI. Le tableau 89 répertorie les paramètres réels spécifiés pour ce profil HUB\_IN.

Tableau 89. Profil de réseau pour HUB\_IN

Propriété de réseau	Valeur du profil HUB_IN
ID réseau	HUB_IN
Routine de communication	VANIMQ
Programme réseau	EDIMQSR
Paramètres réseau	SENDMQ=EDI_OU_A RECEIVMQ=XML_IN_A

2. Créez un second profil de réseau appelé HUB\_OUT.

Ce profil de réseau est utilisé dans la conversion de messages EDI reçus de WebSphere Partner Gateway Enterprise Edition. Un second profil de réseau est requis car WebSphere Partner Gateway Enterprise Edition place les messages dans les files d'attente WebSphere MQ qui contiennent des en-têtes RFH2. Le tableau 90 répertorie les propriétés spécifiées pour ce HUB\_OUT.

Tableau 90. Profil de réseau pour HUB\_OUT

Propriété de réseau	Valeur du profil HUB_OUT
ID réseau	HUB_OUT
Routine de communication	VANIMQ
Programme réseau	EDIRFH2
Paramètres réseau	SENDMQ=XML_OU_A RECEIVMQ=EDI_IN_A

## Profil de boîte aux lettres

Les profils de boîte aux lettres contiennent les informations dont WebSphere Data Interchange a besoin pour identifier les personnes et les groupes de votre organisation qui reçoivent des documents à convertir. Le tableau 91 indique les propriétés à configurer pour chaque profil de boîte aux lettres.

Tableau 91. Propriétés dans un profil de boîte aux lettres

Propriété de boîte aux lettres	Description
ID boîte aux lettres	Identificateur unique pour nommer le profil
ID réseau	ID réseau du profil de réseau créé précédemment

Vous devez créer des profils de boîtes aux lettres pour chaque file d'attente WebSphere MQ afin d'identifier les personnes et les groupes de votre organisation, comme suit :

1. Créez un profil de boîte aux lettres pour chaque file d'attente WebSphere MQ utilisée.

Les paramètres réels utilisés dans chaque profil de boîte aux lettres sont indiqués dans le tableau 92.

Tableau 92. Profils de boîte aux lettres pour XML\_IN\_A et EDI\_OU\_A

Propriété de boîte aux lettres	Valeur pour XML_IN_A	Valeur pour EDI_OU_A
ID boîte aux lettres	XML_IN_A	EDI_OU_A
ID réseau	HUB_IN	HUB_IN
Fichier reçu	XML_IN_A	EDI_OU_A

2. Créez une seconde paire de boîtes aux lettres.

Le tableau 93 répertorie les propriétés spécifiées pour ces composants.

Tableau 93. Profils de boîte aux lettres pour EDI\_IN\_A et XML\_OU\_A

Propriété de boîte aux lettres	Valeur pour EDI_IN_A	Valeur pour XML_OU_A
ID boîte aux lettres	EDI_IN_A	XML_OU_A
ID réseau	HUB_OUT	HUB_OUT
Fichier reçu	EDI_IN_A	XML_OU_A

## Profil de service

L'objectif des profils de service est de vous permettre d'entrer une commande d'utilitaire et de définir tous les fichiers qui seront utilisés au cours de l'exécution de cette commande.

Pour l'exemple de scénario, suivre les étapes suivantes :

1. Créez un profil de service pour XML\_IN\_A. Les propriétés à définir sous l'onglet **Général** sont les suivantes :

- Chaînage de commandes continu : **On Success**
- Commande PERFORM :  
PERFORM TRANSFORM WHERE INFILE(XML\_IN\_A) SYNTAX(X)  
OUTTYPE(MQ)OUTFILE(EDI\_OU\_A)

Le tableau 94 répertorie les propriétés des fichiers communs.

Tableau 94. Fichiers communs pour XML\_IN\_A

Propriété des fichiers communs	Valeur
Fichier de suivi	..\trk\xml_in.trk
Fichier d'exception	..\xex\xml_in.xex
Fichier de travail	..\wrk\xml_in.wrk
Fichier de rapport	..\rpt\xml_in.rpt
Fichier de requête	..\qry\xml_in.qry

2. Entrez ce qui suit dans l'onglet **Fichiers de sortie** :

- Nom dans la commande : **EDI\_OU\_A**
- Nom du système de fichiers : **..\edi\edi\_out.txt**

**Remarque :** Le nom EDI\_OU\_A est utilisé au lieu du nom EDI\_OUT\_A en raison des restrictions de longueur de caractère.

3. Créez un second profil de service pour EDI\_IN\_A. Les propriétés à définir sous l'onglet **Général** sont les suivantes :

- Chaînage de commandes continu : **On Success**
- Commande PERFORM :  
PERFORM TRANSFORM WHERE INFILE(EDI\_IN\_A) SYNTAX(E)  
OUTTYPE(MQ) OUTFILE(XML\_OU\_A)

Le tableau 95 répertorie les propriétés des fichiers communs.

Tableau 95. Fichiers communs pour EDI\_IN\_A

Propriété des fichiers communs	Valeur
Fichier de suivi	..\trk\edi_in.trk
Fichier d'exception	..\xex\edi_in.xex
Fichier de travail	..\wrk\edi_in.wrk
Fichier de rapport	..\rpt\edi_in.rpt
Fichier de requête	..\qry\edi_in.qry

4. Entrez les informations suivantes dans l'onglet **Fichiers de sortie** :

- Nom dans la commande : **XML\_OU\_A**
- Nom du système de fichiers : **..\xml\xml\_out.txt**

**Remarque :** Le nom XML\_OU\_A est utilisé au lieu du nom XML\_OUT\_A en raison des restrictions de longueur de caractère. Cette restriction a été exclue du CSD10 de WebSphere Interchange Server.

## Importation et compilation de mappes de transformation de données

Après avoir créé les profils, comme indiqué dans la section précédente, vous pouvez importer n'importe quelle mappe nécessaire à la transformation de vos données. Vous pouvez ensuite compiler les mappes de transformation et définir une règle pour chacune d'elles. Pour réaliser ces tâches, vous devez utiliser WebSphere Data Interchange Client. Pour plus d'informations, voir la documentation de WebSphere Data Interchange.

## Configuration de l'environnement JMS

Comme indiqué précédemment dans ce chapitre, WebSphere Partner Gateway Enterprise Edition utilise l'implémentation WebSphere MQ de JMS pour l'intégration à WebSphere Data Interchange.

**Remarque :** Vous pouvez également utiliser LDAP ou WebSphere Application Server en tant que fournisseur JNDI.

Cette section indique les étapes de la création d'un environnement JMS sur l'ordinateur B :

- «Configuration de JMSAdmin»
- «Création d'objets JMS»

Les classes WebSphere MQ pour Java et les classes WebSphere MQ pour JMS sont générées dans WebSphere MQ pour Windows version 5.3.

### Configuration de JMSAdmin

Utilisez l'outil JMSAdmin disponible dans WebSphere MQ pour créer les objets JMS dans JNDI. Pour plus d'informations sur le mode de création du fichier de configuration par défaut JMSAdmin.config, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

Pour créer les objets JMS pour ce référentiel :

1. Pour utiliser un fournisseur JNDI basé sur les fichiers, vous devez veiller à ce que le fichier JMSAdmin.config contienne les lignes indiquées ci-dessous :

```
INITIAL_CONTEXT_FACTORY=com.sun.jndi.fscontext.RefFSContextFactory
PROVIDER_URL=file:/opt/mqm/java/JNDI
```

2. Si le répertoire JNDI n'existe pas déjà, créez-le sous le répertoire suivant :  
/opt/mqm/java/bin

Avant d'appeler l'outil JMSAdmin, vous devez veiller à ce que votre variable CLASSPATH contienne les entrées suivantes :

```
/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
```

**Remarque :** vous obtenez les entrées ci-dessus, associées à Linux, si vous utilisez un fournisseur JNDI basé sur les fichiers.

### Création d'objets JMS

Pour créer les objets JMS requis, vous devez utiliser l'outil JMSAdmin. Pour l'exemple de scénario, procédez comme suit :

1. Définissez un nouveau contexte :  
DEF CTX(WdiJms)
2. Accédez au nouveau contexte :  
CHG CTX(WdiJms)
3. Définissez une fabrique de connexions de file d'attente :  
DEF QCF(HUB60\_QM\_QCF) TRAN(CLIENT) HOST(IP\_COMPUTER\_B)  
PORT(9999) CHAN(java.channel) QMANAGER(HUB60\_QM)

4. Définissez la file d'attente EDI\_IN\_B :  
DEF Q(EDI\_IN\_B) QMANAGER(HUB60\_QM) QUEUE(EDI\_IN\_B)
5. Définissez la file d'attente EDI\_OUT\_B :  
DEF Q(EDI\_OUT\_B) QMANAGER(HUB60\_QM) QUEUE(EDI\_OUT\_B)
6. Quittez la session JMSAdmin :  
END

## Configuration de WebSphere Partner Gateway Enterprise Edition

WebSphere Partner Gateway correspond à la couche de communication entre les divers partenaires externes et les processus internes. Lors de la configuration de WebSphere Partner Gateway pour l'utilisation de documents EDI, vous pouvez faire en sorte qu'il procède aux opérations suivantes :

- envoyer et recevoir des documents EDI vers et depuis WebSphere Data Interchange ;
- communiquer les transactions EDI aux partenaires d'échanges externes à l'aide du protocole AS2.

Le *Guide de configuration du concentrateur WebSphere Partner Gateway* fournit des informations complètes sur la configuration de WebSphere Partner Gateway Enterprise et Advanced Editions. Cette section fournit un exemple de la configuration de WebSphere Partner Gateway Enterprise Edition décrite dans l'exemple de scénario. Elle décrit les étapes suivantes :

1. «Création de partenaires»
2. «Configuration des fonctions business-to-business», à la page 219
3. «Création de destinations», à la page 220
4. «Définition d'interactions», à la page 222
5. «Création de connexions de partenaire», à la page 223
6. «Création de récepteurs», à la page 223

**Remarque :** Pour plus d'informations sur la configuration de WebSphere Partner Gateway - Express, voir «Configuration de WebSphere Partner Gateway - Express», à la page 224.

### Création de partenaires

Un profil de partenaire identifie les sociétés auprès du système. Créez des partenaires pour Partenaire numéro un et Partenaire numéro deux dans la console de communauté de WebSphere Partner Gateway Enterprise Edition.

**Création d'un partenaire pour Partenaire numéro un :** Créez un profil de partenaire pour représenter les ordinateurs A et B, qui correspondent aux deux systèmes appartenant à Partenaire numéro un.

Pour créer ce profil de partenaire, procédez comme suit :

1. Ouvrez la console de communauté de WebSphere Partner Gateway.
2. Ouvrez une session en tant que **hubadmin** (administrateur du concentrateur).
3. Cliquez sur **Administrateur du compte > Profils > Partenaire**.
4. Cliquez sur **Créer** et entrez les informations comme indiqué dans le tableau 96, à la page 218 ci-dessous.

Tableau 96. Propriétés du partenaire (Partenaire numéro un)

Nom de zone	Valeur
Nom de connexion de l'entreprise	partnerOne
Nom affiché du partenaire	Partenaire numéro un
Type de partenaire	Partenaire interne
Etat	Activé
Type de fournisseur	Autre
Site Web	http://IP_COMPUTER_A où IP_COMPUTER_A représente l'adresse IP de l'ordinateur A
Type d'ID entreprise	Format libre
Identificateur d'ID entreprise	123456789
Type de destination de l'adresse IP	Production
Adresse IP	IP_COMPUTER_A où IP_COMPUTER_A représente l'adresse IP de l'ordinateur A

**Remarque :** Pour créer le type d'ID entreprise et l'identificateur d'ID entreprise, vous devez d'abord cliquer sur le bouton **Nouveau** sous l'ID entreprise. L'ID entreprise doit être unique. De la même manière, pour indiquer d'autres informations associées à l'adresse IP, vous devez cliquer sur le bouton **Nouveau** sous l'en-tête Adresse IP.

5. Cliquez sur **Sauvegarder**.

WebSphere Partner Gateway Enterprise Edition utilise la propriété Identificateur d'ID entreprise (définie dans le tableau 96) afin d'identifier l'expéditeur ou le récepteur d'un document. Lorsqu'une transaction ANSI X12 EDI est reçue, les données de l'expéditeur et du récepteur de l'échange sont lues pour déterminer la source et la cible de la transaction.

**Important :** Notez le mot de passe de l'administrateur pour le partenaire numéro un, vous en aurez besoin plus tard. Lorsque vous ouvrirez une session sur la console de communauté en tant que Partenaire numéro un, il vous sera demandé de saisir ce mot de passe puis de le modifier.

**Création d'un partenaire pour Partenaire numéro deux :** Ensuite, créez un partenaire externe pour représenter Partenaire numéro deux. Pour créer ce partenaire, procédez comme suit :

1. Cliquez sur **Administrateur du compte > Profils > Partenaire**.
2. Cliquez sur **Créer**.
3. Entrez les valeurs indiquées dans le tableau 97 ci-dessous.

Tableau 97. Propriétés du partenaire (Partenaire numéro deux)

Nom de zone	Valeur
Nom de connexion de l'entreprise	partnerTwo
Nom affiché du partenaire	Partenaire numéro deux
Type de partenaire	Partenaire
Etat	Activé

Tableau 97. Propriétés du partenaire (Partenaire numéro deux) (suite)

Nom de zone	Valeur
Type de fournisseur	Autre
Site Web	http://IP_COMPUTER_C où IP_COMPUTER_C représente l'adresse IP de l'ordinateur C
Type d'ID entreprise	Format libre
Identificateur d'ID entreprise	987654321
Type de destination de l'adresse IP	Production
Adresse IP	IP_COMPUTER_C où IP_COMPUTER_C représente l'adresse IP de l'ordinateur C

4. Cliquez sur **Sauvegarder**.

**Important :** Notez le mot de passe de l'administrateur pour le partenaire numéro deux, vous en aurez besoin plus tard. Lorsque vous ouvrirez une session sur la console de communauté en tant que Partenaire numéro deux, il vous sera demandé de saisir ce mot de passe puis de le modifier.

### Configuration des fonctions business-to-business

Vous pouvez définir les fonctions business-to-business de chaque partenaire dans WebSphere Partner Gateway Enterprise Edition via la console de communauté. Une fois les fonctions business-to-business définies pour les partenaires, vous pouvez établir une définition de flux de document permettant de prendre en charge des types de collaboration métier spécifiques entre ces partenaires.

**Définition des fonctions business-to-business de Partenaire numéro un :** Pour définir les fonctions business-to-business de Partenaire numéro un, procédez comme suit :

1. Cliquez sur **Administrateur du compte > Profils > Partenaire**.
2. Cliquez sur **Rechercher** pour afficher la liste de tous les partenaires définis dans le système.
3. Cliquez sur l'icône **Afficher les détails** en regard de **Partenaire numéro un**, puis cliquez sur **Fonctions business-to-business**.

Pour activer les fonctions business-to-business, il suffit de cliquer sur l'icône **Rôle inactif**. Pour les besoins de cet exemple, seules les fonctions business-to-business nécessaires à l'implémentation du scénario seront configurées.

Pour définir l'empaquetage source et cible de Partenaire numéro un sur None (Aucun), procédez comme suit :

1. Cliquez sur l'icône **Rôle inactif** sous **Définition de la source pour l'empaquetage : None** pour l'activer. Répétez cette opération avec **Définition de la cible**.
2. Cliquez sur l'icône **Développer** pour passer au niveau inférieur.
3. Cliquez sur l'icône **Rôle inactif** pour **Protocole : EDI-X12 (ALL)** pour la source et la cible.
4. Cliquez sur l'icône **Développer**.

5. Cliquez sur l'icône **Rôle inactif** pour **Flux de document : ISA (ALL)** pour la source et la cible.

#### **Définition des fonctions business-to-business pour Partenaire numéro deux :**

Pour définir les fonctions business-to-business de Partenaire numéro deux, procédez comme suit :

1. Cliquez sur **Administrateur du compte > Profils > Partenaire**.
2. Cliquez sur **Rechercher** pour afficher la liste de tous les partenaires définis dans le système.
3. Cliquez sur l'icône **Afficher les détails** en regard de **Partenaire numéro deux**, puis cliquez sur **Fonctions business-to-business**.

Pour définir l'empaquetage source et cible de Partenaire numéro deux sur AS, procédez comme suit :

1. Cliquez sur l'icône **Rôle inactif** sous **Définition de la source** pour **Empaquetage : AS** pour l'activer. Répétez cette opération avec **Définition de la cible**.
2. Cliquez sur l'icône **Développer** pour passer au niveau inférieur.
3. Cliquez sur l'icône **Rôle inactif** pour **Protocole : EDI-X12 (ALL)** pour la source et la cible.
4. Cliquez sur l'icône **Développer**.
5. Cliquez sur l'icône **Rôle inactif** pour **Flux de document : ISA (ALL)** pour la source et la cible.

Ensuite, vous pouvez mettre à jour la définition AS de Partenaire numéro deux pour vous assurer que les notifications de disponibilité de messages (MDN) pour AS2 envoyées à Partenaire numéro deux sont renvoyées à l'adresse appropriée, comme suit :

1. Cliquez sur l'icône **Edition**.
2. Entrez une adresse de courrier électronique AS MDN.  
Il s'agit de l'adresse utilisée pour recevoir les MDN pour AS1.
3. Entrez une URL HTTP AS MDN :  
`http://IP_COMPUTER_B:PORT/bcgreceiver/submit`

**Remarque :** Notez que l'URL définie pour AS2 utilise les paramètres qui seront définis pour le récepteur AS2 plus loin dans ce chapitre.

### **Création de destinations**

Dans WebSphere Partner Gateway, une destination définit un point du réseau qui permet l'entrée sur un autre réseau. La destination contient les informations indiquant à WebSphere Partner Gateway comment faire parvenir les documents à la couche EAI (Enterprise Application Integration).

**Création d'une destination pour Partenaire numéro un :** Partenaire numéro deux envoie un document EDI à Partenaire numéro un via le protocole AS2. La destination de Partenaire numéro un permet d'envoyer les documents EDI reçus via AS2 à une file d'attente JMS, puis à WebSphere Data Interchange pour conversion.

Afin de créer une destination pour Partenaire numéro un, procédez comme suit :

1. Cliquez sur **Administrateur du compte > Profils > Partenaire**.
2. Cliquez sur **Rechercher**.

3. Sélectionnez Partenaire numéro un en cliquant sur l'icône **Afficher les détails**, puis sélectionnez **Destinations**.
4. Cliquez sur **Créer** afin de créer une destination pour Partenaire numéro un.
5. Entrez les valeurs de cette nouvelle destination comme indiqué dans le tableau 98.

Tableau 98. Propriétés de la destination de Partenaire numéro un

Nom de zone	Valeur
Nom de la destination	JMStoPartnerOne
Transport	JMS
URI du récepteur	file:///opt/mqm/java/JNDI/WdiJms
Nom de la fabrique JMS	HUB60_QM_QCF
Classe de message JMS	TextMessage
Type de message JMS	TextMessage
Nom de file d'attente JMS	EDI_OUT_B
Nom JNDI de la fabrique JMS	com.sun.jndi.fscontext.RefFSContextFactory

6. Cliquez sur **Sauvegarder**.

Pour définir la destination JMStoPartnerOne en tant que destination par défaut pour Partenaire numéro un, procédez comme suit :

1. Cliquez sur **Afficher les destinations par défaut**.
2. Dans la liste **Production**, sélectionnez **JMS2toPartnerOne**.
3. Cliquez sur **Sauvegarder**.

**Création d'une destination pour Partenaire numéro deux :** Partenaire numéro un envoie un document EDI à WebSphere Partner Gateway Enterprise Edition, sur une file d'attente JMS. La destination de Partenaire numéro deux est utilisée pour envoyer les documents EDI reçus à Partenaire numéro deux via AS2.

Afin de créer une destination pour Partenaire numéro deux, procédez comme suit :

1. Cliquez sur **Administrateur du compte > Profils > Partenaire**.
2. Cliquez sur **Rechercher**.
3. Sélectionnez Partenaire numéro deux en cliquant sur l'icône **Afficher les détails**, puis sélectionnez **Destinations**.
4. Cliquez sur **Créer** afin de créer une destination pour Partenaire numéro deux.
5. Entrez les valeurs de cette destination comme indiqué dans le tableau 99.

Tableau 99. Propriétés de la destination de Partenaire numéro deux

Nom de la destination	AS2toPartnerTwo
Transport	HTTP/1.1
URI du récepteur	http://IP_COMPUTER_C/input/AS2
Nom d'utilisateur	partnerOne
Mot de passe	partnerOne

6. Cliquez sur **Sauvegarder**.

**Remarque :** Les propriétés Nom d'utilisateur et Mot de passe indiquées ci-dessus concernent la méthode de mappage du partenaire entrant pour HTTP, telle qu'elle est définie dans WebSphere Partner Gateway - Express.

Vous trouverez un exemple de définition de ces propriétés dans WebSphere Partner Gateway - Express, voir «Configuration de WebSphere Partner Gateway - Express», à la page 224.

Notez que AS2toPartnerTwo est en ligne et que son état est **Activé**.

Pour définir la destination AS2toPartnerTwo en tant que destination par défaut pour Partenaire numéro deux, procédez comme suit :

1. Cliquez sur **Afficher les destinations par défaut**.
2. Dans la liste **Production**, sélectionnez **AS2toPartnerTwo**.
3. Cliquez sur **Sauvegarder**.

### Définition d'interactions

Une définition de flux de document est un ensemble de "méta-informations" qui définit les fonctions de traitement des documents du partenaire. Pour permettre au système de traiter un document de gestion, au moins deux définitions de flux de document doivent être liées pour créer une interaction.

Pour créer une interaction entre Partenaire numéro un et Partenaire numéro deux, procédez comme suit :

1. Cliquez sur **Administrateur du concentrateur > Configuration du concentrateur > Définition de document**.
2. Cliquez sur **Gestion des interactions**, puis sur **Création d'une interaction**.
3. Dans la colonne Source, sélectionnez :
  - a. Package : **None**
  - b. Protocole : **EDI-X12 (ALL)**
  - c. Flux de document : **ISA (ALL)**
4. Dans la colonne Récepteur, sélectionnez :
  - a. Empaquetage : **AS**
  - b. Protocole : **EDI-X12 (ALL)**
  - c. Flux de document : **ISA (ALL)**
5. Affectez à Action la valeur **Passe-système**.
6. Cliquez sur **Sauvegarder**.
7. Cliquez de nouveau sur **Création d'une interaction**.
8. Dans la colonne Source, sélectionnez :
  - a. Empaquetage : **AS**
  - b. Protocole : **EDI-X12 (ALL)**
  - c. Flux de document : **ISA (ALL)**
9. Dans la colonne Récepteur, sélectionnez :
  - a. Package : **None**
  - b. Protocole : **EDI-X12 (ALL)**
  - c. Flux de document : **ISA (ALL)**
10. Affectez à Action la valeur **Passe-système**.
11. Cliquez sur **Sauvegarder**.

## Création de connexions de partenaire

Les connexions du partenaire correspondent au mécanisme qui permet au système de traiter et de router des documents entre le Partenaire interne et ses divers partenaires. Ces connexions contiennent les informations nécessaires à l'échange approprié de chaque flux de document.

Pour créer une connexion de partenaire entre Partenaire numéro un et Partenaire numéro deux, procédez comme suit :

1. Cliquez sur **Administrateur du compte** dans le menu principal puis sur **Connexions de partenaire** dans la barre de navigation horizontale.
2. Dans la liste **Source**, sélectionnez **Partenaire numéro un**.
3. Dans la liste **Cible**, sélectionnez **Partenaire numéro deux**.
4. Cliquez sur **Rechercher**.
5. Activez la connexion de partenaire affichée dessous en cliquant sur le bouton **Activer**. Les fonctions business-to-business indiquées dans le tableau 100 devraient s'afficher.

Tableau 100. Activation de la connexion de partenaire de Partenaire numéro un vers Partenaire numéro deux

Type de flux de document	Source	Cible
Empaquetage	None (N/D)	AS (N/D)
Protocole	EDI-X12 (ALL)	EDI-X12 (ALL)
Flux de document	ISA (ALL)	ISA (ALL)

Pour créer une connexion de partenaire où Partenaire numéro deux est la source et Partenaire numéro un la cible, procédez comme suit :

1. Cliquez sur **Administrateur du compte** dans le menu principal, puis sur **Connexions de partenaire** dans la barre de navigation horizontale.
2. Dans la liste **Source**, sélectionnez **Partenaire numéro deux**.
3. Dans la liste **Cible**, sélectionnez **Partenaire numéro un**.
4. Cliquez sur **Rechercher**.
5. Activez la connexion à l'aide des informations dans le tableau 101.

Tableau 101. Activation de la connexion de partenaire de Partenaire numéro deux vers Partenaire numéro un

Type de flux de document	Source	Cible
Empaquetage	AS (N/D)	None (N/D)
Protocole	EDI-X12 (ALL)	EDI-X12 (ALL)
Flux de document	ISA (ALL)	ISA (ALL)

## Création de récepteurs

La page Liste des récepteurs fournit des informations d'emplacement qui permettent au gestionnaire de documents de WebSphere Partner Gateway d'extraire les documents depuis l'emplacement système approprié en fonction du type de transport du document entrant. Vous pouvez créer des configurations de récepteur distinctes en fonction du type de transport. Le gestionnaire de documents peut ensuite interroger les emplacements du référentiel de documents de plusieurs serveurs de messagerie Web, FTP et POP, notamment les répertoires internes et les files d'attente JMS, pour les documents entrants.

Une fois le document extrait par le gestionnaire de documents depuis l'emplacement correspondant à un récepteur prédéfini, l'infrastructure de routage peut traiter le document en fonction de la configuration de canal.

Pour recevoir une transaction EDI depuis WebSphere Data Interchange, créez un récepteur JMS en procédant comme suit:

1. Cliquez sur **Administrateur du concentrateur > Configuration du concentrateur > Récepteurs**.
2. Cliquez sur **Créer un récepteur**.
3. Affectez les propriétés indiquées dans le tableau 102.

Tableau 102. Propriétés du récepteur pour la réception via JMS

Propriété du récepteur	Valeur
Nom du récepteur	WdiJmsListener
Transport	JMS
Type de destination	Production
URL du fournisseur JMS	file:///opt/mqm/java/JNDI/WdiJms
Nom de file d'attente JMS	EDI_IN_B
Nom de la fabrique JMS	HUB60_QM_QCF
Nom JNDI de la fabrique	com.sun.jndi.fscontext.RefFSContextFactory

Un second récepteur est requis pour la réception du document EDI provenant de Partenaire numéro deux via AS2. Pour la création de ce récepteur, procédez comme suit :

1. Cliquez sur **Administrateur du concentrateur > Configuration du concentrateur > Récepteurs**.
2. Cliquez sur **Créer un récepteur**.
3. Affectez les propriétés indiquées dans le tableau 103 ci-dessous :

Tableau 103. Propriétés du récepteur pour la réception via AS2

Nom du récepteur	HubAS2Listener
Transport	HTTP/S
Type de destination	Production
URI	/bcgreceiver/submit <b>Remarque :</b> L'URI de réception du protocole HTTP/S doit toujours commencer par /bcgreceiver

4. Cliquez sur **Sauvegarder**.

---

## Configuration de WebSphere Partner Gateway - Express

Cette section fournit les informations nécessaires à la configuration de l'environnement du partenaire externe. Dans ce cas, la gestion de cet environnement s'effectue via le système WebSphere Partner Gateway - Express. Dans l'exemple de scénario présenté dans ce chapitre, Partenaire numéro deux utilise WebSphere Partner Gateway - Express pour envoyer et recevoir des documents EDI à l'aide du protocole HTTP AS2.

Pour recevoir des messages EDI via HTTP AS2, procédez comme suit :

1. «Configuration de mon profil», à la page 225

2. «Création d'un partenaire pour Partenaire numéro un»
3. «Configuration du partenaire correspondant à Partenaire numéro un»

## Configuration de mon profil

En premier lieu, vous devez commencer par créer un profil pour Partenaire numéro deux dans WebSphere Partner Gateway - Express. Afin de créer un profil pour Partenaire numéro deux, procédez comme suit :

1. Cliquez sur **Configuration** dans le menu principal.
2. Cliquez sur **Mon profil** dans la barre de navigation horizontale.
3. Entrez les informations comme indiqué dans le tableau 104.

Tableau 104. Informations sur mon profil

Domaine non sécurisé de l'adresse de réception	IP_COMPUTER_C  où IP_COMPUTER_C représente l'adresse IP de l'ordinateur C, sur laquelle WebSphere Partner Gateway - Express est en cours de fonctionnement.
Port non sécurisé de l'adresse de réception	80  où 80 est le numéro de port désigné pour l'utilisation de WebSphere Partner Gateway - Express lors de l'installation.
ID expéditeur AS2	987654321
Type d'ID entreprise	DUNS
Identificateur entreprise	987654321

4. Cliquez sur **Sauvegarder**.

## Création d'un partenaire pour Partenaire numéro un

Partenaire numéro un doit être identifié en tant que partenaire de WebSphere Partner Gateway - Express. Pour créer Partenaire numéro un en tant que partenaire, procédez comme suit :

1. Cliquez sur **Configuration** dans le menu principal.
2. Cliquez sur **Partenaires** dans la barre de navigation horizontale.
3. Cliquez sur le bouton **Créer des partenaires**.
4. Affectez les valeurs suivantes :
  - a. Nom du partenaire : **partnerOne**
  - b. ID partenaire AS2 : **123456789**
5. Cliquez sur **Sauvegarder**.

Dans la vue relative à la gestion des partenaires s'affichent les informations détaillées de partnerOne.

## Configuration du partenaire correspondant à Partenaire numéro un

Après la création du partenaire correspondant à Partenaire numéro un, vous devez le configurer de sorte qu'il puisse fonctionner avec les protocoles AS2 et HTTP.

Cette configuration permet d'identifier les paramètres requis par WebSphere Partner Gateway - Express pour l'envoi et la réception via les protocoles HTTP et AS2 pour Partenaire numéro un.

Afin de configurer Partenaire numéro un pour les protocoles HTTP et AS2, procédez comme suit :

1. Cliquez sur **Configuration** dans le menu principal.
2. Cliquez sur **AS2** dans la barre de navigation horizontale.
3. Sélectionnez **partnerOne** dans la liste **Partenaire sélectionné** puis cliquez sur **Edition**.
4. Définissez l'adresse de destination sortante de partnerOne comme suit :  
`http://IP_COMPUTER_B:7080/bcgreceiver/submit`  
Où `IP_COMPUTER_B` correspond à l'adresse IP de l'ordinateur B.
5. Cliquez sur **Sauvegarder**.
6. Cliquez sur **HTTP** dans la barre de navigation horizontale. (**partnerOne** doit encore être affiché en tant que partenaire sélectionné.)
7. Cliquez sur **Edition**.
8. Définissez un nom d'utilisateur et un mot de passe pour les communications entrantes :  
Nom d'utilisateur : **partnerOne**  
Mot de passe : **partnerOne**  
N'oubliez pas que ces valeurs ont déjà été indiquées dans l'exemple de création de la destination par défaut pour Partenaire numéro deux (dans WebSphere Partner Gateway Enterprise Edition sur l'ordinateur B).
9. Définissez l'adresse de destination de la communication sortante :  
`http://IP_COMPUTER_B:7080/bcgreceiver/submit`
10. Cliquez sur **Sauvegarder**.

**Important :** Après avoir apporté ces modifications dans WebSphere Partner Gateway - Express, déconnectez-vous de la console et arrêtez la destination. Redémarrez la destination et la console pour que les modifications prennent effet.

---

## Récapitulatif

Ce chapitre décrit le processus d'interaction entre WebSphere Partner Gateway et WebSphere Data Interchange. Il fournit également les procédures de configuration de l'exemple de scénario décrit dans «Exemple de scénario utilisé dans ce chapitre», à la page 208.

Comme indiqué au début de ce chapitre, vous pouvez suivre le référentiel *Integrating WebSphere Data Interchange V3.2 with WebSphere Business Integration Connect V4.2* pour créer un exemple de configuration. Le référentiel fournit des exemples de scripts et de mappes pour vous aider à configurer l'environnement et vous indique comment tester un exemple d'échange. Pour accéder à ce référentiel, connectez-vous à l'adresse :

[www.ibm.com/developerworks/websphere/](http://www.ibm.com/developerworks/websphere/)

et recherchez le titre du référentiel.

N'oubliez pas que vous devrez sélectionner **ISA (ALL)** pour le flux de document - au lieu de **ALL (ALL)** - lors de la configuration des fonctions business-to-business des partenaires. Le flux de document correspondant aux interactions définies et aux connexions activées sera également défini sur **ISA (ALL)**.



---

## **Partie 5. Intégration à d'autres produits**



---

## Chapitre 13. Intégration à WebSphere Transformation Extender

Le présent chapitre décrit le processus d'intégration de WebSphere Partner Gateway à WebSphere Transformation Extender.

---

### Introduction

#### Qu'est-ce que WebSphere Transformation Extender ?

WebSphere Transformation Extender est une puissante solution d'intégration de données orientée transactions permettant d'automatiser la transformation de larges volumes de transactions complexes sans codage manuel. C'est la garantie pour les entreprises d'un rapide retour sur investissement. Ce logiciel prend en charge EDI, XML, SWIFT, HIPAA et d'autres formats d'intégration business-to-business normalisés, ainsi que l'intégration en temps réel de données provenant de multiples d'applications, bases de données, intergiciels (middleware) de messagerie et technologies de transmission de données répartis sur l'ensemble de l'entreprise.

WebSphere Transformation Extender réalise la transformation et du routage des données des systèmes source aux systèmes cible au sein d'environnements fonctionnant par lots et en temps réel. Ces sources peuvent comprendre des fichiers, des bases de données relationnelles, des middleware orientés messagerie (MOM), des applications mises en package (regroupées) ou d'autres sources externes. Une fois extraites les données de ces sources, le produit WebSphere Transformation Extender les transforme et les route vers un nombre indéfini de cibles qui en ont besoin, tout en offrant le contenu et le format approprié pour chaque système cible. Le logiciel WebSphere Transformation Extender assure les prestations suivantes :

- Connectivité avec une large gamme d'applications, de bases de données et de systèmes de messagerie de type grand systèmes, existants et entreprise, mais aussi avec des sources d'information externes
- Bibliothèque exhaustive de plus de 120 fonctions préconfigurées pour réduire les délais de développement et simplifier la spécification de règles de validation, de transformation et de routage
- Multiples options d'exécution pour prendre en charge la transformation au moment adéquat et avec le style correct -qu'elle s'effectue par lots, en temps réel ou de façon imbriquée
- Fonctionnalités de classe entreprise en termes de développement, de déploiement et de maintenance avec prise en charge des plateformes à haute disponibilité. Cela permettant de réduire les risques liés à l'administration et l'implémentation en continu et de délivrer des résultats plus précoces qu'avec le codage manuel.

## Comment WebSphere Transformation Extender s'interface-t-il avec WebSphere Partner Gateway ?

WebSphere Transformation Extender s'utilise au sein de WebSphere Partner Gateway pour effectuer la transformation des documents. L'intégration à WebSphere Transformation Extender peut être réalisée de deux façons différentes.

Avec la première méthode, il n'y a pas d'interaction directe entre WebSphere Partner Gateway et WebSphere Transformation Extender. Cette méthode implique de traiter WebSphere Transformation Extender comme une autre application dorsale. Il faut alors employer WebSphere Message Broker qui incorpore lui-même WebSphere Transformation Extender. Si l'on utilise cette méthode, on n'attendra pas que le document soit transformé depuis WebSphere Partner Gateway, mais plutôt qu'il transite par WebSphere Message Broker, qui utilisera à son tour WebSphere Transformation Extender pour accomplir la transformation.

Avec l'autre méthode, il y a interaction directe entre WebSphere Partner Gateway et WebSphere Transformation Extender. Un appel à WebSphere Transformation Extender est émis depuis WebSphere Partner Gateway à l'aide d'un exit utilisateur employé sous forme de gestionnaire d'étapes d'action. Un exit utilisateur est implémenté pour utiliser les API Java RMI de WebSphere Transformation Extender pour en appeler à WebSphere Transformation Extender afin de réaliser la transformation.

---

## Planification de l'intégration à WebSphere Transformation Extender

Comme indiqué dans l'introduction, l'intégration à WebSphere Transformation Extender peut être effectuée en suivant deux méthodes : l'une via WebSphere Message Broker et l'autre à l'aide d'un exit utilisateur.

### Intégration via WebSphere Message Broker

#### Exigences liées à l'utilisation de WebSphere Message Broker

Pour utiliser WebSphere Message Broker avec WebSphere Partner Gateway, vous devez posséder :

- la version de WebSphere Partner Gateway prise en charge par WebSphere Message Broker
- la version de WebSphere Transformation Extender prise en charge par WebSphere Message Broker
- WebSphere Transformation Extender for WebSphere Message Broker (courtier de messages de WebSphere)

#### Planification de l'utilisation de WebSphere Message Broker

Via WebSphere Message Broker, WebSphere Partner Gateway interagit indirectement avec WebSphere Transformation Extender. Ce qui signifie que les questions de topologie soulevées par WebSphere Partner Gateway seront liées à WebSphere Message Broker et que les questions de topologie suscitées par WebSphere Transformation Extender seront, elles aussi, liées à WebSphere Message Broker. La création, la configuration et le déploiement des mappes de WebSphere Transformation Extender appartiendront au contexte de WebSphere Message Broker.

## Intégration à l'aide d'exits utilisateur WebSphere Partner Gateway

### Exigences liées à l'utilisation d'un exit utilisateur de WebSphere Partner Gateway à partir d'une action

La version 6.1 de WebSphere Partner Gateway prend en charge l'intégration à WebSphere Transformation Extender 8.1 avec groupe de correctifs v.1 ou ultérieure. Si vous utilisez le groupe de correctifs v.1 et les appels locaux, vous aurez besoin du numéro de module de correction 105385 que vous pourrez vous procurer auprès du support de WebSphere Transformation Extender. Les groupes de correctifs de WebSphere Transformation Extender de version ultérieure peuvent comporter déjà ce module de correction. Les appels locaux, ou intra-processus, s'utilisent soit avec les API Java de WebSphere Transformation Extender, soit avec ses API RMI dans le cas où vous n'utilisez pas de serveur Java RMI (Remote Method Invocation).

### Planification de l'utilisation des exits utilisateur de WebSphere Partner Gateway

WebSphere Transformation Extender peut être appelé depuis WebSphere Partner Gateway aussi bien localement qu'à distance grâce aux API Java RMI. La méthode choisie est fonction de votre environnement de déploiement. Les appels distants offrent une plus grande souplesse et une sécurité accrue à l'égard de WebSphere Partner Gateway.

**Appels locaux (ou intra-processus) :** Les informations qui suivent s'appliquent aux appels locaux à WebSphere Transformation Extender :

- Un appel local peut être effectué à l'aide des API Java de WebSphere Transformation Extender ou à l'aide de ses API RMI en mode local (c'est à dire avec une valeur de nom d'hôte NULL et une valeur de numéro de port égale à 0).
- Vous devez exécuter WebSphere Transformation Extender sur la même machine que WebSphere Partner Gateway. Ce qui implique que chaque instance de WebSphere Partner Gateway requiert une installation de WebSphere Transformation Extender.
- La transformation du document se produit dans la même unité d'exécution et mémoire WebSphere Partner Gateway que l'API de WebSphere Transformation Extender depuis laquelle elle est appelée.
- WebSphere Partner Gateway et WebSphere Transformation Extender doivent être installés sur un système d'exploitation prenant en charge ces deux produits.
- La bibliothèque d'exécution de WebSphere Transformation Extender doit être placée dans l'environnement PATH de WebSphere Partner Gateway.
- Les mappes de WebSphere Transformation Extender doivent être disponibles pour les API de WebSphere Transformation Extender.
- Toute erreur imprévue au cours de la transformation est susceptible d'avoir une incidence sur WebSphere Partner Gateway.

**Appels à distance :** Les informations qui suivent s'appliquent aux appels distants à WebSphere Transformation Extender :

- WebSphere Partner Gateway peut être installé sur une machine distante par rapport à WebSphere Transformation Extender. Ce qui signifie que plusieurs instances de WebSphere Partner Gateway peuvent partager une même instance de WebSphere Transformation Extender.

- Il n'est pas nécessaire que WebSphere Partner Gateway soit exécuté sur un système d'exploitation Windows.
- Il est indispensable que le serveur RMI de WebSphere Transformation Extender soit en cours d'exécution.
- La transformation du document se produit sur le serveur RMI de WebSphere Transformation Extender.
- Le système de fichiers commun de WebSphere Partner Gateway doit être accessible au serveur RMI de WebSphere Transformation Extender pour le document d'entrée et le document de sortie.
- Les mappes de WebSphere Transformation Extender doivent être disponibles pour le serveur RMI de WebSphere Transformation Extender.
- Toute erreur imprévue au cours de la transformation n'aura pas d'incidence négative sur WebSphere Partner Gateway.

---

## **Intégration de WebSphere Message Broker à WebSphere Transformation Extender**

Ce type de déploiement induit que la transformation soit réalisée par WebSphere Transformation Extender à l'aide de WebSphere Message Broker, même s'il est toujours possible d'effectuer la transformation dans WebSphere Partner Gateway au moyen de fonctions externes. Voir Chapitre 11, «Intégration à WebSphere Message Broker», à la page 187 pour des détails sur l'implication de WebSphere Message Broker. Pour toute information sur l'intégration à WebSphere Message Broker, reportez-vous à la documentation de WebSphere Transformation Extender.

---

## **Intégration de l'exit utilisateur de WebSphere Transformation Extender**

Avant toute intégration à WebSphere Partner Gateway, vous devez installer WebSphere Transformation Extender en respectant les spécifications et instructions accompagnant ce produit. En outre, la machine sur laquelle s'exécute WebSphere Transformation Extender nécessite un accès au système de fichiers commun de WebSphere Partner Gateway pour pouvoir lire le document d'entrée à transformer et écrire le document de sortie issu de la transformation.

Pour les exigences de configuration, reportez-vous à la documentation produit de WebSphere Transformation Extender. Pour tout ce qui concerne le serveur RMI de WebSphere Transformation Extender, vous pouvez consulter «WebSphere Transformation Extender - configuration du serveur RMI», à la page 238.

## Création de l'exit utilisateur de WebSphere Partner Gateway pour WebSphere Transformation Extender

L'exit utilisateur WebSphere Transformation Extender doit utiliser les API RMI de WebSphere Transformation Extender car celles-ci offrent la plus grande souplesse quant au choix de l'emplacement de WebSphere Transformation Extender et autorisent l'utilisation d'une même installation de WebSphere Transformation Extender par plusieurs instances de WebSphere Partner Gateway. De la même manière, l'exit utilisateur devra vraisemblablement permettre la configuration du nom d'hôte et du port sur lequel écoute le serveur RMI de WebSphere Transformation Extender.

Pour des informations sur les tâches spécifiques requises pour la création de cet exit utilisateur, reportez-vous à la documentation suivante :

- Le *Guide du programmeur WebSphere Partner Gateway* pour obtenir des informations sur l'implémentation des gestionnaires de flux de travaux variables
- La documentation de WebSphere Transformation Extender pour obtenir des informations sur l'utilisation des API de WebSphere Transformation Extender
- L'exemple d'exit utilisateur de WebSphere Partner Gateway pour obtenir des informations sur l'exit utilisateur lui-même

## Configuration de WebSphere Partner Gateway pour l'exit utilisateur

A partir de la console WebSphere Partner Gateway, vous devez configurer WebSphere Partner Gateway pour l'utilisation de l'exit utilisateur de WebSphere Transformation Extender. Tout d'abord, il vous faut définir l'exit utilisateur vers WebSphere Partner Gateway et créer l'action qui utilisera cet exit utilisateur. Vous devrez accomplir en particulier les tâches suivantes :

1. Télécharger depuis votre système la définition de gestionnaire XML que vous avez créée pour votre exit utilisateur.
2. Créer une action qui utilisera votre exit utilisateur.

Une fois téléchargé l'exit utilisateur WebSphere Transformation Extender depuis votre système et créé l'action, configurez-le comme vous le feriez avec n'importe quel autre flux de document de WebSphere Partner Gateway. Vous devrez accomplir en particulier les tâches suivantes :

1. Configurer les types de document
2. Installer et configurer ses fonctions business-to-business
3. Importer vos mappes WebSphere Partner Gateway, dont se servira l'exit utilisateur.

**Remarque :** Les mappes auront le format que vous avez défini dans votre exit utilisateur.

4. Configurer les interactions.
5. Gérer les connexions.

Pour des informations plus précises sur ces tâches, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

## Configuration de la phase d'exécution de WebSphere Partner Gateway pour l'exit utilisateur

Dans «Configuration de WebSphere Partner Gateway pour l'exit utilisateur», à la page 235, vous avez configuré WebSphere Partner Gateway pour qu'il utilise l'exit utilisateur WebSphere Transformation Extender. Avant de pouvoir traiter des documents au travers de WebSphere Partner Gateway, vous devez rendre le code d'exécution accessible en installant les fichiers de classe Java.

Pour installer les fichiers de classe Java, procédez comme suit :

1. Placez les fichiers de classe Java de l'exit utilisateur dans le répertoire {WPG Install}\router\lib\userexits.
2. A partir du répertoire d'installation de WebSphere Transformation Extender, copiez le fichier JAR du SDK (kit de développement logiciel) dans le répertoire {WPG Install}\router\lib\userexits. Ce fichier JAR contient les classes d'exécution de WebSphere Transformation Extender requises pour appeler la transformation de mappes de WebSphere Transformation Extender.
3. WebSphere Partner Gateway nécessite un redémarrage pour sélectionner les fichiers JAR nouvellement créés.
4. Si vous exécutez le logiciel en local, ajoutez également le répertoire d'installation de WebSphere Transformation Extender dans la PATH du système. WebSphere Partner Gateway nécessite un redémarrage pour sélectionner les paramètres de path nouvellement créés.

## Exemple d'exit utilisateur fourni pour WebSphere Partner Gateway

Pour vous aider à intégrer WebSphere Partner Gateway pour WebSphere Transformation Extender, nous vous offrons un exemple d'exit utilisateur

### Présentation de l'exemple d'exit utilisateur WebSphere Transformation Extender

L'exemple d'exit utilisateur illustre les recommandations faites pour l'appel à WebSphere Transformation Extender depuis une action WebSphere Partner Gateway dans le but d'effectuer la transformation. L'implémentation en conditions réelles de l'exit utilisateur dépendra de la façon dont vous aurez construit vos mappes WebSphere Transformation Extender avec leurs cartes en entrée et leurs cartes en sortie. Cet exemple est fonctionnel et peut être exécuté depuis WebSphere Partner Gateway. Le fichier README de l'installation et de l'utilisation de l'exemple, le code source, les fichiers en entrée et les mappes sont situés dans le répertoire Integration/WebSphereTransformationExtender du CD-ROM du logiciel ;vous pourrez vous y référer pour des précisions complémentaires.

Cet exemple présente deux types de mappes. La mappe de WebSphere Transformation Extender créée à l'aide du concepteur de mappes de WebSphere Transformation Extender et la mappe de WebSphere Partner Gateway qui contient des métadonnées sous la forme d'un fichier de propriétés pour aider l'exit utilisateur à exécuter la mappe de WebSphere Transformation Extender. Le format de la mappe de WebSphere Partner Gateway est spécifique à l'implémentation de l'exit utilisateur. Pour plus de détails sur son contenu, reportez-vous à l'exemple de mappe de WebSphere Partner Gateway.

### Description de la classe de l'exemple d'exit utilisateur

Ces classes encapsulent les API de WebSphere Transformation Extender avec le code d'interface de l'exit utilisateur de WebSphere Partner Gateway nécessaire

pour fonctionner au sein d'une action. Reportez-vous au code source Java pour des détails sur l'implémentation. Voici des descriptions des classes de l'exemple d'exit utilisateur.

#### **WTXTransformationFactory**

Cette classe est le gestionnaire de l'exit utilisateur d'action. Elle crée et renvoie l'instance BusinessProcessInterface à utiliser en fonction des attributs du gestionnaire. Les attributs de gestionnaire eux-mêmes sont employés par l'instance du processus métier pour un complément de configuration

#### **WTXProcess**

C'est la super classe englobant les classes de processus métier, qui s'interface avec WebSphere Transformation Extender. Cette classe contient la logique de flux globale pour les aspects non API de WebSphere Transformation Extender. Les spécificités de l'appel d'API de WebSphere Transformation Extender sont encapsulées dans les sous-classes de la méthode callWTX(). Les sous-classes utilisent la méthode setCfg() pour définir toute information issue des attributs de gestionnaire, qui se révélerait pertinente pour cette sous-classe.

#### **WTXProcessRMI**

C'est une sous-classe de WTXProcess. Cette classe de processus métier appelle une mappe à l'aide des API RMI de WebSphere Transformation Extender sur WebSphere Transformation Extender ; à distance en faisant appel au serveur RMI de WebSphere Transformation Extender, ou localement pour le traitement d'une mappe, sans faire appel au serveur RMI de WebSphere Transformation Extender. Voici certaines de ces caractéristiques :

- Les entrées de la mappe de WebSphere Transformation Extender sont censées être des entrées de fichiers. Les informations de la mappe de WebSphere Partner Gateway servent à déterminer le nom de la mappe et d'autres spécifications de l'appel d'exécution.
- La valeur use RMI Server présente dans la configuration du gestionnaire d'exit utilisateur sert à déterminer si le serveur RMI de WebSphere Transformation Extender doit être employé. Si le serveur RMI de WebSphere Transformation Extender doit être employé, le nom d'hôte et le numéro de port présents dans la configuration du gestionnaire d'exit utilisateur seront utilisés pour y accéder. Les API RMI de WebSphere Transformation Extender peuvent servir localement à appeler une mappe ; par conséquent, si la configuration du gestionnaire d'exit utilisateur mentionne d'appeler la mappe en local, le même code peut être réemployé. Pour appeler une mappe localement, la DLL d'exécution de WebSphere Transformation Extender doit être présente dans le chemin d'accès.
- Notez que WebSphere Transformation Extender s'exécute sous Windows ; par conséquent, si vous utilisez WebSphere Partner Gateway sur une machine UNIX, c'est l'appel distant qui doit être utilisé.

#### **WTXProcessLocal**

C'est une sous-classe de WTXProcess. Cette classe de processus métier appelle une mappe localement sur WebSphere Transformation Extender. Voici certaines de ces caractéristiques : - Les entrées de la mappe de WebSphere Transformation Extender sont censées être des entrées de fichiers. Les informations de la mappe de WebSphere Partner Gateway servent à déterminer le nom de la mappe et d'autres spécifications de l'appel d'exécution.

Pour appeler une mappe localement, la DLL d'exécution de WebSphere Transformation Extender doit être présente dans le chemin d'accès.

**Remarque :** La classe WTXTransformationFactory n'instanciera pas cette sous-classe car les API RMI de WebSphere Transformation Extender de la classe WTXProcessRMI peuvent servir également aux appels locaux. Cette classe est fournie en exemple pour utiliser les API Java de WebSphere Transformation Extender.

### **Fonctionnement basique d'exécution de l'exemple d'exit utilisateur**

Les informations qui suivent décrivent le fonctionnement basique d'exécution de l'exemple d'exit utilisateur :

- WebSphere Partner Gateway reçoit un document XML du partenaire externe.
- La connexion WebSphere Partner Gateway est consultée sur la base du format XML et des ID entreprise obtenus auprès du document XML et le document est traité grâce à cette connexion.
- L'action personnalisée dans cette connexion est configurée pour utiliser l'exemple d'exit utilisateur Transformation de WebSphere Transformation Extender. Lorsque l'exit utilisateur Transformation de WebSphere Transformation Extender est appelé, il réalise les actions suivantes :
  - Obtenir le nom d'hôte et le numéro de port de la machine sur laquelle le serveur RMI de WebSphere Transformation Extender RMI Server s'exécute
  - Définir les informations de carte en entrée et en sortie de mappe, basées sur la mappe de WebSphere Partner Gateway Transformation Extender
  - Appeler la mappe de WebSphere Transformation Extender et placer le résultat dans l'objet Document de gestion

## **WebSphere Transformation Extender - configuration du serveur RMI**

Cette section expose ce qui doit être pris en considération lorsqu'il s'agit de configurer le serveur RMI de WebSphere Transformation Extender en lien avec WebSphere Partner Gateway. Pour tout détail sur les propriétés de configuration, reportez-vous à la documentation de WebSphere Transformation Extender.

Si la mappe de WebSphere Transformation Extender est appelée localement, la configuration du serveur RMI n'a pas lieu d'être. L'appel local (intra-processus) appelle la mappe au sein de la même unité d'exécution que celle qui s'exécute dans WebSphere Partner Gateway. Lors d'un appel distant à la mappe, c'est le serveur RMI qui est utilisé et les informations présentées dans cette section peuvent s'appliquer. Pour tout ce qui concerne les appels locaux (intra-processus), vous pouvez vous reporter à la propriété `server.mode.multi.process` décrite plus bas.

Le serveur RMI se démarre en ligne de commande. Le paramètre de ligne de commande `"-verbose"` (prolix) est recommandé puisque ce paramètre permettra d'imprimer vers la sortie standard certaines des caractéristiques de configuration de la phase d'exécution.

Au démarrage du serveur RMI, les paramètres de configuration de l'installation sont lus à partir d'un fichier de propriétés. Voici ce que vous devez prendre en compte vis-à-vis de WebSphere Partner Gateway lorsque vous configurez ces paramètres.

## **server.mode.multi.process**

Valeur recommandée : true

**Description** : cette propriété permet l'exécution du serveur RMI avec l'option de configuration multi-processus. Chaque appel au serveur RMI s'exécute dans un processus distinct et offre la meilleure protection si une exception vient à se produire sur la mappe. Si une exception se produit sur la mappe, le serveur RMI continuera de fonctionner. Si vous choisissez l'option de configuration mono-processus (à savoir `server.mode.multi.process=false`), chaque appel au serveur RMI s'exécutera dans le même processus, mais au sein d'une unité d'exécution différente. Toutefois, si une exception se produit sur la mappe, il y a un risque que le serveur RMI cesse de fonctionner.

### **Considérations sur la configuration multi-processus :**

- Chaque appel de mappe s'exécute dans son propre processus.
- Chaque processus a son propre segment de mémoire Java.
- La sécurité est accrue car une exception survenant dans une mappe n'entraînera pas l'arrêt du serveur RMI
- Chaque processus charge une instance de WebSphere Transformation Extender en mémoire et la mémoire peut supporter jusqu'à 30 Mo.
- Si un processus n'est pas disponible, la propriété `pool.acquire.process.timeout` du serveur RMI détermine la durée du temps d'attente avant expiration de l'appel de mappe.
- Pour garantir la disponibilité d'un nombre suffisant de processus, il vous faut configurer au moins autant de processus qu'il y a de gestionnaires de documents WebSphere Partner Gateway exécutés multipliés par le nombre d'unités d'exécution dans ces gestionnaires. Le gestionnaire de documents est constitué de plusieurs composants. Le composant qui exécutera l'exit utilisateur est le BPE (Business Process Engine), configuré par défaut sur deux unités d'exécution. En pratique, si deux instances de gestionnaire de documents sont installées suivant cette configuration par défaut, quatre unités d'exécution s'exécuteront (deux instances X deux unités d'exécution) et le serveur RMI devra être configuré pour quatre processus au minimum. Référez-vous à l'autre propriété de configuration du serveur RMI (`pool.max.process.count`) pour commander le nombre de processus.

### **Considérations sur la configuration mono-processus :**

- Chaque appel de mappe s'exécute dans son unité d'exécution particulière, mais dans le même processus que les autres unités d'exécution de la mappe.
- Une exception produite sur la mappe peut occasionner un arrêt du serveur RMI.
- Avec une seule instance de WebSphere Transformation Extender chargée, le temps système utilisé par la mémoire est moindre.

### **Considérations sur la configuration intra-processus :**

- L'exécution est locale et se fait sans l'intermédiaire du serveur RMI. Ceci n'est pas recommandé par WebSphere Transformation Extender dans un environnement serveur car la mappe s'exécute au sein de l'unité d'exécution effectuant l'appel, dans le cas présent : WebSphere Partner Gateway. Une exception anormale est susceptible de causer des incidents dans WebSphere Partner Gateway.
- WebSphere Transformation Extender doit être installé sur la même machine que WebSphere Partner Gateway.
- Restriction concernant uniquement les systèmes d'exploitation Windows.

## **pool.mode.managed**

Valeur recommandée : true

**Description** : cette propriété régit le nombre de processus pouvant être utilisés simultanément. Si vous définissez sa valeur sur false, ce nombre sera illimité. Recommandé pour prévenir les situations où les processus n'ont aucune limite à leur augmentation.

## **pool.max.process.count**

Valeur recommandée : (nombre total d'instances de WebSphere Partner) X (ce qui a été configuré pour leurs unités d'exécution)

**Description** : Le gestionnaire de documents est constitué de plusieurs composants. Le composant qui exécutera l'exit utilisateur est le BPE (Business Process Engine), configuré par défaut sur deux unités d'exécution. En pratique, si deux instances de gestionnaire de documents sont installées suivant cette configuration par défaut, quatre unités d'exécution s'exécuteront (deux instances X deux unités d'exécution) et le serveur RMI devra être configuré pour quatre processus au minimum. Cette propriété permet de garantir que les processus sont en nombre suffisant pour gérer les appels de l'exit utilisateur au serveur RMI. Si le nombre de processus est inférieur au nombre d'unités d'exécution de WebSphere Partner Gateway, les appels de l'exit utilisateur au serveur RMI devront attendre qu'un processus se libère, dans la mesure où les autres unités d'exécution de WebSphere Partner Gateway utilisent ces processus.

## **pool.max.keep.idle.count**

Valeur recommandée : aucune valeur fixée en particulier, mais une valeur suffisamment élevée néanmoins pour ne pas avoir à créer en permanence des processus, ce qui affecterait les performances. La valeur à inscrire est définie en secondes.

**Description** : cette propriété commande la durée de veille d'un processus d'être récupéré. En revanche, si le processus est requis à nouveau et que le plafond du pool.max.process.count n'est pas atteint, le processus sera recréé.

## **pool.acquire.process.timeout**

Valeur recommandée : 10 (secondes)

**Description** : cette propriété commande la durée d'attente d'un appel à une mappe pour l'obtention d'un processus que la mappe puisse exécuter. Si pool.max.process.count est correctement configuré, le délai d'attente ne devrait plus être un problème.

## **pool.map.auto.unload.timeout**

Valeur recommandée : 120 (secondes)

**Description** : le serveur RMI place les mappes en mémoire cache. Cette propriété commande la durée de conservation d'une mappe dans le pool de mémoire du serveur RMI avant sa suppression du pool. Si la mappe est supprimée du pool mais redevient nécessaire, elle sera lue à partir du système de fichiers. Si l'on suppose que les mappes de WebSphere Partner sont bien reconnues en production, il y aura un certain avantage à les conserver plus longtemps dans le cache. Le réglage optimal est celui que dicte l'expérience.

---

## Chapitre 14. Intégration à WebSphere Transformation Extender Trading Manager

Le présent chapitre décrit le processus d'intégration de WebSphere Partner Gateway à WebSphere Transformation Extender Trading Manager.

---

### Introduction

#### Qu'est-ce que WebSphere Transformation Extender Trading Manager ?

WebSphere Transformation Extender Trading Manager est un logiciel client/serveur de gestion et de traitement des données de commerce électronique qui offre des fonctions pour gérer et commander l'intégration business-to-business (d'entreprise à entreprise) des relations partenaires et des flux de messages. Les utilisateurs de Trading Manager procéder à des vérifications, des contrôles, des surveillances et visualiser la totalité de l'environnement d'intégration business-to-business à travers l'entreprise étendue grâce à des échanges de données sécurisés parfaitement intégrés avec les systèmes dorsaux.

Pour plus d'informations sur Trading Manager et ses composants, consultez la documentation de Trading Manager.

#### Comment WebSphere Transformation Extender Trading Manager s'interface-t-il avec WebSphere Partner Gateway ?

L'intégration à Trading Manager est analogue aux autres applications dorsales. L'usage typique est d'utiliser Trading Manager pour traiter les documents EDI. Exemple : un document AS2 comportant des données utiles d'EDI est reçu d'un partenaire externe et les données utiles EDI sont envoyées à Trading Manager pour traitement ultérieur. Les interfaces compatibles entre WebSphere Partner Gateway (destinations) et Trading Manager (bureaux de Poste) que vous pouvez utiliser sont Fichier, HTTP ou JMS.

Voici les différents cas de figure d'une intégration :

- Partenaire externe → WebSphere Partner Gateway → Trading Manager (partenaire interne) → Partenaire externe Application dorsale
- WebSphere Partner Gateway → Trading Manager (partenaire interne) → Application dorsale

#### Documentation applicable

La documentation applicable à WebSphere Partner Gateway est le *Guide de configuration du concentrateur WebSphere Partner Gateway*. Concernant Trading Manager, ce sera la documentation de Trading Manager.

---

## Planification de l'intégration à WebSphere Transformation Extender Trading Manager

La version 6.1 de WebSphere Partner Gateway prend en charge l'intégration à WebSphere Transformation Extender Trading Manager en version 7.8 ou ultérieure. Si c'est l'interface Fichier vers Trading Manager qui est utilisée, WebSphere Partner Gateway comme Trading Manager doivent avoir accès au même répertoire du système de fichiers que celui utilisé pour l'échange de documents.

---

## Configuration commune entre WebSphere Partner Gateway et WebSphere Transformation Extender Trading Manager

Une certaine proportion de la configuration est commune à WebSphere Partner Gateway et à Trading Manager. Le tableau qui suit répertorie les références se faisant écho entre les deux logiciels.

Tableau 105. Termes liés à la configuration de WebSphere Transformation Extender Trading Manager

WebSphere Partner Gateway	Trading Manager	Commentaires additionnels
Partenaire interne	Partenaire d'échange interne	
Partenaire externe	Partenaire d'échange externe	
ID entreprise du partenaire. S'agissant des EDI, il se compose du qualificatif d'échange EDI et de l'ID	ID entreprise du partenaire. S'agissant des EDI, le qualificatif d'échange et l'ID sont configurés séparément	
Destination	Bureau de poste GET (Facteur)	
Récepteur	Bureau de poste PUT (Boîte aux lettres)	
Connexion – du partenaire externe source au partenaire interne cible	Lien d'échange entrant	La relation n'est pas exactement de un pour un puisqu'il peut y avoir plus d'un lien d'échange entrant défini sur la base de la configuration du groupe EDI.
Connexion – du partenaire interne source au partenaire externe cible	Lien d'échange sortant	La relation n'est pas exactement de un pour un puisqu'il peut y avoir plus d'un lien d'échange sortant défini sur la base de la configuration du groupe EDI.

---

## Configuration de WebSphere Partner Gateway

Cette section décrit les points pertinents pour l'interface de WebSphere Transformation Extender Trading Manager.

Les flux de document sont définis comme normaux pour toute intégration. Les partenaires doivent être définis à la fois dans WebSphere Partner Gateway et dans Trading Manager même si vous pouvez limiter la configuration de WebSphere Partner Gateway à ce qui est obligatoire simplement pour l'opération de passe-système des documents.

Les problèmes spécifiques liés à l'interfaçage de Trading Manager au regard du partenaire interne, du partenaire externe, de la définition du document, des interactions, de la connexion et du récepteur sont exposés à la suite.

Pour toute information complémentaire concernant la définition des partenaires, des destinations, et des flux de document, reportez-vous au *Guide de configuration du concentrateur WebSphere Partner Gateway*.

## Problèmes liés au partenaire interne

Le partenaire interne est supposé servir à Trading Manager, pour lequel, il correspond au partenaire d'échange interne.

Tableau 106. Problèmes liés à la configuration du partenaire interne pour WebSphere Transformation Extender Trading Manager

Item	Considérations spécifiques
ID entreprise	L'ID entreprise représentant le partenaire interne dans l'échange EDI devra être configuré. Si le document est transmis à Trading Manager, ce sera le qualificatif et l'ID du récepteur de l'échange. Si le document est émis par Trading Manager, ce sera le qualificatif et l'ID de l'expéditeur de l'échange. Pour des détails, consultez Chapitre 15, «Routage de documents EDI», à la page 249.
Destination	Le type et la configuration du transport correspondront au bureau de poste GET (facteur) utilisé par Trading Manager, tel que Fichier, HTTP ou JMS. <ul style="list-style-type: none"> <li>• Si c'est l'interface Fichier qui est utilisée, WebSphere Partner Gateway et Trading Manager doivent avoir accès tous deux au répertoire qu'ils partagent.</li> <li>• Si c'est l'interface JMS qui est utilisée, WebSphere Partner Gateway comme Trading Manager doivent être configurés en vue d'utiliser les mêmes files d'attente JMS.</li> </ul>
Fonctions business-to-business	Les fonctions business-to-business du partenaire interne doivent être activées pour les types de document Source et Cible qu'utilise Trading Manager Concernant les documents, ce sont : <ul style="list-style-type: none"> <li>• Conditionnement – Aucun.</li> <li>• Protocole – EDI-EDIFACT ou EDI-X12.</li> <li>• Type de document – UNB (EDIFACT) ou ISA (X12).</li> </ul>

## Problèmes liés au partenaire externe

Le partenaire externe correspond ici au partenaire d'échange externe.

Tableau 107. Problèmes liés à la configuration du partenaire externe pour WebSphere Transformation Extender Trading Manager

Item	Considérations spécifiques
ID entreprise	L'ID entreprise représentant le partenaire externe dans l'échange EDI devra être configuré. Si le document est transmis à Trading Manager, ce sera le qualificatif et l'ID de l'expéditeur de l'échange. Si le document est émis par Trading Manager, ce sera le qualificatif et l'ID du récepteur de l'échange. Pour des détails, consultez Chapitre 15, «Routage de documents EDI», à la page 249.
Destination	La configuration de la destination du partenaire externe n'est pas nécessaire pour Trading Manager.

Tableau 107. Problèmes liés à la configuration du partenaire externe pour WebSphere Transformation Extender Trading Manager (suite)

Item	Considérations spécifiques
Fonctions business-to-business	<p>Les fonctions business-to-business du partenaire externe doivent être activées pour les types de document Source et Cible qu'utilise Trading Manager. Par exemple, pour des documents EDI utilisant le conditionnement de transport AS2, ce sera :</p> <ul style="list-style-type: none"> <li>• Empaquetage – AS</li> <li>• Protocole – EDI-EDIFACT ou EDI-X12.</li> <li>• Type de document – UNB (EDIFACT) ou ISA (X12).</li> </ul>

## Problèmes liés à la définition du document

Lors des envois ou réceptions de Trading Manager, utilisez le conditionnement de transport Aucun comme définition de document. Si le document est EDI, la définition de document à utiliser sera :

- Empaquetage – Aucun
- Protocole – EDI-EDIFACT ou EDI-X12
- Type de document – UNB (EDIFACT) ou ISA (X12).

## Problèmes liés aux interactions

Pour un envoi à Trading Manager, la définition du document cible sera celle définie à la section Définition du document. Pour une réception à partir de Trading Manager, la définition du document source sera celle définie à la section Définition du document. L'action à utiliser sera le passe-système.

## Problèmes liés à la connexion

Il n'existe aucun paramètre de configuration de la connexion spécifique à Trading Manager.

## Problèmes liés au récepteur

Pour le récepteur, le type et la configuration du transport correspondront au bureau de poste PUT (boîte aux lettres) utilisé par Trading Manager, tel que Fichier, HTTP ou JMS.

- Si c'est l'interface Fichier qui est utilisée, WebSphere Partner Gateway et Trading Manager doivent avoir accès tous deux au répertoire qu'ils partagent.
- Si c'est l'interface JMS qui est utilisée, WebSphere Partner Gateway comme Trading Manager doivent être configurés en vue d'utiliser les mêmes files d'attente JMS.

---

## Configuration de WebSphere Transformation Extender Trading Manager

Trading Manager se configure en mode normal. Bien que les partenaires soient définis dans WebSphere Partner Gateway, vous devrez aussi les définir dans Trading Manager. Le nom des partenaires n'a pas à être nécessairement identique entre WebSphere Partner Gateway et Trading Manager, mais conserver les mêmes noms vous aidera à administrer les deux logiciels.

Les points suivants sont spécifiques à l'interfaçage de WebSphere Partner Gateway :

Tableau 108. Considérations à propos de la configuration de WebSphere Transformation Extender Trading Manager

Item	Considérations spécifiques
Partenaire d'échange interne	<p>Le partenaire d'échange interne de Trading Manager correspond au partenaire interne dans WebSphere Partner Gateway.</p> <p>L'ID entreprise représentant le partenaire d'échange interne dans l'échange EDI devra être configuré. Si le document est transmis à WebSphere Partner Gateway, ce sera le qualificatif et l'ID de la source de l'échange. Si le document est émis par WebSphere Partner Gateway, ce sera le qualificatif et l'ID du récepteur de l'échange.</p>
Partenaire d'échange externe	<p>Le partenaire d'échange externe de Trading Manager correspond au partenaire externe dans WebSphere Partner Gateway.</p> <p>L'ID entreprise représentant le partenaire d'échange externe dans l'échange EDI devra être configuré. Si le document est transmis à WebSphere Partner Gateway, ce sera le qualificatif et l'ID du récepteur de l'échange. Si le document est émis par WebSphere Partner Gateway, ce sera le qualificatif et l'ID de l'expéditeur de l'échange.</p>
Bureau de poste GET (Facteur)	<p>Le type et la configuration du transport correspondront à la destination du partenaire interne dans WebSphere Partner Gateway utilisée pour l'interfaçage avec Trading Manager, tel que Fichier, HTTP ou JMS.</p> <ul style="list-style-type: none"> <li>• Si c'est l'interface Fichier qui est utilisée, WebSphere Partner Gateway et Trading Manager doivent avoir accès tous deux au répertoire qu'ils partagent.</li> <li>• Si c'est l'interface JMS qui est utilisée, WebSphere Partner Gateway comme Trading Manager doivent être configurés en vue d'utiliser les mêmes files d'attente JMS.</li> </ul>
Bureau de poste PUT (Boîte aux lettres)	<p>Le type et la configuration du transport correspondront au récepteur utilisé par WebSphere Partner Gateway, à savoir Fichier, HTTP ou JMS.</p> <ul style="list-style-type: none"> <li>• Si c'est l'interface Fichier qui est utilisée, WebSphere Partner Gateway et Trading Manager doivent avoir accès tous deux au répertoire qu'ils partagent.</li> <li>• Si c'est l'interface JMS qui est utilisée, WebSphere Partner Gateway comme Trading Manager doivent être configurés en vue d'utiliser les mêmes files d'attente JMS.</li> </ul>
Lien d'échange entrant	Aucune configuration particulière
Lien d'échange sortant	Aucune configuration particulière

Pour des informations complémentaires sur la configuration, reportez-vous à la documentation de WebSphere Transformation Extender Trading Manager.



---

## **Partie 6. Autres informations spécifiques aux protocoles de gestion**

Les chapitres suivants fournissent des informations spécifiques aux protocoles de gestion, comme le routage des documents EDI, ainsi que des instructions sur les traitements supplémentaires nécessaires pour les empaquetages AS, RosettaNet et ebMS.



---

## Chapitre 15. Routage de documents EDI

Cette section décrit le processus utilisé par WebSphere Partner Gateway pour déterminer les informations d’adressage des documents EDI qu’il envoie et qu’il reçoit. Elle décrit l’enchaînement général pour ce traitement (voir Présentation du routage EDI).

Vous pouvez ajouter des informations supplémentaires sur la manière d’utiliser l’intégration basée sur les fichiers lors du routage de documents EDI dans «Protocole de système de fichiers», à la page 44. Pour ce qui concerne spécifiquement l’empaquetage AS, voir Chapitre 16, «Considérations spécifiques à l’empaquetage AS», à la page 251.

---

### Présentation du routage EDI

Un document EDI contient des informations sur l’expéditeur et le récepteur du document. WebSphere Partner Gateway utilise ces informations lorsqu’il route le document EDI. Le flux général se présente comme suit :

1. WebSphere Partner Gateway détermine le protocole utilisé en étudiant les trois premiers caractères du document. Le tableau 109 indique le protocole du type de document associé à chaque code.

Tableau 109. Codes EDI, types de document et protocoles associés

Code	Type de document	Protocole du type de document	Communication sortante comme type de contenu
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 extrait du document EDI les informations relatives à l’expéditeur, en fonction de l’élément et de la position de ce type de document spécifique, comme indiqué dans le tableau 110.

Tableau 110. Codes EDI et emplacement des informations sur l’expéditeur et le récepteur

Code	Depuis le qualificatif	Depuis l’ID	Vers le qualificatif	Vers l’ID	Notes de support du composant EDI
ISA	Elément 105 à la position 5	Elément 107 à la position 6	Elément 105 à la position 7	Elément 106 à la position 8	Supporté
GS	N/D	Elément 142 à la position 2	N/D	Elément 124 à la position 3	adressage "GS-unique" non supporté

Tableau 110. Codes EDI et emplacement des informations sur l'expéditeur et le récepteur (suite)

Code	Depuis le qualificatif	Depuis l'ID	Vers le qualificatif	Vers l'ID	Notes de support du composant EDI
UNB UNA	Sous-élément 0007 à la position 2 de l'élément composite S002 à la position 20 (2ème composite) du segment UNB	Sous-élément 0004 à la position 2 de l'élément composite S002 à la position 20 (2ème composite) du segment UNB	Sous-élément 0007 à la position 2 de l'élément composite S003 à la position 30 (3ème composite) du segment UNB	Sous-élément 0010 à la position 1 de l'élément composite S003 à la position 30 (3ème composite) du segment UNB	Supporté
ICS	Elément X05 à la position 4	Elément X06 à la position 5	Elément X05 à la position 6	Elément X08 à la position 7	Précurseur d'ISA - Non supporté
STX	Elément FROM1 à la position 3	Elément FROM2 à la position 3	Elément UNT1 à la position 4	Elément UNT2 à la position 4	Non supporté dans cette édition
BG	N/D	Elément BG03 à la position 3	N/D	Elément BG04 à la position 4	Supporté

- WebSphere Partner Gateway détermine l'ID expéditeur à partir de l'ID expéditeur et du qualificatif du document EDI.  
Notez que certaines enveloppes EDI (par exemple, GS) n'ont pas la notion de qualificatif. Dans ce cas, WebSphere Partner Gateway utilise uniquement l'ID.
- WebSphere Partner Gateway enchaîne le qualificatif et l'ID au moyen d'un trait d'union (-) pour rechercher l'ID expéditeur dans le référentiel des profils WebSphere Partner Gateway. Par exemple, si, dans le message EDI pour l'expéditeur, le qualificatif est AB et l'identificateur est 1234567, WebSphere Partner Gateway s'attendra à trouver un partenaire externe doté de l'identificateur AB-1234567 dans le référentiel des profils. Si WebSphere Partner Gateway ne parvient pas à trouver cet ID, le document EDI n'est pas routé.
- Pour rechercher le partenaire de réception, WebSphere Partner Gateway détermine le qualificatif et l'ID récepteur à partir du message EDI.
- WebSphere Partner Gateway enchaîne le qualificatif et l'ID au moyen d'un trait d'union (-) pour rechercher l'ID récepteur dans le référentiel des profils.
- WebSphere Partner Gateway route le document vers le récepteur prévu.

---

## Chapitre 16. Considérations spécifiques à l’empaquetage AS

Cette section décrit le traitement supplémentaire requis lorsque l’empaquetage AS a été spécifié.

Lorsque l’empaquetage du document est en mode AS, WebSphere Partner Gateway procède à quelques opérations supplémentaires.

Pour plus d’informations sur les empaquetages Backend et AS, voir «Regroupement d’intégration dorsale», à la page 27.

---

### Routage des documents entrants

Lorsqu’un document provient d’un partenaire externe :

1. WebSphere Partner Gateway vérifie tout d’abord les informations de l’en-tête AS1 ou AS2. Il vérifie précisément les informations relatives à l’expéditeur et au récepteur pour déterminer si elles correspondent aux ID de partenaires externes valides.

- Pour AS1, il utilise la zone d’en-tête Objet sous la forme *ToID;FromID*.
- Pour AS2, il utilise les zones d’en-tête AS2-From et AS2-To.

Si les valeurs contenues dans les zones d’en-tête ne correspondent pas à des ID valides, WebSphere Partner Gateway ne route pas le document.

2. WebSphere Partner Gateway effectue ensuite les étapes habituelles pour les données utiles (il détermine le protocole et le type de document, extrait les ID entreprise, etc.). Pour des informations plus spécifiques sur les données utiles EDI, voir «Présentation du routage EDI», à la page 249.

---

### Routage des documents sortants

Lorsqu’un document provient d’une application dorsale, WebSphere Partner Gateway détermine si un attribut AS BusinessID a été spécifié pour l’empaquetage source (Aucun) et l’empaquetage récepteur (AS) :

- Si l’attribut AS BusinessId a été spécifié, WebSphere Partner Gateway utilise ces informations pour générer les ID De et A dans l’en-tête AS1 ou AS2.
- Si l’attribut n’a pas été spécifié, WebSphere Partner Gateway utilise les ID de l’expéditeur et du récepteur contenus dans le document. Dans le cas d’un échange EDI, les ID de l’expéditeur et du récepteur sont concaténés avec le qualificatif (voir «Présentation du routage EDI», à la page 249).

---

### Configuration des deux ID dans le profil de partenaire

Dans la mesure où WebSphere Partner Gateway utilise à la fois les informations de l’en-tête AS1 ou AS2 et les informations dérivées du document EDI, les ID d’un même partenaire peuvent avoir un format différent. Par exemple, les informations d’en-tête AS concernant l’expéditeur peuvent être 123456789 tandis que les informations dérivées du document EDI peuvent être AB-12345678.

Veillez à répertorier les deux ID du partenaire externe dans le profil. Pour plus d’informations, voir le *Guide de configuration du concentrateur WebSphere Partner Gateway*.

---

## Traitement des documents AS3 MDN

En général, lorsque WebSphere Partner Gateway renvoie un MDN au partenaire, il utilise pour ce faire la valeur de l'adresse IP provenant de l'attribut d'en-tête "Disposition-notification-to" du document AS3 demandé. Le comportement spécifique est déterminé par la destination de retour configurée sur la connexion utilisée pour le document AS3 demandé initial du partenaire.

Sur la connexion utilisée pour le document AS3 demandé initial du partenaire figure une configuration de destination de retour. La valeur de cette configuration fonctionne comme suit :

- **Destination FTP** : la valeur de l'adresse IP provenant de l'attribut d'en-tête "Disposition-notification-to" du document AS3 demandé initial est utilisée à la place de la valeur configurée pour cette destination FTP.
- **Destination de script FTP** : la valeur de l'adresse IP provenant de l'attribut d'en-tête "Disposition-notification-to" du document AS3 demandé initial ne sera pas utilisée. C'est la valeur configurée pour la destination de script FTP qui sera utilisée à la place.
- **Autre type de transport de destination** : le document MDN échoue.

---

## Autres références AS

Pour plus d'informations sur Backend Integration et AS, voir «Regroupement d'intégration dorsale», à la page 27.

---

## Chapitre 17. Considérations spécifiques à l’empaquetage RosettaNet

---

### Autres références RosettaNet

Pour toute information sur Backend Integration et RosettaNet, voir :

- «RosettaNet», à la page 16
- «Regroupement d’intégration dorsale», à la page 27

Pour toute information sur WebSphere Process Server et RosettaNet, voir «Planification de l’intégration à WebSphere Process Server», à la page 61.



---

## Chapitre 18. Considérations spécifiques à l’empaquetage ebMS

---

### Autres références ebMS

Pour plus d’informations sur Backend Integration et ebMS, voir les sections suivantes :

- «ebMS», à la page 17
- «Regroupement d’intégration dorsale», à la page 27



---

## Remarques

Le présent document porte sur des produits et services annoncés aux Etat-Unis.

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets. Toute demande de licence peut être adressée par écrit à :

*IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, N.Y. 10504-1785  
U.S.A.*

Les informations sur les licences concernant les produits utilisant un jeu de caractères double octet peuvent être obtenues par écrit à l'adresse suivante :

*IBM World Trade Asia Corporation Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106-0032, Japan.*

**Le paragraphe suivant ne s'applique ni au Royaume-Uni, ni dans aucun pays dans lequel il serait contraire aux lois locales.** LE PRESENT DOCUMENT EST LIVRE "EN L'ETAT" SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et logiciels décrits dans ce document.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des programmes créés de façon indépendante et d'autres programmes (dont celui-ci) et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Burlingame Laboratory Director  
IBM Burlingame Laboratory  
577 Airport Blvd., Suite 800  
Burlingame, CA 94010  
U.S.A

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans ce document et tous les éléments sous licence disponibles s'y rapportant sont fournis par IBM conformément aux dispositions de l'ICA, des Conditions internationales d'utilisation des logiciels IBM ou de tout autre accord équivalent.

Les données de performance indiquées dans ce document ont été déterminées dans un environnement contrôlé. Par conséquent, les résultats peuvent varier de manière significative selon l'environnement d'exploitation utilisé. Certaines mesures évaluées sur des systèmes en cours de développement ne sont pas garanties sur tous les systèmes disponibles. En outre, elles peuvent résulter d'extrapolations. Les résultats peuvent donc varier. Il incombe aux utilisateurs de ce document de vérifier si ces données sont applicables à leur environnement d'exploitation.

Les informations concernant des produits non IBM ont été obtenues auprès des fournisseurs de ces produits, par l'intermédiaire d'annonces publiques ou via d'autres sources disponibles. IBM n'a pas testé ces produits et ne peut confirmer l'exactitude de leurs performances ni leur compatibilité. Elle ne peut recevoir aucune réclamation concernant des produits non IBM. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Toute instruction relative aux intentions d'IBM pour ses opérations à venir est susceptible d'être modifiée ou annulée sans préavis, et doit être considérée uniquement comme un objectif.

Tous les tarifs indiqués sont les prix de vente actuels suggérés par IBM et sont susceptibles d'être modifiés sans préavis. Les tarifs appliqués peuvent varier selon les revendeurs.

Ces informations sont fournies uniquement à titre de planification. Elles sont susceptibles d'être modifiées avant la mise à disposition des produits décrits.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

## LICENCE SUR LES DROITS D'AUTEUR

Le présent logiciel contient des exemples de programmes d'application en langage source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation des plateformes pour lesquels ils ont été écrits ou aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes.

Toute copie totale ou partielle de ces programmes exemples et des oeuvres qui en sont dérivées doit comprendre une notice de copyright, libellée comme suit :

Copyright (c) 1995-2008 International Business Machines Corporation and others  
All rights reserved.

Si vous visualisez ces informations en ligne, il se peut que les photographies et illustrations en couleur n'apparaissent pas à l'écran.

---

## Informations relatives aux interfaces de programmation

Les informations relatives aux interfaces de programmation, lorsqu'elles sont disponibles, ont pour objet de vous aider à créer des applications à l'aide de ce programme. Les interfaces de programmation génériques vous permettent de créer des logiciels d'application qui obtiennent les services des outils de ce programme. Toutefois, ces informations peuvent également contenir des données de diagnostic ainsi que les modifications et les optimisations effectuées. Ces informations sont mises à votre disposition pour vous permettre de résoudre les incidents liés à vos applications.

**Avertissement :** N'utilisez pas les informations relatives aux diagnostics, aux modifications et à l'optimisation comme une interface de programmation dans la mesure où elles sont susceptibles d'être modifiées.

---

## Marques

Les termes qui suivent sont des marques d'International Business Machines Corporation aux Etats-Unis et/ou dans certains autres pays :

IBM	DB2	IMS	MQIntegrator	Tivoli
le logo IBM	DB2	Informix	MVS	WebSphere
AIX	Universal Database	iSeries	OS/400	z/OS
CICS	Domino	Lotus	Avantage du passeport	
CrossWorlds	IBMLink	Lotus Notes	SupportPac	
	i5/OS			

Microsoft, Windows, Windows NT et le logo Windows sont des marques de Microsoft Corporation aux Etats-Unis et/ou dans certains autres pays.

MMX, Pentium et ProShare sont des marques d'Intel Corporation aux Etats-Unis et/ou dans certains autres pays.

Solaris, Java et toutes les marques incluant Java sont des marques de Sun Microsystems, Inc. aux Etats-Unis et/ou dans certains autres pays.

Linux est une marque de Linus Torvalds aux Etats-Unis et/ou dans certains autres pays.

D'autres sociétés sont propriétaires des autres marques, noms de produits ou logos qui pourraient apparaître dans ce document.

WebSphere Partner Gateway Enterprise et Advanced Editions comprennent des logiciels développés par Eclipse Project ([www.eclipse.org](http://www.eclipse.org)).



---

# Index

## A

Adapter for HTTP  
  configuration 155  
  gestionnaire de données utiles 155  
  gestionnaire de protocole 156  
  installation 155  
  Liens de collaboration 165  
  structure d'objet métier 158  
Adapter for JMS  
  configuration 176  
  configuration de la file d'attente d'entrée 177  
  gestionnaire de données utiles 176  
  Liens de collaboration 183  
  structure d'objet métier 177  
Adapter for Web Services 166  
administrateur du concentrateur 4  
application dorsale  
  envoi de documents à 48  
  réception de documents de 52  
attribut content-length 28  
attribut du type de contenu 28, 35  
attribut Encoding 36  
Aucun emballage  
  description 26  
  protocole de transport HTTP 40  
  si nécessaire 36, 40, 42  
authentification pour les services Web 110, 111

## B

balises  
  données utiles 35  
  enveloppe de transport 35  
  pièce jointe 35  
brevets 257

## C

classe  
  BCGBackendIntegrationDataBindingUtil  
  exemple de code 84  
  méthode getAsByteArray 79  
  méthode getAsString 79  
  méthode getDataObject 80  
  Méthode getxAuxHeader 80  
  méthode read 80  
  méthode setBOPrefix 80  
  méthode setDataObject 80  
  méthode setDebugLevel 81  
  méthode setFromByteArray 81  
  méthode setFromString 81  
  méthode setOptions 82  
  méthode setPackagingSchema 82  
  méthode setTLOTypeNeName 83  
  méthode setxAuxHeader 83  
  méthode write 83  
  méthodes 78  
  présentation 78

classe  
  BCGBackendIntegrationJMSDataBindingImpl  
  méthode getMessageType 84  
  méthode isBusinessException 85  
  méthode read 85  
  méthode setBusinessException 85  
  méthode write 85  
  méthodes 84  
  présentation 84  
client Data Interchange Services 5, 11  
collaborations  
  Adapter for HTTP 165  
  Adapter for JMS 183  
  description 131  
contenu de service ebMS  
  description 17  
contenu de service RosettaNet  
  création d'objets métier 72  
  description 16  
contenu de service XMLevent  
  description 21

## D

définition du flux de document  
  définition 50  
  description 5  
définitions d'objet métier, InterChange Server  
  création 127  
  gestionnaire de données de pièces jointes 144  
  HTTP 157  
  JMS 177  
désignation "Vers le partenaire" sur le service Web 109  
destinations  
  définition 49  
  description 5, 49  
document ROD/à plat  
  emplacement des données utiles 34  
documents  
  AS1 31  
  AS2 30  
  cXML 10, 88, 91  
  ebMS 17, 32  
  EDI 249  
  RosettaNet 16, 29  
  SOAP 9, 107, 165, 197  
  XMLevent 21  
documents AS1 et en-têtes de niveau transport 31  
documents AS2 et en-têtes de niveau transport 30  
documents binaires  
  protocoles de transport pris en charge 38  
  restrictions d'emballage 40, 42  
documents cXML  
  description 10  
  emballage requis 40

documents cXML (*suite*)  
  protocoles de transport pris en charge 38  
Documents cXML  
  et WebSphere Process Server 88, 91  
  gestionnaire de données 129  
documents dorsaux 53  
documents ebMS  
  contenu de service 17  
  emballage pour 38  
  présentation 17  
  protocoles de transport pris en charge 38  
documents ebMS et en-têtes de niveau transport 32  
documents EDI  
  choix d'emballage 40  
  emplacement des données utiles 34  
  et WebSphere Data Interchange 206  
  protocoles de transport pris en charge 39  
  roulage 249  
documents ROD/à plat  
  protocoles de transport pris en charge 38  
documents RosettaNet  
  contenu de service 16  
  emballage pour 40  
  emplacement des données utiles 34  
  en tant que source pour les en-têtes de niveau transport 29  
  messages OA1 17  
  présentation 16  
  protocoles de transport pris en charge 38  
Documents RosettaNet  
  emballage pour 38  
  exemples d'intégration à WebSphere Process Server 63  
  exemples d'intégration d'InterChange Server 122  
documents SOAP  
  emballage requis 40  
  InterChange Server via HTTP 165  
  introduction 9  
  protocoles de transport pris en charge 38  
  WebSphere Message Broker via HTTP 197  
  WebSphere Process Server via HTTP 107  
documents XML  
  choix d'emballage 40  
  emplacement des données utiles 34  
  protocoles de transport pris en charge 38  
documents XMLevent  
  contenu de service 21  
  présentation 21  
données utiles  
  attributs de marque 35

- données utiles (*suite*)
  - balise 35
  - content type 35
  - description 34
  - encoding 36
  - message de notification d'événement 22
  - notification d'événement 22

## E

- échanges, EDI 11
- empaquetage
  - AS 251
  - Aucun 26
  - Backend Integration 26
  - description 5, 26
  - ebMS 255
  - RosettaNet 253
- empaquetage AS
  - considérations spécifiques 251
- empaquetage Backend Integration
  - liaison de données 66
- Empaquetage Backend Integration
  - description 26
  - exemple 36
  - indicateur de l'enveloppe 35
  - protocole de transport HTTP 40
  - Protocole de transport JMS 42
  - protocoles de transport 37
  - si nécessaire 36, 40, 42
- empaquetage ebMS
  - considérations spécifiques 255
- empaquetage RosettaNet
  - considérations spécifiques concernant 253
- en ligne/hors ligne, destination 46
- en-têtes de niveau transport
  - contenu 27
  - InterChange Server 162, 179
  - source AS1 31
  - source AS2 30
  - source ebMS 32
  - source RosettaNet 29
  - WebSphere Process Server 71
- enveloppes de transport
  - balise 35
  - et pièces jointes 35
  - schéma XML 35

## F

- files d'attente, envoi de messages 46
- flux de messages
  - création pour le transport HTTP 195
  - création pour le transport JMS 203
  - description 193
- Fonctions business-to-business
  - configuration des 51, 54
  - description 6

## G

- gestionnaire de données de pièces jointes
  - configuration 139
  - définitions d'objet métier 144

- gestionnaire de données de pièces jointes (*suite*)
  - emplacement 139
  - fichier référentiel 139
  - introduction 132
  - représentation des pièces jointes 147
- gestionnaire de données utiles
  - Adapter for HTTP 155
  - Adapter for JMS 176
- gestionnaire de files d'attente 43
- gestionnaire de protocole HTTP 156
- gestionnaires de données
  - méta-objet de niveau supérieur 143
  - méta-objet enfant 140
  - pièce jointe 132
- groupes, EDI 11

## H

- HTTPReply, noeud 196, 198
- HTTPRequest, noeud 196, 198

## I

- identificateur global unique (GUID) 47
- indicateur de l'enveloppe 35
- interactions
  - création 51, 55
  - description 6
  - pour l'envoi à l'application dorsale 51
- InterChange Server
  - composants requis pour l'envoi
    - via HTTP 151
    - via JMS 169
  - composants requis pour la réception
    - via HTTP 151
    - via JMS 169
  - configuration 127, 151, 164, 182
  - création d'artefacts pour 164, 182
  - exemples d'intégration 122
  - intégration à 169
  - introduction 119
  - planification de l'intégration 120
  - protocole de transport HTTP 151, 165
  - Protocole de transport JMS 169
  - protocoles de transport pris en charge 121
  - traitement des demandes 155
  - versions prises en charge 120
- intervalle de relance, destination 46

## L

- liaison d'exportation
  - description 60
  - JMS 67, 95
  - SCA 87
  - Service Web 108
- liaison d'exportation du service Web 108
- liaison d'importation
  - description 61
  - JMS 67
  - services Web 109

- Liaison d'importation
  - JMS 97
  - SCA 90
- liaison d'importation de services Web 109
- liaison de données
  - empaquetage Backend Integration 66
  - présentation 66
  - transport HTTP 67
  - transport JMS 67
- licence, brevets 257

## M

- messages de notification d'événement
  - exemple 25
  - zones 24
- messages OA1 17
- méta-objet de configuration de protocole HTTP 163
- méta-objet de niveau supérieur
  - MO\_DataHandler\_Default 143
- méta-objets dynamiques
  - HTTP 163
  - JMS 181
- méthode getAsByteArray 77, 79
- méthode getAsString 77, 79
- méthode getDataObject 80
- méthode getMethodType 84
- Méthode getxAuxHeader 80
- méthode isBusinessException 85
- méthode read 73, 80, 85
- méthode setBOPrefix 73, 80
- méthode setBusinessException 85
- méthode setDataObject 80
- méthode setDebugLevel 81
- méthode setFromByteArray 73, 81
- méthode setFromString 73, 81
- méthode setOptions 77, 82
- méthode setPackagingSchema 77, 82
- méthode setTLOTypeName 73, 83
- méthode setxAuxHeader 74, 83
- méthode write 77, 83, 85
- méthodes
  - getAsByteArray 77, 79
  - getAsString 77, 79
  - getDataObject 80
  - getMessageType 84
  - getxAuxHeader 80
  - isBusinessException 85
  - read 73, 80, 85
  - setBOPrefix 73, 80
  - setBusinessException 85
  - setDataObject 80
  - setDebugLevel 81
  - setFromByteArray 73, 81
  - setFromString 73, 81
  - setOptions 77, 82
  - setPackagingSchema 77, 82
  - setTLOTypeName 73, 83
  - setxAuxHeader 74, 83
  - write 77, 83, 85
- MQInput, noeud 196, 203, 204
- MQOutput, noeud 196, 203, 204

## N

noeud de traitement 196, 198, 203, 204  
noeud HTTPInput 196, 198  
nombre d'unités d'exécution,  
destination 46  
nombre de relances, destination 46  
notification d'événement 16, 20, 21, 22

## O

objet de niveau supérieur  
attributs 68  
description 68  
par défaut 74  
objet métier conteneur de données  
utiles 69  
objet métier conteneur de pièces jointes  
InterChange Server 148  
WebSphere Process Server 70  
objet métier en-têtes d'empaquetage 71  
objets métier  
InterChange Server  
données utiles 128, 146, 149  
méta-objet de configuration  
HTTP 163  
méta-objet dynamique 163, 181  
niveau supérieur 159  
pièce jointe 148  
propriétés définies par  
l'utilisateur 163  
propriétés JMS 180  
WebSphere Process Server  
conteneur de données utiles 69  
conteneur de pièces jointes 70  
en-têtes d'empaquetage 71  
niveau supérieur 68  
objets métier de données utiles  
WebSphere Process Server 69  
objets métier de niveau supérieur  
InterChange Server 159  
WebSphere Process Server 68  
objets métier de propriétés JMS 180  
objets métier données utiles  
InterChange Server 149, 158, 178  
objets métier propriétés définies par  
l'utilisateur 163  
octroi de licence  
adresse 257

## P

partenaire interne 4  
pièces jointes  
attributs de marque 35  
balise 35  
content type 35  
description 35  
encoding 36  
gestionnaire de données 132  
InterChange Server 146  
WebSphere Process Server 70  
présentation du traitement des  
documents 3  
présentation générale de l'intégration à  
un système dorsal 3  
propriété intellectuelle 257

protocole de transport fichier-répertoire  
structure de répertoires 44  
protocole de transport HTTP  
description 40  
emplacement des données utiles 34  
InterChange Server et 121, 151, 165  
servlet pour WebSphere Process  
Server 87, 92  
WebSphere Message Broker 188, 193  
WebSphere Process Server et 87  
protocole de transport JMS  
InterChange Server et 122  
Protocole de transport JMS  
création d'informations d'en-tête 179  
description 41  
emplacement des données utiles 34  
gestionnaire de files d'attente 43  
InterChange Server et 169  
structure d'objet métier InterChange  
Server 177  
WebSphere Message Broker 189, 198  
WebSphere Process Server et 95  
protocole de transport répertoire de  
fichiers  
description 44  
et WebSphere Process Server 113  
protocoles de gestion 9  
protocoles de transport  
Empaquetage Backend Integration 37  
liste des 37  
requis pour ebMS 17  
requis pour RosettaNet 16  
requis pour XMLEvent 21

## R

récepteurs  
définition 53  
description 4  
répertoire de documents 44

## S

schéma, XML  
données utiles de notification  
d'événement 22  
enveloppes de transport 35  
schéma XML  
données utiles de notification  
d'événement 22  
enveloppe de transport 35  
services Web  
authentification 110, 111  
et WebSphere Process Server 107  
fournis par le partenaire interne 9,  
108  
fournis par les partenaires 10, 109  
spécification "Vers le partenaire" 10,  
109  
URL du noeud final,  
modification 111  
servlet pour WebSphere Process Server,  
HTTP 87, 92

## T

traitement message en double 47  
transactions, EDI 11

## U

URL du noeud final, modification 111

## W

WebSphere Data Interchange 205  
WebSphere Message Broker  
composants requis pour l'envoi  
via HTTP 193  
via JMS 198  
composants requis pour la réception  
via HTTP 193  
via JMS 198  
configuration 192  
exemples d'intégration 189  
introduction 187  
planification de l'intégration 188  
protocole de transport HTTP 193  
Protocole de transport JMS 198  
protocoles de transport pris en  
charge 188  
versions prises en charge 188  
WebSphere Partner Gateway  
configuration 47  
envoi de documents à partir de 48  
installation avec WebSphere Process  
Server 62  
intégration  
à InterChange Server 119  
à WebSphere Data  
Interchange 205  
à WebSphere Message Broker 187  
à WebSphere Process Server 59  
à WebSphere Transformation  
Extender 231  
à WebSphere Transformation  
Extender Trading Manager 241  
réception de documents de 52  
WebSphere Partner Gateway - Express 4  
WebSphere Process Server  
exemples d'intégration 63  
liaison de données 66  
liaison de données de l'empaquetage  
Backend Integration 66  
objet de niveau supérieur 68  
objet métier conteneur de données  
utiles 69  
objet métier conteneur de pièces  
jointes 70  
objet métier en-têtes  
d'empaquetage 71  
planification de l'intégration 61  
présentation 59  
protocole de transport HTTP 87  
Protocole de transport JMS 95  
protocole de transport répertoire de  
fichiers 113  
Services Web 107  
versions prises en charge 61  
WebSphere Transformation Extender 231

WebSphere Transformation Extender  
Trading Manager 241

## X

x-aux-create-datetime 27  
x-aux-event-status-code 28  
x-aux-in-file-name 28  
x-aux-IntelligibleCheckRequired 29  
    source ebMS 34  
x-aux-msg-id  
    description 27  
    source AS1 32  
    source AS2 31  
    source ebMS 34  
    source RosettaNet 30  
x-aux-payload-root-tag  
    description 28  
    source AS1 32  
    source AS2 31  
    source RosettaNet 30  
x-aux-process-instance-id  
    description 28  
    source ebMS 33  
    source RosettaNet 30  
x-aux-process-type  
    description 27  
    source AS1 32  
    source AS2 31  
    source ebMS 33  
    source RosettaNet 30  
x-aux-process-version  
    description 27  
    source AS1 32  
    source AS2 31  
    source ebMS 33  
    source RosettaNet 30  
x-aux-production  
    description 27  
    source RosettaNet 30  
x-aux-protocol  
    description 27  
    source AS1 32  
    source AS2 31  
    source ebMS 33  
    source RosettaNet 30  
x-aux-protocol-version  
    description 27  
    source AS1 32  
    source AS2 31  
    source ebMS 33  
    source RosettaNet 30  
x-aux-receiver-id  
    description 27  
    source AS1 32  
    source AS2 31  
    source ebMS 33  
    source RosettaNet 30  
x-aux-sender-id  
    description 27  
    source AS1 32  
    source AS2 31  
    source ebMS 33  
    source RosettaNet 29  
x-aux-SyncResponse 28  
    source ebMS 33

x-aux-system-msg-id  
    description 27  
    source AS1 32  
    source AS2 31  
    source ebMS 33  
x-aux-third-party-bus-id 28  
    source ebMS 33  
x-aux-TimeToAccept 28  
    source ebMS 33  
x-aux-transport-retry-count 28  
x-out-filename 28

## Z

zone EventMessageID 24  
zone GlobalMessageID 24  
zone ID d'objet métier 24  
zone StatusCode 24  
zone StatusMessage 24  
zone Timestamp 24



**IBM**