

WebSphere Adapters



Adapter for JDBC - Guide de l'utilisateur

Version 6.0

Important

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section «Remarques», à la page 91.

Remarque

Les captures d'écran de ce manuel ne sont pas disponibles en français à la date d'impression.

Première édition - avril 2006

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 2006. Tous droits réservés.

© **Copyright International Business Machines Corporation 2005, 2006. All rights reserved.**

Table des matières

Avis aux lecteurs canadiens	v
WebSphere Adapter for JDBC Version 6.0 - Guide de l'utilisateur	1
Présentation du produit	1
Adaptateurs IBM WebSphere	1
Public visé	2
Organigramme des tâches : IBM WebSphere Adapter for JDBC	3
Reconnaissance de service d'entreprise	4
Architecture	5
Prise en charge des paramètres régionaux et de l'internationalisation	6
Présentation des objets métier	8
Structure des objets métier	8
Propriétés d'attribut d'objets métier	12
Opérations prises en charge	14
Gestion de transactions	15
Prise en charge des événements entrants	16
Traitement des objets métier des opérations de communication sortante	18
Informations spécifiques à l'application	25
Installation de l'adaptateur	33
Environnement de l'adaptateur	33
Informations d'installation spécifiques à l'adaptateur pour JDBC	33
Structure de fichiers installée	34
Création du projet d'adaptateur	35
Création d'un projet pour l'adaptateur	36
Ajout de bibliothèques fournisseur	36
Génération d'objets métier	37
Génération de liaisons de référence	52
Déploiement du projet d'adaptateur	53
Configuration de l'adaptateur sur le serveur	54
Identification et résolution des incidents	55
Pour contacter le centre de support logiciel IBM	56
Activation de la journalisation	57
Activation de la fonction de trace	59
Activation de Common Event Infrastructure (CEI)	60
Accès aux données de l'ordinateur principal	61
Utilisation des exemples de fichier et de l'exemple d'application	62
Exemples de fichier d'importation, d'exportation et WSDL	62
Exemple : Mise à jour d'une application de base de données	64
Référence	75
Propriétés de configuration	75
Propriétés de connexion	86
Propriétés de sélection des objets	88
Remarques	91
Informations sur les interfaces de programmation	94
Marques et marques de service	94

Avis aux lecteurs canadiens

Le présent document a été traduit en France. Voici les principales différences et particularités dont vous devez tenir compte.

Illustrations

Les illustrations sont fournies à titre d'exemple. Certaines peuvent contenir des données propres à la France.

Terminologie

La terminologie des titres IBM peut différer d'un pays à l'autre. Reportez-vous au tableau ci-dessous, au besoin.

IBM France	IBM Canada
ingénieur commercial	représentant
agence commerciale	succursale
ingénieur technico-commercial	informaticien
inspecteur	technicien du matériel

Claviers

Les lettres sont disposées différemment : le clavier français est de type AZERTY, et le clavier français-canadien de type QWERTY.

OS/2 et Windows - Paramètres canadiens

Au Canada, on utilise :

- les pages de codes 850 (multilingue) et 863 (français-canadien),
- le code pays 002,
- le code clavier CF.

Nomenclature

Les touches présentées dans le tableau d'équivalence suivant sont libellées différemment selon qu'il s'agit du clavier de la France, du clavier du Canada ou du clavier des États-Unis. Reportez-vous à ce tableau pour faire correspondre les touches françaises figurant dans le présent document aux touches de votre clavier.

France	Canada	Etats-Unis
 (Pos1)		Home
Fin	Fin	End
 (PgAr)		PgUp
 (PgAv)		PgDn
Inser	Inser	Ins
Suppr	Suppr	Del
Echap	Echap	Esc
Attn	Intrp	Break
Impr écran	ImpEc	PrtSc
Verr num	Num	Num Lock
Arrêt défil	Défil	Scroll Lock
 (Verr maj)	FixMaj	Caps Lock
AltGr	AltCar	Alt (à droite)

Brevets

Il est possible qu'IBM détienne des brevets ou qu'elle ait déposé des demandes de brevets portant sur certains sujets abordés dans ce document. Le fait qu'IBM vous fournisse le présent document ne signifie pas qu'elle vous accorde un permis d'utilisation de ces brevets. Vous pouvez envoyer, par écrit, vos demandes de renseignements relatives aux permis d'utilisation au directeur général des relations commerciales d'IBM, 3600 Steeles Avenue East, Markham, Ontario, L3R 9Z7.

Assistance téléphonique

Si vous avez besoin d'assistance ou si vous voulez commander du matériel, des logiciels et des publications IBM, contactez IBM direct au 1 800 465-1234.

WebSphere Adapter for JDBC Version 6.0 - Guide de l'utilisateur

IBM WebSphere Adapter pour JDBC fournit une connectivité bidirectionnelle entre les applications J2EE et les systèmes EIS (Enterprise Information System).

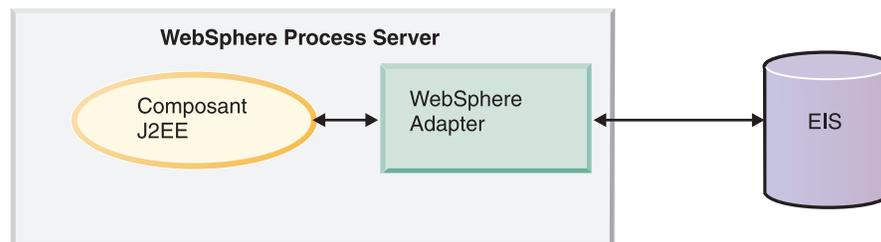
Présentation du produit

La présente rubrique aborde les concepts de base de WebSphere Adapter pour JDBC, son architecture et son environnement requis.

Adaptateurs IBM WebSphere

Les adaptateurs IBM WebSphere mettent en oeuvre l'architecture Java 2 Enterprise Edition (J2EE) Connector architecture (JCA), version 1.5. Egalement appelés adaptateurs de ressources ou adaptateurs JCA, les adaptateurs WebSphere autorisent une connectivité gérée et bidirectionnelle entre les systèmes EIS (enterprise information System) et les composants J2EE pris en charge par WebSphere Process Server.

Adaptateur WebSphere



La famille IBM WebSphere Adapter est une nouvelle génération d'adaptateurs, basée sur Java 2 Platform, Enterprise Edition (J2EE). JCA est une architecture standard qui permet d'intégrer des applications J2EE à des systèmes d'information d'entreprise. Chacun de ces systèmes fournit des interfaces API natives qui permettent d'identifier une fonction à appeler en spécifiant ses données d'entrée et en traitant ses données de sortie. L'architecture JCA vise à offrir une API indépendante pour la programmation de ces fonctions, faciliter le partage des données et intégrer les applications J2EE aux EIS existantes et nouvelles. La norme JCA atteint cet objectif en définissant une série de contrats régissant les interactions entre un système EIS et des composants J2EE au sein d'un serveur d'applications.

Totalement compatibles avec la norme JCA, les adaptateurs WebSphere ont été développés pour s'exécuter sur WebSphere Process Server. Un adaptateur WebSphere :

- S'intègre à WebSphere Process Server.
- Connecte avec un EIS une application exécutée sous WebSphere Process Server.
- Permet l'échange de données entre l'application et l'EIS.

Chaque adaptateur WebSphere est composé des éléments suivants :

- Une implémentation de l'architecture (J2EE) Connector Architecture (JCA), version 1.5 qui prend en charge WebSphere Process Server.

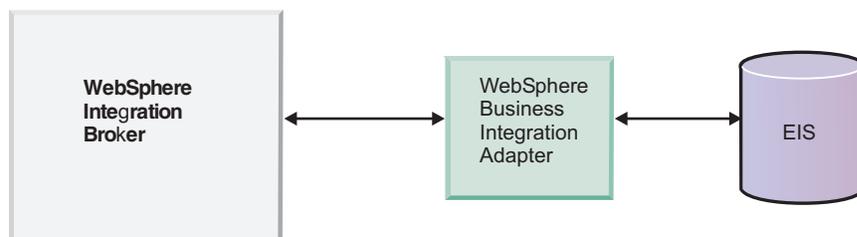
- Un composant de reconnaissance des méta-données d'entreprise (utilisé avec l'assistant de reconnaissance de service d'entreprise pour inspecter l'EIS), servant à générer des objets métier et autres objets d'architecture de composant de service (SCA), compilés dans un fichier standard d'archive d'application d'entreprise (EAR).

Les adaptateurs WebSphere utilisent des objets de données de service (SDO) pour représenter les objets de données.

WebSphere Adapters et WebSphere Business Integration Adapters

Contrairement aux adaptateurs WebSphere, les adaptateurs WebSphere Business Integration ne sont pas conformes JCA.

WebSphere Business Integration Adapter



Comme indiqué dans la figure, les adaptateurs WebSphere Business Integration sont répartis. Ils résident hors du serveur d'applications. Le serveur, ou courtier d'intégration, communique avec ce type d'adaptateur via une couche de transport JMS (Java Messaging Service).

Voici d'autres différences qui distinguent les adaptateurs WebSphere des adaptateurs WebSphere Business Integration :

- **Gestion de la connexion.** Les adaptateurs WebSphere utilisent des contrats JCA standard pour la gestion des tâches liées au cycle de vie, telles que l'arrêt et le démarrage ; en revanche, les adaptateurs WebSphere Business Integration s'appuient sur WebSphere Adapter Framework pour la gestion de la connectivité.
- **Notification d'événement.** Connue en tant que notification d'événements entrants.
- **Traitement de requête.** Connue en tant que prise en charge sortante dans WebSphere Adapters.
- **Définition d'objet.** Avec WebSphere Adapters, vous utilisez un composant de reconnaissance de métadonnées d'entreprise pour sonder un EIS et développer des objets métier et autres objets utiles. Ce composant de reconnaissance de métadonnées d'entreprise fait partie de WebSphere Adapter. Les adaptateurs WebSphere Business Integration utilisent un objet ODA (Object Discovery Agent) pour examiner un système EIS et générer des schémas de définitions d'objets métier.

Public visé

Cette rubrique définit les utilisateurs des produits WebSphere Adapter et détaille les compétences qu'ils doivent avoir.

Le guide de l'utilisateur de l'adaptateur est destiné aux intégrateurs de données et d'applications chargés d'assembler les composants d'application en une solution complète et de préparer cette solution en vue des tests et du déploiement. Ces utilisateurs doivent :

- Avoir une bonne compréhension de la solution et de l'environnement métier
- Connaître les composants de l'application et de la solution pour assurer leur collaboration efficace lors de l'exécution
- Avoir une connaissance approfondie des bases de données, des problèmes d'accès aux données, des modèles transactionnels et des connexions entre bases de données relationnelles, files d'attente et services Web hétérogènes
- Etre familiarisé avec les outils d'intégration

L'intégrateur d'application est également chargé des activités de test approfondi et doit avoir les compétences suivantes :

- La création des scripts, outils et modèles requis pour les tests et le déploiement
- La création d'espaces de travail d'intégration ainsi que de systèmes et de sous-systèmes intégrateurs
- La résolution des interdépendances entre des entités telles que les EJB, les flux de travaux et les pages Web
- La validation de l'application ou de la solution

L'intégrateur de données doit également permettre aux développeurs d'applications d'accéder à une variété de sources de données. Ce qui requiert les compétences suivantes :

- L'installation et la configuration de fonctions d'intégration ou de passerelles point à point
- L'écriture de procédures relatives à l'utilisation de la logique d'accès à la base de données
- La création de modèles de données pour les outils d'accès aux données externes
- La mise en oeuvre de mesures de sécurité

Concepts associés

«Exemple : Mise à jour d'une application de base de données», à la page 64
Des exemples de fichier sont fournis avec IBM WebSphere Adapter pour JDBC pour que vous puissiez vous entraîner à mettre à jour des tables spécifiques dans une application de base de données. Deux scénarios pas-à-pas sont fournis, l'un concernant l'intégrateur d'applications, l'autre l'intégrateur de données. Suivant votre rôle, vous pouvez vous entraîner à générer des objets métier, à déployer l'adaptateur et à configurer ce dernier pour communiquer entre les applications J2EE et les systèmes EIS (Enterprise Information System).

Organigramme des tâches : IBM WebSphere Adapter for JDBC

IBM WebSphere Adapter pour JDBC facilite les échanges d'objets métier au niveau de la base de données entre les applications J2EE et les systèmes EIS (Enterprise Information System) des fournisseurs de bases de données.

Tâche	Description
Installation de l'adaptateur	Informations décrivant les considérations à prendre en compte pour l'installation de l'adaptateur pour JDBC, avec un lien aux instructions d'installation.

Tâche	Description
Création du projet d'adaptateur	Créez un projet d'adaptateur, générez des objets métier et des blocs de construction de service, définissez les propriétés de configuration, puis déployez le projet d'adaptateur sur le serveur d'applications.
Déploiement du projet d'adaptateur	Exportez le projet dans un fichier EAR (Enterprise Application Archive) et installez ce fichier sur le serveur d'applications.
Configuration de l'adaptateur sur le serveur	Reconfigurez les propriétés utilisées par l'adaptateur pour configurer un canal de communication vers une application de base de données spécifique.
Résolution des incidents	Configurez un fichier journal pour suivre le statut du traitement des événements et déterminez le niveau des erreurs ou avertissements à capturer dans le fichier journal. Apprenez à utiliser le service de support logiciel IBM et à accéder aux notes techniques.
Mise à jour d'un exemple d'application de base de données	Utilisez les exemples de fichier fournis avec l'adaptateur pour JDBC pour vous entraîner à déployer, configurer et exécuter un exemple d'application de base de données. Deux scénarios sont fournis, avec des niveaux de complexité différents. Le second scénario inclut la génération d'objets métier.

Reconnaissance de service d'entreprise

L'assistant de reconnaissance de service d'entreprise permet de générer des objets métier pour les entités EIS ou de base de données.

L'assistant de reconnaissance de service d'entreprise fournit une méthodologie d'objet métier. Il vous permet de parcourir les métadonnées d'un EIS ou d'une base de données, de sélectionner des artefacts intéressants et de générer des objets de service déployables et des descriptions. En sélectionnant des noeuds de méta-objet dans l'arborescence des métadonnées, vous pouvez générer des objets métier pour les entités EIS ou de base de données. Les métadonnées sont transformées en objets de données de service, composés de graphiques métier et d'objets métier.

L'assistant de reconnaissance de service d'entreprise permet d'effectuer les opérations suivantes :

- Générer des objets métier
- Définir des informations propres à l'application sur les objets métier
- Définir des informations propres à l'application sur les propriétés
- Fournir des descriptions de service pour les événements entrants et sortants
- Fournir des descriptions de connexion pour les événements entrants et sortants

Architecture

WebSphere Adapter pour JDBC est un adaptateur de ressources qui fournit la connectivité bidirectionnelle entre les applications J2EE et les systèmes EIS (Enterprise Information System). Pour de telles applications, l'échange de données, qui prend la forme d'objets métier, est effectué au niveau de la base de données.

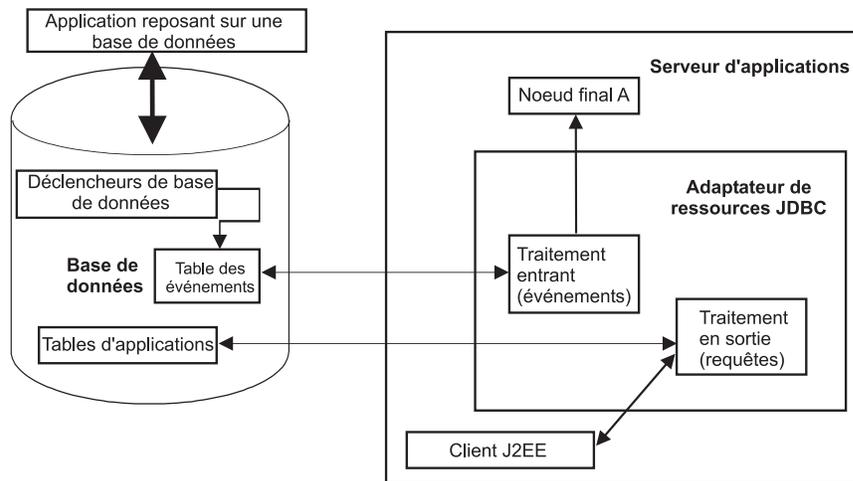
Adapter for JDBC communique avec un système EIS qui est un fournisseur de bases de données. Des mises à jour à une base de données peuvent devoir être appliquées à un autre système EIS et des modifications apportées aux données d'un système EIS peuvent devoir être appliquées à une base de données. L'adaptateur de ressources peut être intégré à toute application générée sur une base de données possédant un pilote JDBC qui prend en charge la spécification JDBC version 2.0 ou ultérieure. Il peut s'agir par exemple d'une base de données IBM DB2, Oracle, Microsoft SQLServer, Sybase ou Informix.

Pour prendre en charge l'intégration, l'adaptateur de ressources traite les demandes reçues d'un système EIS et les événements générés suite à des mises à jour de base de données. L'adaptateur transmet ces événements à divers noeuds finaux prédéfinis du serveur d'applications. Les *noeuds finaux* sont des applications J2EE ou d'autres consommateurs client de l'événement. Les mises à jour des données des tables d'une base de données peuvent être automatiquement propagées à d'autres applications connectées au serveur d'applications (par exemple, à des applications Siebel, PeopleSoft et Oracle), par l'intermédiaire des notifications d'événement placées dans un magasin d'événements. L'adaptateur met à jour les tables de base de données à l'aide de requêtes SQL ou de procédures stockées, comme indiqué dans les objets métier.

L'adaptateur pour JDBC prend en charge l'intégration des bases de données accessibles via l'API JDBC en prenant en charge les communications entrantes et sortantes sous la spécification JCA (Java 2 Enterprise Edition (J2EE) Connector Architecture). Sous les opérations *sortantes*, un objet métier est transmis à l'adaptateur sous forme de demande traitée conformément à l'opération spécifiée dans l'objet métier (create, retrieve, update, delete ou retrieveall). Les demandes proviennent de différentes applications EIS pour lesquelles les mises à jour doivent être appliquées à la base de données gérée par l'adaptateur. Le traitement de ces demandes entraîne la création, l'extraction, la mise à jour ou la suppression de lignes dans les tables de base de données correspondantes.

Sous les opérations *entrantes*, à mesure que les données sont modifiées dans les tables d'application de la base de données, les événements appropriés sont insérés dans un magasin d'événements, avec les informations adéquates, telles que les valeurs de clé. Pour capturer les données modifiées, des déclencheurs sont placés dans les tables respectives ou d'autres méthodes sont utilisées, telles qu'Oracle Change Data Capture, qui est fournie pour les bases de données Oracle. L'adaptateur pour JDBC interroge le magasin d'événements et extrait un lot d'événements. Ces événements sont traités et chaque événement est utilisé pour créer un graphique métier JDBC. Le graphique métier est ensuite envoyé aux noeuds finaux abonnés à l'objet métier spécifique. Seuls les objets image postérieure sont pris en charge pour les opérations entrantes.

La figure intitulée «Traitement dans l'adaptateur JDBC» illustre des opérations entrantes et sortantes.



Traitement dans l'adaptateur JDBC

Prise en charge des paramètres régionaux et de l'internationalisation

Ce connecteur a été internationalisé pour accepter les jeux de caractères à un et deux octets, et transmettre le texte du message dans la langue indiquée.

Cet adaptateur prend en charge le traitement de script bidirectionnel pour l'hébreu et l'arabe. Pour utiliser cette fonction, vous devez configurer les propriétés bidirectionnelles. Dans ce manuel de l'utilisateur, *propriétés bidirectionnelles* est le terme qui désigne les propriétés qui contrôlent l'appel de la prise en charge bidirectionnelle.

Si votre système d'informations d'entreprise (EIS) utilise un format bidirectionnel différent du format Windows standard, toutes les propriétés avec prise en charge bidirectionnelle sont transformées du format Windows standard vers le format bidirectionnel de l'EIS cible. L'adaptateur transforme également de telles données provenant de l'EIS au format Windows standard, avant de les transmettre à WebSphere Process Server.

L'environnement d'exécution Java dans la machine virtuelle Java (JVM) représente les données dans le jeu de codes de caractères Unicode. Le format Unicode contient des codes pour les caractères présents dans la plupart des jeux de codes de caractères connus (à la fois mono-octet et multi-octets). La plupart des composants du système WebSphere Business Integration sont écrits en Java. Par conséquent, lorsque des données sont transférées entre la plupart des composants du système WebSphere Business Integration, la conversion des caractères est inutile.

Pour consigner les messages d'erreur et d'informations dans la langue et le pays ou territoire approprié, l'adaptateur utilise les paramètres régionaux du système qui l'exécute.

Format de langue bidirectionnelle WebSphere Process Server

WebSphere Process Server utilise le format de langue bidirectionnelle ILYNN (par défaut, gauche à droite, activé, désactivé, nominal), qui est également le format de

langue bidirectionnelle de Windows. Tous les autres formats de langues bidirectionnelles doivent être convertis avant d'être présentés à WebSphere Process Server.

Cinq attributs doivent être définis pour le format de langue bidirectionnelle adéquat. Les attributs et paramètres sont indiqués dans le tableau «Attributs bidirectionnels».

Attributs bidirectionnels

Position de la lettre	Objet	Valeurs	Description	Paramètre par défaut
1	Schéma d'ordre	I ou V	Implicite (logique) ou Visuel	I
2	Direction	L R C D	De gauche à droite De droite à gauche De gauche à droite selon le contexte De droite à gauche selon le contexte	L
3	Permutation symétrique	Y ou N	La Permutation symétrique est activée ou désactivée	Y
4	Mise en forme	S N I M F B	Le texte est mis en forme Le texte n'est pas mis en forme Mise en forme initiale Mise en forme intermédiaire Mise en forme finale Mise en forme isolée	N
5	Mise en forme numérique	H, C, ou N	Hindi, contextuel ou nominal	N

L'adaptateur est chargé de transformer les données en format Gauche à droite logique, avant de les envoyer aux composants WebSphere Process Server.

Remarque : Les paramètres régionaux de l'interface utilisateur (navigateur) définissent le format d'édition et d'affichage de la langue bidirectionnelle. Les interfaces utilisateur WebSphere Process Server doivent convertir les formats propres aux paramètres régionaux dans le format par défaut de WebSphere Process Server.

Niveaux de propriété bidirectionnelle

Vous pouvez définir des propriétés bidirectionnelles à plusieurs niveaux. Pour plus d'informations sur ces propriétés et savoir comment les définir à l'aide de l'assistant de reconnaissance de service d'entreprise, voir les sections relatives à la création du projet d'adaptateur et à sa configuration.

Edition des propriétés bidirectionnelles

Vous pouvez éditer les propriétés bidirectionnelles des objets métier et attributs d'objet métier à l'aide d'annotations dans Business Object Editor de WebSphere Integration Developer. Les annotations sont conservées dans l'objet métier (le

fichier *.xsd). Pour d'informations, voir la documentation Business Object Editor sur le site Web de WebSphere Integration Developer à l'adresse <http://www.ibm.com/software/integration/wid>.

Vous pouvez également éditer certaines propriétés bidirectionnelles une fois qu'elles ont été définies, à l'aide de Assembly Editor dans WebSphere Integration Developer. Pour plus d'informations sur l'utilisation des propriétés bidirectionnelles lors de l'exécution, voir le document technique général et le document technique de l'adaptateur concernant la prise en charge bidirectionnelle. Pour plus d'informations sur Assembly Editor, voir la documentation s'y rapportant sur le site Web de WebSphere Integration Developer à l'adresse <http://www.ibm.com/software/integration/wid>.

Présentation des objets métier

Cet ensemble de rubriques décrit la structure, les propriétés d'attribut, les opérations prises en charge, les informations spécifiques à l'application et les conventions d'attribution de nom des objets métier. Elle contient des informations sur la gestion des transactions.

Structure des objets métier

Chaque objet métier correspond à une vue ou une table de base de données et chaque attribut simple de l'objet correspond à une colonne de cette table ou de cette vue.

Un *attribut simple* est un attribut qui représente une valeur unique, telle qu'une chaîne, un entier ou une date. De cette façon, les attributs à l'intérieur d'un même objet métier ne peuvent pas être stockés dans différentes tables de base de données ; cependant, les cas suivants sont possibles :

- La table de base de données peut comporter davantage de colonnes que l'objet métier correspondant ne dispose d'attributs simples ; en d'autres termes, certaines colonnes de la base de données ne sont pas représentées dans l'objet métier. Seules les colonnes requises pour le traitement de l'objet métier doivent être incluses dans votre conception.
- L'objet métier peut comporter davantage d'attributs simples que la table de base de données correspondante ne dispose de colonnes ; en d'autres termes, certains attributs de l'objet métier ne sont pas représentés dans la base de données. Les attributs n'ayant aucune représentation dans la base de données ne disposent d'aucune information spécifique à l'application, utilisent les valeurs par défaut ou indiquent des procédures stockées.
- L'objet métier peut représenter une vue qui concerne plusieurs tables de base de données. L'adaptateur peut utiliser cet objet métier lors du traitement des événements déclenchés dans l'application, tels que les événements de création, de mise à jour et de suppression. Toutefois, lors du traitement des demandes d'objet métier, l'adaptateur ne peut utiliser cet objet métier que pour les demandes d'extraction (Retrieve) et d'extraction globale (RetrieveAll).

Remarque : Si un objet métier est basé sur une procédure stockée, chaque attribut simple (autre que les attributs de procédures stockées spécifiques) peuvent avoir des informations spécifiques à l'application ou non. Pour plus d'informations, voir «Définition de procédure stockée».

Les objets métier peuvent être plats ou hiérarchiques. Tous les attributs d'un objet métier plat sont simples et représentent une ligne de la table de base de données. Le terme d'objet métier *hiérarchique* désigne un objet métier complet, incluant tous

les objets métier enfant qu'il contient à n'importe quel niveau. Le terme d'objet métier *individuel* désigne un objet métier, indépendant des objets métier qu'il peut contenir ou qui le contiennent. L'objet métier peut représenter une vue qui concerne plusieurs tables de base de données. Le terme d'objet métier *de niveau supérieur* désigne l'objet métier individuel en haut de la hiérarchie qui ne possède pas lui-même d'objet métier parent.

Un objet métier hiérarchique dispose d'attributs qui représentent un objet métier enfant, un tableau d'objets métier enfant ou une combinaison des deux. A son tour, chaque objet métier enfant peut contenir un objet métier enfant ou un tableau d'objets métier enfant et ainsi de suite. Le terme de *relation à cardinalité simple* est utilisé lorsqu'un attribut contenu dans un objet métier parent représente un objet métier enfant. Dans ce cas, l'attribut est de même type que l'objet métier enfant.

Le terme de *relation à cardinalité multiple* est utilisé lorsqu'un attribut contenu dans l'objet métier parent représente un tableau d'objets métier enfant. Dans ce cas, l'attribut est de même type que les objets métier enfant.

L'adaptateur prend en charge les relations suivantes entre les objets métier :

- Relations à cardinalité simple
- Relations à cardinalité simple et données sans droits de propriété
- Relations à cardinalité multiple

Dans chaque type de cardinalité, la relation entre les objets métier parent et enfant est décrite par les informations spécifiques à l'application de l'attribut clé de l'objet métier hébergeant la relation. Pour plus d'informations sur ces informations spécifiques à l'application, voir «Clé externe».

Référence associée

«Définition de procédure stockée», à la page 27

Les procédures stockées sont définies au niveau des instructions. Chaque définition de procédure stockée comprend les éléments suivants : StoredProcedureType, StoredProcedureName, ResultSet et Parameters.

«Propriétés d'attribut d'objets métier», à la page 12

L'architecture d'un objet métier définit les différentes propriétés s'appliquant aux attributs. Cette section décrit comment l'adaptateur interprète ces propriétés.

Relations à cardinalité simple dans les objets métier

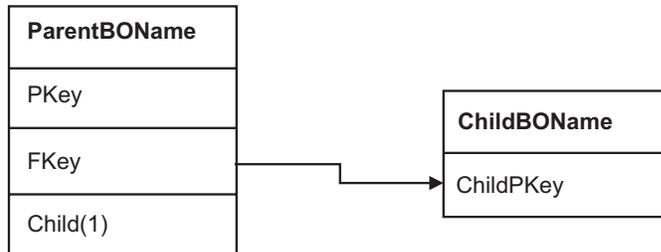
La présente rubrique décrit les relations à cardinalité simple prises en charge par cet adaptateur.

Relations à cardinalité simple

En règle générale, un objet métier contenant un objet métier enfant de type cardinalité simple possède au moins deux attributs représentant la relation. L'un des attributs est de même type que l'enfant. L'autre attribut est un attribut simple qui contient la clé principale de l'enfant considérée comme la clé externe dans le parent. Le parent dispose d'autant d'attributs de clé externe que l'enfant dispose d'attributs de clé principale.

Dans la mesure où les clés externes qui établissent la relation sont stockées dans le parent, chaque parent ne peut contenir qu'un seul objet métier enfant d'un type donné.

La figure intitulée «Relation type de cardinalité simple» illustre une relation type de cardinalité simple. Dans cet exemple, l'attribut FKey de la case ParentBOName représente l'attribut simple qui contient la clé principale de l'enfant et Child(1), qui se trouve également dans la case ParentBOName, désigne l'attribut qui représente l'objet métier enfant.



Relation type à cardinalité simple

Un objet métier parent peut comporter un enfant OWNERSHIP à cardinalité simple et un enfant NOOWNERSHIP à cardinalité simple.

Relations à cardinalité simple et données sans droits de propriété

En règle générale, chaque objet métier parent est propriétaire des données contenues dans l'objet métier enfant qu'il contient. Par exemple, si chaque objet métier Customer contient un objet métier Address, une nouvelle ligne est alors insérée dans les tables des clients et des adresses lors de la création d'un client. La nouvelle adresse est propre au nouveau client. De la même manière, lors de la suppression d'un client de la table des clients, l'adresse du client est également supprimée de la table des adresses.

Toutefois, il existe certains cas dans lesquels plusieurs objets métier hiérarchiques contiennent les mêmes données dont aucun d'eux n'est propriétaire. Par exemple, supposons qu'un objet métier Address possède un attribut *StateProvince*[1] représentant la *table de recherche StateProvince avec le type de cardinalité simple*. Dans la mesure où la table de recherche est rarement mise à jour et qu'elle est gérée indépendamment des données d'adresse, la création ou la modification des données d'adresse n'affecte pas les données contenues dans la table de recherche. L'adaptateur recherche un nom de région ou de département existant ou échoue. Il n'ajoute ni ne modifie aucune valeur dans la table de recherche.

Lorsque plusieurs objets métier contiennent le même objet métier enfant de type cardinalité simple, l'attribut de clé externe contenu dans chaque objet métier parent doit spécifier la relation NOOWNERSHIP. Lorsqu'un serveur d'applications envoie un objet métier hiérarchique à l'adaptateur avec une demande de création, de suppression ou de mise à jour, l'adaptateur ignore les enfants de type cardinalité simple contenus sans droits de propriété. L'adaptateur réalise uniquement l'extraction de ces objets métier. Si l'adaptateur ne parvient pas à extraire cet objet métier de type cardinalité simple, il renvoie un message d'erreur et met fin au traitement.

Données dénormalisées et données sans droits de propriété

Outre la simplification de l'utilisation des tables de recherche statiques, le confinement sans droits de propriété présente un autre intérêt : la synchronisation des données normalisées et dénormalisées.

Synchronisation des données normalisées avec les données dénormalisées :

Grâce à la relation NOOWNERSHIP vous pouvez créer ou modifier les données lorsque vous effectuez une synchronisation depuis une application normalisée vers une application dénormalisée. Par exemple, supposons qu'une application source normalisée stocke des données dans deux tables, A et B. Supposons également que l'application cible dénormalisée stocke toutes les données dans une table de telle manière que chaque entité de la table A stocke de manière redondante les données de la table B.

Dans cet exemple, pour synchroniser une modification apportée aux données de la table B depuis l'application source vers l'application cible, vous devez déclencher un événement de table A chaque fois que les données de la table B sont modifiées. De plus, dans la mesure où les données de la table B sont stockées de manière redondante dans la table A, vous devez envoyer un objet métier pour chaque ligne de la table A qui contient les données modifiées de la table B.

Remarque : Lorsque vous mettez à jour des tables dénormalisées, vérifiez que chaque enregistrement possède une clé unique afin qu'une même mise à jour ne modifie pas plusieurs lignes. Si cette clé n'existe pas, l'adaptateur pour JDBC génère une erreur indiquant que plusieurs enregistrements ont été mis à jour.

Synchronisation des données dénormalisées avec les données normalisées : Lors de la synchronisation de données depuis une application source dénormalisée vers une application cible normalisée, l'adaptateur ne crée, ne supprime ni ne met à jour aucune donnée contenue sans droits de propriété dans l'application normalisée.

Lors de la synchronisation de données vers une application normalisée, l'adaptateur ignore tous les enfants de type cardinalité simple contenus sans droits de propriété. Pour pouvoir créer, supprimer ou modifier ces données enfant, vous devez traiter les données manuellement.

Relations à cardinalité multiple dans les objets métier

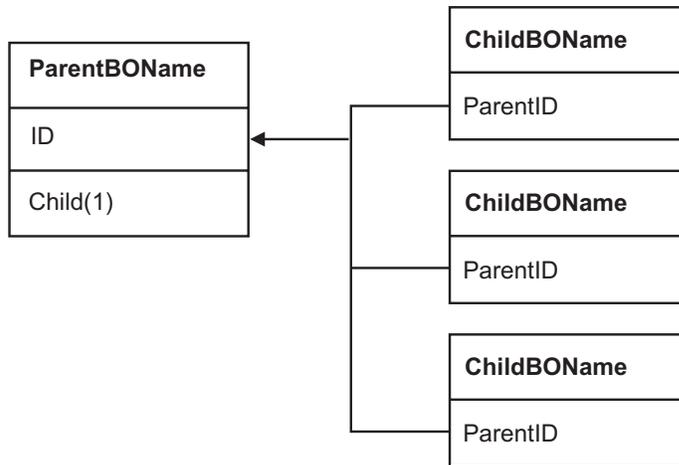
L'adaptateur prend en charge les relations à cardinalité multiple.

En règle générale, un objet métier contenant un tableau d'objets métier enfant possède un seul attribut représentant la relation. L'attribut est un tableau de même type que les objets métier enfant. Pour permettre à un parent de contenir plusieurs enfants, les clés externes qui établissent la relation sont stockées dans l'enfant.

Par conséquent, chaque enfant possède au moins un attribut simple contenant la clé principale du parent considérée comme une clé externe. L'enfant dispose d'autant d'attributs de clé externe que le parent dispose d'attributs de clé principale.

Dans la mesure où les clés externes qui établissent la relation sont stockées dans l'enfant, chaque parent peut n'avoir aucun enfant ou en avoir plusieurs.

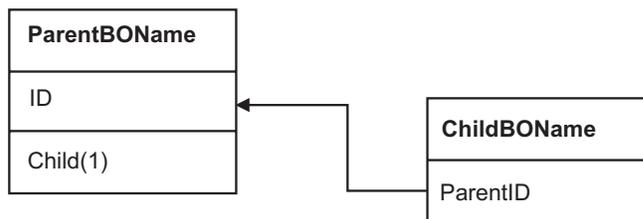
La figure intitulée «Relation d'objet métier à cardinalité multiple avec N=1» illustre une relation à cardinalité multiple. Dans cet exemple, dans les trois cases ChildBOName, ParentId représente l'attribut simple qui contient la clé principale du parent et Child1, dans la case ParentBOName, désigne l'attribut qui représente le tableau des objets métier enfant.



Relation d'objet métier à cardinalité multiple

Une relation à cardinalité multiple peut être une relation N=1. Certaines applications stockent une entité enfant de sorte que la relation parent-enfant est stockée dans l'enfant plutôt que dans le parent. En d'autres termes, l'enfant contient une clé externe dont la valeur est identique à celle stockée dans la clé principale du parent.

Les applications utilisent ce type de relation lorsque les données enfant n'existent pas indépendamment de leurs parents et qu'elles sont accessibles uniquement par l'intermédiaire des parents. Ces données enfant requièrent l'existence du parent et de sa valeur de clé principale avant que l'enfant et sa valeur de clé externe ne puissent être créés. La figure intitulée «Relation à cardinalité multiple avec N=1» illustre ce type de relation.



Relation à cardinalité multiple avec N=1

Propriétés d'attribut d'objets métier

L'architecture d'un objet métier définit les différentes propriétés s'appliquant aux attributs. Cette section décrit comment l'adaptateur interprète ces propriétés.

Le tableau suivant, intitulé "Propriétés des attributs", indique l'interprétation et les paramètres de ces propriétés.

Propriétés des attributs

Propriétés	Interprétation et paramètres
Cardinalité	<p>Chaque attribut d'objet métier représentant un enfant ou un tableau d'objets métier enfant possède la valeur cardinalité simple (1) ou multiple (n). Tous les attributs représentant des objets métier enfant ont également une propriété ContainedObjectVersion (qui indique le numéro de version de l'enfant) et une propriété Relationship (qui indique la valeur du paramètre Containment).</p>
Clé externe	<p>Lorsque des tableaux d'objets métier dont la cardinalité est n sont extraits, des clés externes sont utilisées dans la clause WHERE des instructions SELECT.</p> <p>Remarque : L'adaptateur ne prend pas en charge la spécification d'un attribut qui représente un objet métier enfant comme clé externe. L'instruction RetrieveAll se substitue à l'utilisation des clés et des clés externes.</p>
Clé	<p>Un attribut simple au minimum doit être spécifié dans chaque objet métier en tant que clé.</p> <p>Remarque : L'adaptateur ne prend pas en charge la spécification d'un attribut qui représente un objet métier enfant ou un tableau d'objets métier enfant en tant qu'attribut de clé. Si la propriété de clé a la valeur true pour un attribut simple, l'adaptateur ajoute cet attribut à la clause WHERE de l'instruction SELECT et des instructions UPDATE SQL qu'il génère lors du traitement de l'objet métier. L'instruction RetrieveAll se substitue à l'utilisation des clés et des clés externes.</p>
Nom	<p>Cette propriété représente le nom unique de l'attribut, s'il s'agit d'un attribut simple ou nom de l'objet métier, s'il s'agit d'un objet métier enfant.</p>

Propriétés	Interprétation et paramètres
Obligatoire	Spécifie si un attribut doit contenir une valeur. Si cette propriété a la valeur true pour un conteneur dont la cardinalité est simple (1), l'adaptateur oblige l'objet métier parent à contenir un objet métier enfant pour cet attribut. Les objets métier transmis à l'adaptateur pour les opérations de création, de mise à jour et de suppression doivent également contenir un objet métier enfant. La cardinalité est simple (1) pour les attributs simples et multiple (n) pour les attributs de conteneur. L'adaptateur fait échouer une opération de création si un objet métier ne dispose d'aucune valeur valide ou d'aucune valeur par défaut pour un attribut requis. L'opération échoue également si aucune donnée n'est disponible lors de l'extraction de cet objet de la base de données.
Type	Type de l'attribut (Integer, String, Date, Boolean, Double ou Float) s'il s'agit d'un attribut simple ou type d'objet métier s'il s'agit d'un objet métier enfant. Si l'adaptateur rencontre un attribut qu'il ne prend pas en charge, il insère la valeur entre guillemets, puis la traite en tant que données de type caractères.

Concepts associés

«Structure des objets métier», à la page 8

Chaque objet métier correspond à une vue ou une table de base de données et chaque attribut simple de l'objet correspond à une colonne de cette table ou de cette vue.

Opérations prises en charge

L'adaptateur effectue des opérations entrantes et sortantes. Les objets image postérieure et delta sont pris en charge pour les opérations sortantes. L'adaptateur ne prend en charge que les objets image postérieure pour les opérations entrantes. Les opérations prises en charge sont répertoriées ci-après.

Opérations sortantes

L'adaptateur prend en charge deux styles d'objet métier qui indiquent la quantité d'informations transmises par l'objet métier et leur objectif : image postérieure et delta. Une *image postérieure* représente l'état d'un objet métier après que toutes les modifications ont été apportées. Un *delta* est un objet métier utilisé dans une opération de mise à jour qui contient uniquement les valeurs de clés et les données à modifier. L'adaptateur prend en charge les objets image postérieure et delta pour les opérations ci-après :

- Create
- Update
- Delete
- ApplyChanges
- Retrieve

- RetrieveAll

Les instructions ne peuvent être définies que pour les objets métier des images postérieures. Une *instruction* reflète l'état de l'objet métier, tandis qu'une *opération* reflète l'opération à effectuer par l'adaptateur. Les instructions de niveau supérieur prises en charge pour l'image postérieure sont les suivantes :

- Create
- Update
- Delete
- UpdateWithDelete

Opérations entrantes

L'adaptateur ne prend en charge que les objets image postérieure pour les opérations entrantes. Lorsque l'adaptateur reçoit un objet métier pour mise à jour, il suppose que ce dernier représente l'état désiré des données après la mise à jour. Voici les opérations prises en charge :

- Create
- Update
- Delete

Gestion de transactions

L'adaptateur pour JDBC prend en charge les transactions locales et XA.

Dans l'adaptateur, une transaction correspond à une interaction isolée avec la base de données dorsale ou le système EIS (Enterprise Information System). Une transaction peut comprendre plusieurs opérations sur la base de données qui sont effectuées comme unité atomique. Ces opérations ne sont pas affectées par des opérations effectuées simultanément par d'autres applications client de la base de données.

L'adaptateur pour JDBC ne prend en charge les transactions que si la base de données dorsale les prend en charge. Les types de transactions admis sont les transactions locales et XA :

- Dans une *transaction locale*, une application client donnée définit le début et la fin de la transaction avec la base de données. Elle utilise un protocole de validation à une phase.
- Dans une *transaction XA*, la transaction s'étend à plusieurs bases de données hétérogènes. Elle utilise un protocole de validation global ou à deux phases.

L'adaptateur prend en charge les transactions XA pour les bases de données IBM DB2 et Oracle.

Utilisez les propriétés XADatasourceName et XADatabaseName avec les transactions XA. Pour plus de détails sur ces propriétés, voir les «propriétés des fabriques de connexions J2C» dans la section «Référence».

Remarque : Si vous utilisez une base de données DB2, vous devez configurer la propriété XADatabaseName.

La section «Référence» contient des exemples de valeur pour les propriétés utilisées avec les transactions locales et XA.

Référence associée

«Propriétés des fabriques de connexions J2C», à la page 79

Les propriétés des fabriques de connexions J2C permettent de configurer une instance de système EIS (Enterprise Information System) cible. Ces propriétés affectent le traitement des communications sortantes et correspondent à l'interface ManagedConnectionFactory de la spécification JCA (J2EE Connector Architecture).

«Référence», à la page 75

Cette section décrit les propriétés de configuration, de connexion et de sélection d'objets.

Prise en charge des événements entrants

L'adaptateur pour JDBC prend en charge la gestion des événements entrants avec la distribution asynchrone des événements.

La distribution asynchrone des événements s'effectue par l'intermédiaire d'une table d'événements, appelée magasin d'événements, et d'une table de transfert ou table de distribution des événements. A chaque modification des tables utilisateur, l'application alimente la table des événements dans le système EIS (Enterprise Information System). Les mises à jour sont effectuées en plaçant des déclencheurs dans les tables utilisateur qui enregistrent les événements de la table des événements correspondant aux mises à jour de la table utilisateur.

La table de transfert est déployée comme partie intégrante d'une base de données compatible XA, telle que Cloudscape, qui est incluse avec WebSphere Application Server, version 6.0. Lorsqu'un événement est extrait du système EIS, une référence à l'événement est enregistrée dans la table de transfert pour chaque noeud final actif. Chaque événement de la table de transfert est distribué à son noeud final correspondant dans le cadre d'une transaction XA unique contrôlée par le serveur d'applications. Lorsque le noeud final utilise son événement et que la transaction est validée, la référence à l'événement dans la table de transfert est supprimée pour ce noeud final et ce noeud final uniquement.

La table des événements est décrite dans le tableau ci-après.

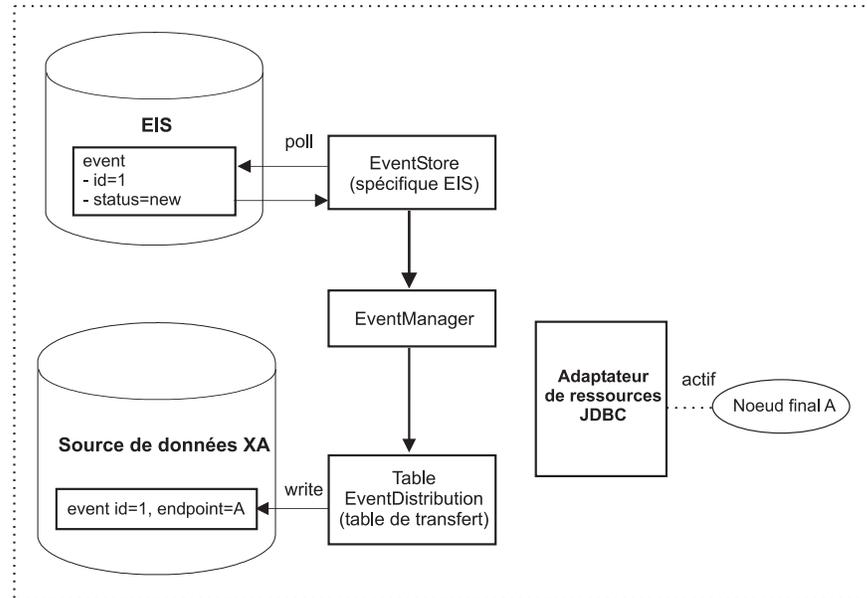
Table des événements

Nom	Type	Description
event_id	Number	ID événement unique correspondant à une clé principale de la table
object_key	String (Chaîne)	Chaîne délimitée contenant les clés de l'objet métier (non null)
object_name	String (Chaîne)	Nom de l'objet métier (non null)
object_function	String (Chaîne)	Opération correspondant à l'événement (Delete, Create, Update, etc.) (non null)
event_priority	Number	Non null
event_time	Date	Date et heure auxquelles l'événement a été généré
event_status	Number	En cours, Réussite, Echec, etc. (non null)
event_comment	String (Chaîne)	Description

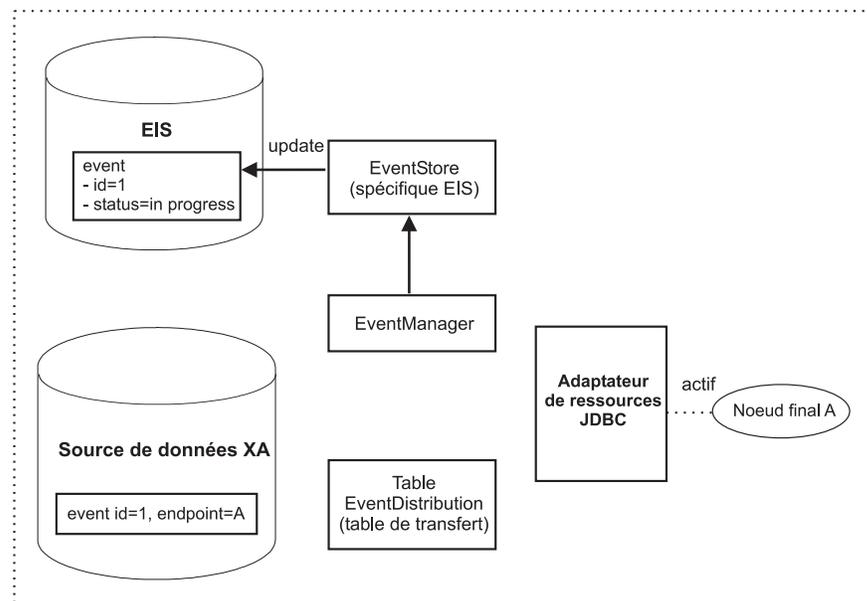
Traitement des événements dans l'adaptateur pour JDBC

Le flux complet du traitement des événements par l'adaptateur pour JDBC est décrit en quatre étapes illustrées dans les quatre figures ci-après.

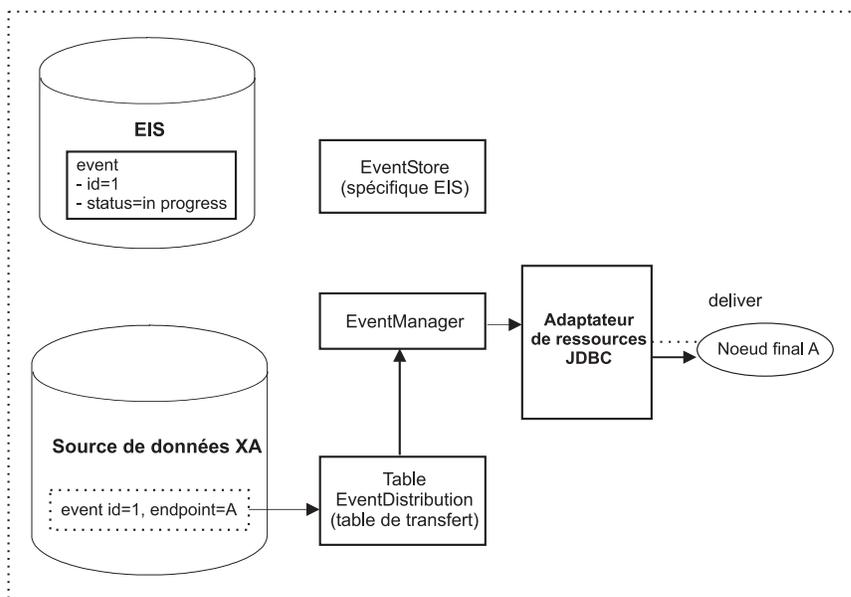
La première figure illustre l'étape 1, au cours de laquelle l'événement a été détecté dans le système EIS et ajouté à la table de transfert de chaque noeud final actif.



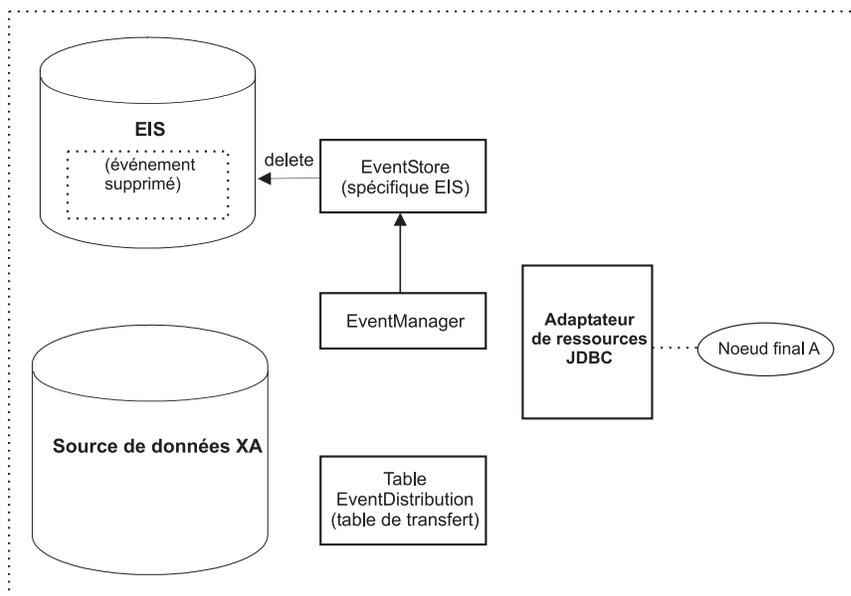
La figure suivante illustre l'étape 2, au cours de laquelle l'événement du système EIS est marqué comme étant en cours.



La figure qui suit illustre l'étape 3, au cours de laquelle l'événement est publié sur tous les noeuds finaux actifs, en utilisant la table de transfert comme guide pour lequel les noeuds finaux requièrent une distribution. Simultanément, l'événement est supprimé de la table de transfert dans le cadre de transactions XA englobant chaque noeud final et son enregistrement spécifique dans la table de transfert. En effet, chaque noeud final est isolé des autres de par la manière dont il reçoit et il traite l'événement.



Dans l'étape 4, une fois que tous les noeuds finaux ont reçus l'événement (il n'en reste plus de références dans la table de transfert), l'événement d'origine est supprimé du système EIS. Les étapes 1 à 4 doivent être répétées pour chaque événement.



Traitement des objets métier des opérations de communication sortante

Cet ensemble de rubriques présente les étapes réalisées par l'adaptateur lors de la création, l'extraction, la mise à jour ou la suppression d'un objet métier. L'adaptateur traite les objets métier hiérarchiques de manière récursive. En d'autres termes, il procède aux mêmes étapes pour chaque objet métier enfant jusqu'à ce qu'il ait traité tous les objets métier individuels.

Comparaison des objets métier

A différents points des opérations sur les instructions d'objet métier, l'adaptateur compare deux objets métier pour voir s'ils sont identiques. Par exemple, au cours d'une opération de mise à jour, l'adaptateur détermine si un objet métier spécifique existe dans un tableau d'objets métier. Pour effectuer cette vérification, l'adaptateur compare l'objet métier concerné avec chaque objet métier du tableau. Pour que deux objets métier soient considérés comme identiques, les deux conditions suivantes doivent être remplies :

- Les objets métier comparés doivent être de type identique. Par exemple, un objet métier Customer n'est jamais considéré comme identique à un objet métier Contact, même s'ils ont exactement les mêmes attributs.
- Tous les attributs de clé correspondants dans les deux objets métier doivent contenir des valeurs identiques. Si un attribut de clé n'est pas défini dans les deux objets métier, l'adaptateur les considère comme identiques. Toutefois, si un attribut de clé est défini dans l'un des objets métier et non dans l'autre, les objets métier ne sont pas identiques.

Opération Create (création)

En présence d'un objet métier hiérarchique, l'opération de création effectue un balayage récursif de l'objet métier et crée des lignes qui correspondent à chaque tableau.

L'adaptateur procède de la manière suivante :

1. L'opération de création insère de manière récursive chaque objet métier enfant de type cardinalité simple contenu avec des droits de propriété dans la base de données. En d'autres termes, l'adaptateur crée l'enfant et tous les objets métier enfant que l'enfant et ses enfants contiennent.
Si la définition d'objet métier spécifie qu'un attribut représente un objet métier enfant de type cardinalité simple et que cet attribut est vide, l'adaptateur ignore l'attribut. Toutefois, si la définition d'objet métier requiert que l'attribut représente un enfant et que ce n'est pas le cas, l'adaptateur renvoie un message d'erreur et met fin au traitement.
2. L'opération de création extrait chaque objet métier enfant de type cardinalité simple dépourvu de droits de propriété et en vérifie l'existence. Si l'extraction échoue, indiquant ainsi que l'enfant n'existe pas dans la base de données, l'adaptateur renvoie un message d'erreur et met fin au traitement. Si l'extraction réussit, l'adaptateur met à jour l'objet métier enfant de manière récursive.

Remarque : Pour permettre à cette approche de fonctionner correctement lorsque l'objet métier enfant existe dans la base de données de l'application, les références croisées des attributs de clé principale contenus dans les objets métier enfant doivent être correctes dans les opérations de création. Si l'objet métier enfant n'existe pas dans la base de données de l'application, les attributs de clé principale ne doivent pas être définis.

3. Il insère l'objet métier de niveau supérieur dans la base de données comme suit :
 - a. Il affecte à chaque valeur de clé externe les valeurs de clé principale de l'objet métier enfant correspondant représenté avec le type cardinalité simple. Dans la mesure où les valeurs des objets métier enfant peuvent être définies par des compteurs ou des séquences de base de données ou par la base de données elle-même lors de la création de l'enfant, cette étape

- garantit la validité des valeurs de clé externe dans le parent avant que ce dernier ne soit inséré dans la base de données par l'adaptateur.
- b. Il génère une nouvelle valeur d'ID unique pour chaque attribut défini automatiquement par la base de données. Le nom du compteur ou de la séquence de base de données est stocké dans les informations spécifiques à l'application de l'attribut. Si un attribut possède un compteur ou une séquence de base de données qui lui est associé, la valeur générée par l'adaptateur remplace toute valeur transmise par le serveur d'applications. Pour plus d'informations sur la spécification d'un compteur ou d'une séquence de base de données, voir UID=AUTO dans «Informations spécifiques à l'application des attributs simples.»
 - c. Il insère l'objet métier de niveau supérieur dans la base de données.
4. Il traite chaque objet métier enfant de type cardinalité multiple comme suit :
 - a. Il définit les valeurs de clé externe dans chaque enfant pour faire référence à la valeur contenue dans les attributs de clé principale correspondants dans le parent. Dans la mesure où les valeurs de clé principale du parent peuvent avoir été générées au cours de la création du parent, la validité des valeurs de clé étrangère dans chaque enfant est ainsi garantie avant que l'enfant ne soit inséré dans la base de données par l'adaptateur.
 - b. Il insère chaque objet métier enfant de type cardinalité multiple dans la base de données.

Opération Retrieve (extraction)

La présente rubrique décrit les étapes effectuées par l'adaptateur pour extraire un objet métier hiérarchique.

L'adaptateur effectue l'opération d'extraction de la manière suivante :

1. Il supprime tous les objets métier enfant de l'objet métier de niveau supérieur reçu. En d'autres termes, il effectue une copie de l'objet métier de niveau supérieur sans enfants.
2. Il extrait l'objet métier de niveau supérieur de la base de données.
 - Si l'extraction renvoie une ligne, l'adaptateur poursuit le traitement.
 - Si l'extraction ne renvoie aucune ligne, indiquant que l'objet métier de niveau supérieur n'existe pas dans la base de données, l'adaptateur renvoie l'erreur `RecordNotFoundException`.
 - Si l'extraction renvoie plusieurs lignes, l'adaptateur renvoie une erreur.
3. Il extrait de manière récursive tous les objets métier enfant de type cardinalité multiple.

Remarque : L'adaptateur n'impose pas le caractère unique des valeurs lors du remplissage d'un tableau d'objets métier. Il en va de la responsabilité de la base de données d'assurer cette unicité. Si la base de données renvoie des objets métier enfant en double, l'adaptateur renvoie à son tour des enfants en double.

4. Il extrait de manière récursive chaque enfant de type cardinalité simple, que l'objet métier enfant soit contenu avec ou sans droits de propriété.

Remarque : Tous les objets métier enfant de type cardinalité simple sont traités en fonction de leur occurrence dans l'objet métier et avant le traitement de l'objet métier parent. La propriété ou la non-propriété des objets enfant ne détermine pas la séquence de traitement mais en détermine le type.

Opération RetrieveAll (extraction globale)

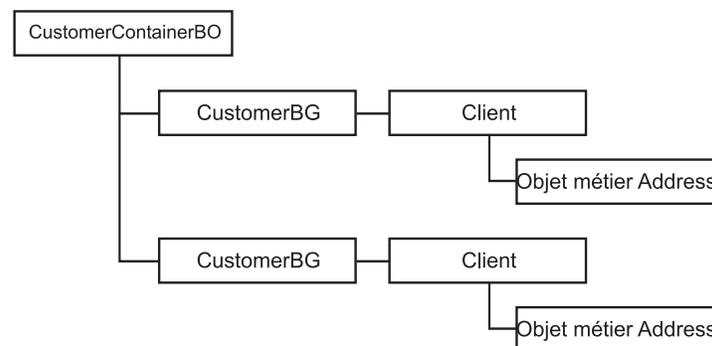
Cette opération permet à l'adaptateur d'extraire un tableau d'objets métier de la base de données.

Les critères de sélection sont déterminés par l'ensemble des attributs clé et non clé alimentés dans l'objet métier entrant. L'adaptateur peut extraire plusieurs lignes de l'objet métier de niveau supérieur de la base de données, suivant les attributs sélectionnés. Si aucun attribut n'est alimenté dans l'objet métier entrant, toutes les lignes sont extraites de leur table respective dans la base de données.

Pour extraire un tableau d'objets métier, l'adaptateur procède de la manière suivante :

1. Pour chacune des lignes extraites de la base de données, l'adaptateur génère un graphique métier de niveau supérieur et crée un conteneur de graphiques métier à l'aide de toutes les lignes extraites. Le nom du graphique métier du conteneur est *NomObjetMétier + GraphiqueMétierConteneur*.
2. L'adaptateur extrait chacun des graphiques métier du conteneur à l'aide de l'opération d'extraction.

La figure ci-après décrit la structure de l'objet métier renvoyé dans une opération d'extraction globale.



Les erreurs suivantes peuvent être générées lors d'une opération d'extraction globale :

- Si un objet métier alimenté dans l'objet en entrée n'existe pas dans le système EIS, l'adaptateur renvoie l'erreur `RecordNotFoundException`.
- Si le nombre d'occurrences dans le système EIS dépasse la valeur de `ResultSetLimit` définie dans la spécification d'interaction, l'adaptateur renvoie l'erreur `MatchesExceededLimitException`. La propriété `MatchCount` contient le nombre réel d'occurrences détectées par l'adaptateur dans le système EIS, pour que vous puissiez augmenter la limite ou affiner la recherche en conséquence.

Remarque : Si la valeur de la propriété `ResultSetLimit` est très importante, des incidents peuvent se produire en raison d'une mémoire insuffisante, suivant la taille des objets métier renvoyés et leur nombre.

- Si une erreur irréparable est signalée par le système EIS, l'adaptateur renvoie l'erreur `EISSystemException`.

Opération Update (mise à jour)

L'opération de mise à jour est effectuée en comparant l'objet métier entrant à un objet métier extrait de la base de données à l'aide des clés principales spécifiées dans l'objet métier entrant, de niveau supérieur.

Lors de la mise à jour d'un objet métier hiérarchique, l'adaptateur procède aux étapes suivantes :

1. Il utilise les valeurs de clé principale de l'objet métier source pour extraire l'entité correspondante de la base de données. L'objet métier extrait est une représentation exacte de l'état en cours des données dans la base de données.
Si l'extraction échoue, indiquant que l'objet métier de niveau supérieur n'existe pas dans la base de données, l'adaptateur renvoie l'erreur `RecordNotFoundException` et la mise à jour échoue.
Si l'extraction réussit, l'adaptateur compare l'objet métier extrait avec l'objet métier source pour déterminer les objets métier enfant qui requièrent des modifications dans la base de données. L'adaptateur ne doit pas, toutefois, comparer les valeurs contenues dans les attributs simples de l'objet métier source avec celles contenues dans l'objet métier extrait. L'adaptateur met à jour les valeurs de tous les attributs simples non clés.
Si tous les attributs simples contenus dans l'objet métier de niveau supérieur représentent des clés, l'adaptateur ne peut générer de requête de mise à jour pour l'objet métier de niveau supérieur. Dans ce cas, l'adaptateur consigne un message d'avertissement et procède à l'étape 2.
2. Il met à jour de manière récursive tous les enfants de type cardinalité simple de l'objet métier de niveau supérieur.
Si la définition d'objet métier nécessite qu'un attribut représente un objet métier enfant, l'enfant doit exister à la fois dans l'objet métier source et dans l'objet métier extrait. Dans le cas contraire, la mise à jour échoue et l'adaptateur renvoie un message d'erreur.
L'adaptateur gère les enfants de type cardinalité simple contenus avec des droits de propriété de l'une des manières suivantes :
 - Si l'enfant est présent dans les objets métier source et extraits, l'adaptateur supprime l'enfant existant dans la base de données et en crée un nouveau au lieu de le mettre à jour.
 - Si l'enfant est présent dans l'objet métier source, mais non dans l'objet métier extrait, l'adaptateur le crée de manière récursive dans la base de données.
 - Si l'enfant est présent dans l'objet métier extrait, mais non dans l'objet métier source, l'adaptateur le supprime de manière récursive de la base de données.

Pour les enfants de type cardinalité simple contenus sans droits de propriété, l'adaptateur tente d'extraire chaque enfant de la base de données présent dans l'objet métier source. S'il y parvient, l'adaptateur alimente l'objet métier enfant sans le mettre à jour dans la mesure où les enfants de type cardinalité simple contenus sans droits de propriété ne peuvent être modifiés par l'adaptateur.
3. Il met à jour tous les attributs simples de l'objet métier extrait, à l'exception de ceux dont l'attribut correspondant dans l'objet métier source n'est pas spécifié.
Dans la mesure où l'objet métier mis à jour doit être unique, l'adaptateur vérifie qu'une seule ligne est traitée en retour. Il renvoie un message d'erreur si plusieurs lignes sont traitées.
4. Il traite chaque enfant de type cardinalité multiple de l'objet métier extrait de l'une des manières suivantes :
 - Si l'enfant existe dans les tableaux des objets métier source et extraits, l'adaptateur le met à jour de manière récursive dans la base de données.
 - Si l'enfant existe dans le tableau de l'objet métier source, mais non dans le tableau de l'objet métier extrait, l'adaptateur le crée de manière récursive dans la base de données.

- Si l'enfant existe dans le tableau de l'objet métier extrait, mais non dans le tableau de l'objet métier source, l'adaptateur le supprime de manière récursive de la base de données à moins que la valeur `KeepRelationship` des informations spécifiques à l'application de l'attribut représentant l'enfant dans le parent soit `true`. Dans ce cas, l'adaptateur ne supprime pas l'enfant de la base de données.

Opération UpdateWithDelete (mise à jour avec suppression)

Cette forme particulière de l'opération de mise à jour `Update` permet d'obtenir de meilleures performances que l'opération `Update`.

L'opération `UpdateWithDelete` requiert une interface `ChangeSummary`. L'interface `ChangeSummary` doit inclure des opérations de création et de suppression au niveau objet métier. Cela permet à l'adaptateur d'effectuer des opérations sans avoir à extraire les entités existantes du système EIS (ce qui nécessite un temps système important) et sans avoir à effectuer de comparaisons, l'interface `ChangeSummary` indiquant les actions à effectuer.

Si l'interface `ChangeSummary` est vide, l'adaptateur n'effectue aucune action sur la demande.

Opération DeltaUpdate (mise à jour delta)

Si l'opération dans l'interface `InteractionSpec` est `ApplyChanges` et que l'instruction n'existe pas dans le graphique métier, l'adaptateur effectue une opération de mise à jour delta. L'adaptateur inspecte l'interface `ChangeSummary` pour identifier l'opération de chaque objet métier dans la hiérarchie des entrées, puis effectue l'opération identifiée.

Les opérations de mise à jour delta diffèrent des opérations de mise à jour de la manière suivante :

- Dans une opération de mise à jour delta, aucune opération d'extraction ne survient avant la mise à jour.
- Aucune comparaison n'est effectuée entre l'objet métier entrant et l'objet métier contenu dans la base de données.
- Tous les enfants sont traités en fonction de l'instruction définie dans chaque objet enfant. Si aucune instruction n'est définie pour l'un des enfants, l'adaptateur renvoie un message d'erreur.

Lors de la mise à jour d'un objet métier hiérarchique à l'aide de l'opération `DeltaUpdate`, l'adaptateur procède aux étapes suivantes : Il ne traite que les modifications d'objet de l'interface `ChangeSummary` :

1. Il traite de manière récursive tous les enfants de type cardinalité simple de l'objet parent. Si un enfant est signalé comme étant requis dans la spécification de l'objet métier, il doit être présent dans l'objet des communications entrantes. Dans le cas contraire, l'opération `DeltaUpdate` échoue et l'adaptateur renvoie un message d'erreur.
2. Il affecte à toutes les valeurs de clé externe du parent qui font référence aux attributs contenus dans les enfants de type cardinalité simple les valeurs des enfants correspondants. Cette étape est nécessaire dans la mesure où des enfants de type cardinalité simple peuvent avoir été ajoutés à la base de données au cours des étapes précédentes, entraînant ainsi la création de valeurs de séquence supplémentaires.
3. Il met à jour l'objet en cours de traitement à l'aide d'une instruction SQL `Update` ou d'une procédure stockée. Tous les attributs simples de l'objet métier individuel sont mis à jour. L'adaptateur n'utilise pas de modifications au

niveau propriété pour déterminer les attributs devant être ajoutés à l'instruction de mise à jour : ils sont tous mis à jour. L'objet mis à jour devant être unique, l'adaptateur vérifie qu'une seule ligne est traitée en retour. Un message d'erreur est renvoyé si plusieurs lignes sont traitées.

4. Il affecte à toutes les valeurs de clé externe de tous les enfants de cardinalité N de l'objet courant qui fait référence aux attributs parent les valeurs parent correspondantes. En règle générale, ces valeurs ont déjà été renvoyées au cours du mappage des données. Cela peut toutefois ne pas être le cas pour les nouveaux enfants dans les conteneurs de cardinalité N. Cette étape garantit la validité de toutes les valeurs de clé externe de tous les enfants de cardinalité N avant la mise à jour de ces enfants.
5. Il met à jour tous les conteneurs de cardinalité N de l'objet courant.
Lors du traitement des objets enfant, l'instruction de chaque enfant est prise en compte et l'opération appropriée est effectuée. Les opérations autorisées sur un enfant dans une instruction DeltaUpdate sont les suivantes : create, delete et update.
 - Si une instruction Create est trouvée dans l'enfant, ce dernier est créé dans la base de données s'il s'agit d'un enfant avec droits de propriété. Les enfants sans droits de propriétés sont extraits pour que leur présence soit validée dans la base de données.
 - Si une instruction Delete est trouvée dans l'enfant, ce dernier est supprimé.
 - Si une instruction Update est trouvée dans l'enfant, ce dernier est mis à jour dans la base de données.

Opération Delete (suppression)

L'opération de suppression est effectuée en supprimant l'objet métier entrant, puis en extrayant l'intégralité de l'objet métier de la base de données. L'opération de suppression est ensuite appliquée de manière récursive à chaque objet métier de la hiérarchie.

L'opération de suppression prend en charge les suppressions physiques et logiques, suivant la valeur du paramètre StatusColumnName dans les informations spécifiques à l'application de l'objet. Si la valeur de StatusColumnName est définie, l'adaptateur procède à une opération de suppression logique. Si la valeur de StatusColumnName n'est pas définie, l'adaptateur procède à une opération de suppression physique.

Pour les suppressions physiques, l'adaptateur exécute les actions suivantes :

- Il supprime de manière récursive tous les objets métier enfant de type cardinalité multiple.
- Il supprime l'objet métier de niveau supérieur.
- Il supprime de manière récursive tous les objets métier enfant de type cardinalité simple contenus avec droits de propriété.

Pour les suppressions logiques, l'adaptateur exécute les actions suivantes :

- Il émet une instruction Update qui affecte à l'attribut d'état de l'objet métier la valeur spécifiée par les informations spécifiques à l'application de l'objet métier. L'adaptateur vérifie que la mise à jour concerne une seule ligne de la base de données et renvoie un message d'erreur dans le cas contraire.
- Il supprime logiquement tous les enfants de type cardinalité simple contenus avec droits de propriété et tous les enfants de type cardinalité multiple de manière récursive. L'adaptateur ne supprime pas les enfants de type cardinalité simple contenus sans droits de propriété.

Opération ApplyChanges

L'opération ApplyChanges s'applique à un grand nombre de situations. Elle permet à un objet métier nécessitant une opération de création, de mise à jour ou de suppression d'être traité en conséquence par l'adaptateur.

Si l'instruction de niveau supérieur existe dans l'objet métier, ce dernier est traité comme image postérieure. Si aucune instruction de niveau supérieur n'existe dans l'objet métier, l'interface ChangeSummary est traitée.

Informations spécifiques à l'application

Les informations spécifiques à l'application dans les définitions d'objet métier fournissent à l'adaptateur les instructions dépendantes de l'application concernant le traitement des objets métier. L'adaptateur analyse les informations spécifiques à l'application depuis les attributs ou l'instruction d'un objet métier ou depuis l'objet métier lui-même pour générer des requêtes pour les opérations de création, de mise à jour, d'extraction ou de suppression.

L'adaptateur stocke quelques-unes des informations spécifiques à l'application de l'objet métier en mémoire cache et les utilise pour générer des requêtes pour toutes les instructions.

Dans un objet métier étendu ou modifié, spécifique à l'application, les informations spécifiques à l'application qui se trouvent dans la définition de l'objet métier doivent respecter la syntaxe attendue par l'adaptateur.

Tâches associées

«Recherche d'objets de base de données», à la page 46

Après avoir configuré les propriétés de connexion, vous pouvez exécuter une requête sur les objets de base de données. Vous pouvez explorer l'arborescence des métadonnées pour comprendre la structure des objets du système EIS (Enterprise Information System) et sélectionner les objets requis pour la description de service.

Conventions de dénomination relatives aux objets métier

Les noms d'objet métier doivent refléter la structure qu'ils représentent (par exemple Customer ou Address). Ces noms sont généralement déterminés lors de la procédure d'importation des métadonnées de l'outil EMD (Enterprise Metadata Discovery), en fonction du nom indiqué par le système EIS (Enterprise Information System).

Les noms des objets métier doivent être convertis en casse chameau, dans laquelle les séparateurs tels que les espaces et les traits de soulignement sont supprimés et la première lettre de chaque mot est en majuscule (par exemple, ORDER_LINE_ITEM devient OrderLineItem).

Le graphique de l'objet métier parent doit porter le nom de l'objet métier contenu, suivi de BG (par exemple, CustomerBG pour un objet métier Customer).

Les noms d'objet métier n'ont aucune valeur sémantique pour l'adaptateur de la base de données.

Les objets métier contiennent des métadonnées spécifiques à la base de données. Ils peuvent comporter une chaîne telle que JDBC ou %AppName% comme préfixe, pour distinguer les deux types d'objet métier : spécifiques à l'application et génériques. Le reste du nom peut décrire la table ou la procédure stockée représentée par l'objet métier. Par exemple, si la définition d'objet métier est

générée pour la table Employee d'une application de base de données, telle que Human Resources (HR), le nom d'objet métier respectif est HREmployee.

Informations spécifiques à l'application au niveau de l'objet métier

Les informations spécifiques à l'application au niveau de l'objet métier permettent de spécifier le nom de la table de base de données correspondante et de fournir les informations nécessaires à une opération de suppression physique ou logique.

Au niveau de l'objet métier, les informations spécifiques à l'application comportent un code xml défini dans la définition de schéma jdbcasi.xsd, où :

- *TableName* identifie la table de base de données associée à l'objet métier,
- *StatusColumnName* correspond au nom de la colonne de base de données utilisée pour effectuer des opérations de suppression logiques et
- *StatusValue* représente la valeur indiquant qu'un objet métier est inactif ou supprimé.

Par exemple, supposons que la valeur suivante soit indiquée pour les informations spécifiques à l'application d'un objet métier Customer :

```
<jdbcasi:TableName>customer</jdbcasi:TableName>  
<jdbcasi:StatusColumnName>status</jdbcasi:StatusColumnName>  
<jdbcasi::StatusValue>deleted</jdbcasi:StatusValue>
```

Supposons que l'adaptateur reçoive une demande de suppression d'un client. Une telle demande oblige l'adaptateur à exécuter l'instruction SQL suivante :

```
UPDATE customer set status = 'deleted' where pkey = . . . .
```

Si le paramètre *StatusColumnName* n'est pas inclus ou qu'aucune valeur n'est spécifiée pour ce paramètre, l'adaptateur supprime physiquement l'objet métier de la base de données. En d'autres termes, si l'objet métier inclut le paramètre *StatusColumnName* dans ses informations spécifiques à l'application, l'adaptateur procède à une suppression logique. Si l'objet métier n'inclut pas le paramètre *StatusColumnName* dans ses informations spécifiques à l'application, l'adaptateur procède à une suppression physique.

Les opérations de mise à jour et de suppression peuvent utiliser la valeur de la propriété *StatusColumnName* :

- Pour supprimer de manière logique des données enfant, l'adaptateur utilise la valeur de son paramètre *StatusColumnName* pour obtenir le nom de la colonne d'état et le texte de la valeur d'état. Pour plus d'informations, voir «Opérations de mise à jour.»
- Lors d'une opération de suppression, l'adaptateur utilise la valeur de son paramètre *StatusColumnName* pour déterminer si l'intégralité de l'objet métier doit être supprimée de manière logique ou physique. Si le paramètre *StatusColumnName* contient une valeur, l'adaptateur procède à une suppression logique. Si le paramètre *StatusColumnName* ne contient aucune valeur, l'adaptateur procède à une suppression physique. Pour plus d'informations, voir «Opérations Delete (suppression).»

Les paramètres de l'ASI qui peuvent être utilisés avec les langues bidirectionnelles sont *TableName* et *StatusColumnName*. Le format de ces paramètres est converti en fonction des attributs définis pour la propriété *BiDi.Metadata*. Pour plus d'informations sur cette propriété, voir «Propriétés de configuration», à la page 75.

Informations d'instruction spécifiques à l'application

L'adaptateur met à jour les tables de base de données à l'aide de requêtes SQL ou de procédures stockées (groupes d'instructions SQL), comme indiqué dans les objets métier. Cette section décrit les procédures stockées et les éléments d'une définition de procédure stockée. Elle inclut un exemple de définition de procédure stockée.

Concepts associés

«Sélection et génération d'objets», à la page 38

Pour générer des objets métier, vous devez sélectionner des noeuds d'objet de base de données. L'assistant ESD (Enterprise Service Discovery) génère ensuite des objets métier pour les objets des noeuds sélectionnés.

Référence associée

«Attributs d'objets métier et leurs informations ASI (informations spécifiques à l'application)», à la page 40

Les attributs d'un objet métier sont générés à partir de la liste des colonnes de l'objet de base de données. L'assistant ESD (Enterprise Service Discovery) affecte le nom de la colonne au nom de l'attribut. Les caractères globalisés ne sont pas pris en charge dans les noms d'attribut. L'adaptateur ajoute le nom de l'attribut, son type et les informations spécifiques à l'application.

Présentation des procédures stockées :

Une procédure stockée est un groupe d'instructions SQL formant une unité logique et réalisant une tâche spécifique. Une procédure stockée rassemble un ensemble d'opérations ou de requêtes que l'adaptateur doit exécuter sur un objet dans un serveur de base de données.

L'adaptateur peut utiliser des instructions SQL simples pour les opérations de sélection, de mise à jour, d'extraction, de suppression ou d'extraction globale. Les noms des colonnes relatives aux instructions SQL proviennent de la propriété AppSpecificInfo d'un attribut. La clause WHERE est construite à l'aide des valeurs de clé spécifiées dans l'objet métier. Chaque requête couvre une seule table, à moins qu'elle ne désigne une vue.

L'adaptateur fait appel aux procédures stockées dans les cas suivants :

- avant le traitement d'un objet métier pour réaliser les processus opérationnels préparatoires ;
- après le traitement d'un objet métier pour réaliser les processus post-opérationnels ;
- pour réaliser un ensemble d'opérations sur un objet métier, au lieu d'utiliser une simple instruction Create, Update, Delete, Retrieve ou RetrieveAll.

Lors du traitement d'un objet métier hiérarchique, l'adaptateur peut utiliser une procédure stockée pour traiter l'objet métier de niveau supérieur ou l'un de ses objets métier enfant. Toutefois, chaque objet métier ou tableau d'objets métier doit avoir sa propre procédure stockée.

Définition de procédure stockée :

Les procédures stockées sont définies au niveau des instructions. Chaque définition de procédure stockée comprend les éléments suivants : StoredProcedureType, StoredProcedureName, ResultSet et Parameters.

L'élément **StoredProcedureType** définit le type de procédure stockée à utiliser et détermine à quel moment la procédure stockée est appelée. Les valeurs possibles sont les suivantes :

- BeforeCreateSP
- AfterCreateSP
- CreateSP
- BeforeUpdateSP
- AfterUpdateSP
- UpdateSP
- BeforeDeleteSP
- AfterDeleteSP
- DeleteSP
- BeforeRetrieveSP
- AfterRetrieveSP
- RetrieveSP
- BeforeRetrieveAllSP
- AfterRetrieveAllSP
- RetrieveAllSP

Remarque : Les types de procédure stockée associés à la propriété RetrieveAll ne s'appliquent qu'aux objets métier de niveau supérieur.

StoredProcedureName correspond au nom de la procédure stockée associée à l'élément StoredProcedureType approprié. Activée pour une utilisation avec les langues bidirectionnelles.

ResultSet détermine si la procédure stockée renvoie un résultat ou non (true|false). Si le jeu de résultats est renvoyé, un enfant de cardinalité N est créé pour l'objet métier en cours, à l'aide des valeurs renvoyées dans les lignes du jeu de résultats.

Parameters peut être une combinaison d'entrées seules (IP), de sorties seules (OP) et d'entrées et de sorties (IO). Dans le cas des procédures stockées Oracle, un jeu de résultats ne peut être renvoyé que comme paramètre de sortie. Dans ce cas, l'une des valeurs de la liste des paramètres correspond au jeu de résultats (RS). Les paramètres sont activés pour une utilisation avec les langues bidirectionnelles.

Voici un exemple de définition de procédure stockée :

```
<jdbcasi:JDBCBusinessObjectTypeMetadata
xmlns:jdbcasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/jdbc/metadata">
  <jdbcasi:TableName>customer</jdbcasi:TableName><jdbcasi:Operation>
    <jdbcasi:Name>Retrieve</jdbcasi:Name>
    <jdbcasi:StoredProcedures>
      <jdbcasi:StoredProcedureType>RetrieveSP</jdbcasi:StoredProcedureType>
      <jdbcasi:StoredProcedureName>retrieve_cust</jdbcasi:StoredProcedureName>
      <jdbcasi:ResultSet>false</jdbcasi:ResultSet>
      <jdbcasi:Parameters>
        <jdbcasi:Type>IP</jdbcasi:Type>
        <jdbcasi:PropertyName>primaryKey</jdbcasi:PropertyName>
      </jdbcasi:Parameters>
      <jdbcasi:Parameters>
        <jdbcasi:Type>OP</jdbcasi:Type>
        <jdbcasi:PropertyName>custCode</jdbcasi:PropertyName>
      </jdbcasi:Parameters>
      <jdbcasi:Parameters>
```

```

        <jdbcasi:Type>OP</jdbcasi:Type>
        <jdbcasi:PropertyName>firstName</jdbcasi:PropertyName>
    </jdbcasi:Parameters>
</jdbcasi:Parameters>
    <jdbcasi:Type>OP</jdbcasi:Type>
        <jdbcasi:PropertyName>lastName</jdbcasi:PropertyName>
    </jdbcasi:Parameters>
</jdbcasi:StoredProcedures>
</jdbcasi:StoredProcedures>
<jdbcasi:StoredProcedureType>AfterRetrieveSP</jdbcasi:StoredProcedureType>
<jdbcasi:StoredProcedureName>retrieve_cust</jdbcasi:StoredProcedureName>
    <jdbcasi:ResultSet>>false</jdbcasi:ResultSet>
    <jdbcasi:Parameters>
        <jdbcasi:Type>IP</jdbcasi:Type>
        <jdbcasi:PropertyName>primaryKey</jdbcasi:PropertyName>
    </jdbcasi:Parameters>
</jdbcasi:Parameters>
    <jdbcasi:Type>OP</jdbcasi:Type>
        <jdbcasi:PropertyName>custCode</jdbcasi:PropertyName>
    </jdbcasi:Parameters>
</jdbcasi:Parameters>
    <jdbcasi:Type>OP</jdbcasi:Type>
        <jdbcasi:PropertyName>firstName</jdbcasi:PropertyName>
    </jdbcasi:Parameters>
</jdbcasi:Parameters>
    <jdbcasi:Type>OP</jdbcasi:Type>
        <jdbcasi:PropertyName>lastName</jdbcasi:PropertyName>
    </jdbcasi:Parameters>
</jdbcasi:StoredProcedures>
</jdbcasi:Operation>
</jdbcasi:JDBCBusinessObjectTypeMetadata>

```

La propriété ReturnDummyBOForSP renvoie des paramètres de sortie même lorsque le jeu de résultats est true, mais vide. Dans le cas de la propriété RetrieveSP, un jeu de résultats est renvoyé. Si le jeu de résultats est vide, aucun objet métier n'est créé et il n'est pas possible d'extraire les paramètres de sortie renvoyés par l'appel de procédure. Si la propriété ReturnDummyBOForSP a la valeur true, un objet métier fictif comprenant des valeurs issues des paramètres d'entrée/sortie indiqués dans les attributs correspondants est renvoyé. La valeur par défaut de cette propriété est false.

Concepts associés

«Structure des objets métier», à la page 8

Chaque objet métier correspond à une vue ou une table de base de données et chaque attribut simple de l'objet correspond à une colonne de cette table ou de cette vue.

Informations d'attribut spécifiques à l'application

La présente rubrique décrit les informations d'attribut spécifiques (ASI) pour l'application et répertorie les paramètres pris en charge et leur description.

Les informations d'attribut spécifiques à l'application diffèrent selon que l'attribut est un attribut simple ou un attribut représentant un enfant ou un tableau d'objets métier enfant. Les informations spécifiques à l'application d'un attribut représentant un enfant diffèrent également selon que la relation parent-enfant est stockée dans l'enfant ou dans le parent.

Informations spécifiques à l'application des attributs simples

Pour les attributs simples, le format des informations spécifiques à l'application consiste en un certain nombre de paramètres et leur valeur. Le format des informations spécifiques à l'application est indiqué dans l'exemple de section suivant d'un fichier .xsd :

```
<jdbcasi:ColumnName>pkey</jdbcasi:ColumnName>
<jdbcasi:PrimaryKey>true</jdbcasi:PrimaryKey>
<jdbcasi:FixedChar>true</jdbcasi:FixedChar>
</jdbcasi:JDBCAttributeTypeMetadata>
</appinfo>
</annotation>
<simpleType>
  <restriction base="string">
    <maxLength value="10"/>
  </restriction>
</simpleType>
</element>
<element name="custCode" type="string">
<annotation>
<appinfo source="WBI">
<jdbcasi:JDBCAttributeTypeMetadata
xmlns:jdbcasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/jdbc/metadata">
  <jdbcasi:ColumnName>ccode</jdbcasi:ColumnName>
  <jdbcasi:ForeignKey>custinfoObj/custCode</jdbcasi:ForeignKey>
</jdbcasi:JDBCAttributeTypeMetadata>
</appinfo>
</annotation>
</element>
<element name="firstName" type="string">
<annotation>
<appinfo source="WBI">
<jdbcasi:JDBCAttributeTypeMetadata
xmlns:jdbcasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/jdbc/metadata">
  <jdbcasi:ColumnName>fname</jdbcasi:ColumnName>
</jdbcasi:JDBCAttributeTypeMetadata>
</appinfo>
</annotation>
</element>
<element name="lastName" type="string">
<annotation>
<appinfo source="WBI">
<jdbcasi:JDBCAttributeTypeMetadata
xmlns:jdbcasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/jdbc/metadata">
  <jdbcasi:ColumnName>lname</jdbcasi:ColumnName>
</jdbcasi:JDBCAttributeTypeMetadata>
</appinfo>
</annotation>
</element>
```

Le seul paramètre requis pour qu'un attribut simple soit traité par le connecteur est le nom de colonne. Par exemple, pour indiquer uniquement le nom de colonne, utilisez le format suivant (ccode correspond au code client) :

```
<jdbcasi:ColumnName>ccode</jdbcasi:ColumnName>
```

Les paramètres ASI des attributs qui peuvent être utilisés avec les langues bidirectionnelles sont ColumnName et ForeignKey. Le format de ces paramètres est converti en fonction des attributs définis pour la propriété BiDi.Metadata. Pour plus d'informations sur cette propriété, voir la rubrique "Propriétés de configuration" dans la section "Référence". Pour plus d'informations sur les propriétés bidirectionnelles, reportez-vous au document technique général et au document technique de l'adaptateur relatifs au support bidirectionnel, sur le site Web d'IBM developerWorks.

Le tableau intitulé «Paramètres des informations d'attribut spécifiques à l'application» répertorie chaque paramètre et sa description.

Paramètres des informations d'attribut spécifiques à l'application

Paramètre	Description
ByteArray	Si la valeur de ce paramètre est true, l'adaptateur lit et écrit des données binaires dans la base de données et envoie ces données sous forme de chaîne au serveur d'applications. La valeur par défaut est false. Pour plus d'informations, voir «Utilisation des données binaires.»
ColumnName	La valeur de ce paramètre correspond au nom de la colonne de base de données de cet attribut. Activée pour une utilisation avec les langues bidirectionnelles.
FixedChar	<p>Ce paramètre indique si la longueur de l'attribut est fixe lorsque les colonnes de la table sont de type CHAR, et non VARCHAR. Par exemple, si un attribut spécifique est lié à une colonne de type CHAR, l'adaptateur complète la valeur d'attribut par des blancs jusqu'à ce que la longueur maximale de l'attribut soit atteinte, lors de l'interrogation de la base de données. La valeur par défaut est false.</p> <p>Ce paramètre doit être mis à jour manuellement dans le fichier .xsd de l'objet métier. Vous pouvez éditer ce fichier en mode texte ou à l'aide de l'éditeur d'objet métier de WebSphere Integration Developer. Vérifiez qu'aucune erreur de validation ne se produit dans le fichier .xsd après sa mise à jour. Reportez-vous à l'exemple de code relatif à ce paramètre dans le tableau ci-après.</p>
ForeignKey	<p>La valeur de cette propriété diffère selon que la relation parent-enfant est stockée dans l'objet métier parent ou dans l'enfant.</p> <p>Si elle est stockée dans le parent, définissez la valeur pour inclure le type de l'objet métier enfant ainsi que le nom de l'attribut dans l'enfant à utiliser en tant que clé externe (<i>ChildBOname/ChildPropertyName</i>).</p> <p>Si elle est stockée dans l'enfant, définissez la valeur pour inclure uniquement le nom de l'attribut dans le parent à utiliser en tant que clé externe.</p> <p>Si un attribut n'est pas une clé externe, n'incluez pas ce paramètre dans les informations spécifiques à l'application.</p> <p>Ce paramètre est activé pour une utilisation avec les langues bidirectionnelles.</p>
KeepRelationship	Si sa valeur est true, ce paramètre empêche la suppression d'un objet métier enfant lors d'une opération de mise à jour.

Paramètre	Description
OrderBy	Si une valeur est spécifiée pour ce paramètre et que l'attribut se trouve dans un objet métier enfant, l'adaptateur utilise la valeur de l'attribut dans la clause ORDER BY des requêtes d'extraction. L'adaptateur peut extraire les objets métier enfant dans l'ordre croissant (ASC) ou décroissant (DESC). Si vous n'incluez pas ce paramètre dans les informations spécifiques à l'application, l'adaptateur n'utilise pas cet attribut lors de la spécification de l'ordre d'extraction.
Ownership	Ce paramètre indique qu'un objet métier enfant appartient au parent. Si sa valeur est true, les opérations de création, mise à jour et suppression sur l'objet métier sont autorisées. Si sa valeur est false, aucune de ces mises à jour ne peut être appliquée à l'objet métier enfant. Lors de la création du parent, l'existence de l'enfant est validée pour garantir que l'intégrité de la relation est conservée dans la base de données.
PrimaryKey	Si la valeur de ce paramètre est true, la colonne qui lui est associée correspond à une clé dans la table correspondante de la base de données.
UniqueIdentifier (UID)	L'adaptateur utilise ce paramètre pour générer l'ID unique de l'objet métier. Il prend en charge la génération de séquences et de colonnes d'identité (<i>UID=AUTO SequenceName</i>). Les séquences ne peuvent être définies que pour les bases de données DB2 et Oracle. Les colonnes d'identité peuvent être définies pour DB2 et Microsoft SQL Server. Si l'attribut ne requiert pas d'ID unique, n'incluez pas ce paramètre dans les informations spécifiques à l'application.

Exemple de paramètre FixedChar dans le fichier .xsd de l'objet métier

```

<element name="primaryKey">
  <annotation>
    <appinfo source="WBI">
      <jdbcasi:JDBCAttributeTypeMetadata
xmlns:jdbcasi="http://www.ibm.com/xmlns/prod/websphere/j2ca/jdbc/metadata">
        <jdbcasi:ColumnName>pkey</jdbcasi:ColumnName>
        <jdbcasi:PrimaryKey>>true</jdbcasi:PrimaryKey>
        <jdbcasi:FixedChar>>true</jdbcasi:FixedChar>
      </jdbcasi:JDBCAttributeTypeMetadata>
    </appinfo>
  </annotation>
  <simpleType>
    <restriction base="string">
      <maxLength value="10"/>
    </restriction>
  </simpleType>
</element>

```

Référence associée

«Attributs d'objets métier et leurs informations ASI (informations spécifiques à l'application)», à la page 40
Les attributs d'un objet métier sont générés à partir de la liste des colonnes de l'objet de base de données. L'assistant ESD (Enterprise Service Discovery) affecte le nom de la colonne au nom de l'attribut. Les caractères globalisés ne sont pas pris en charge dans les noms d'attribut. L'adaptateur ajoute le nom de l'attribut, son type et les informations spécifiques à l'application.

Installation de l'adaptateur

Cette section aborde les conditions requises par l'installation de l'adaptateur pour JDBC et décrit la structure de fichiers installée.

Pour plus d'informations sur la manière d'installer l'adaptateur, voir Installation d'adaptateurs IBM WebSphere.

Environnement de l'adaptateur

L'environnement matériel et logiciel requis par l'adaptateur est consultable en ligne.

Pour en prendre connaissance, reportez-vous au document «IBM WebSphere Adapters and IBM WebSphere Business Integration Adapters: software requirements» à l'adresse suivante : <http://www-1.ibm.com/support/docview.wss?uid=swg27006249>. Sélectionnez votre adaptateur dans la liste des adaptateurs WebSphere.

Informations d'installation spécifiques à l'adaptateur pour JDBC

Des conditions d'installation en matière de pilote JDBC et de configuration du magasin d'événements sont requises pour l'adaptateur pour JDBC.

Prérequis du pilote JDBC

Vous devez configurer le pilote JDBC pour que son fichier JAR se trouve dans le chemin d'accès aux classes.

L'ESD (Enterprise Service Discovery) JDBC se trouve dans un projet de l'assistant (Enterprise Service Discovery). Le fichier JAR du pilote JDBC doit se trouver dans le chemin d'accès aux classes du projet JDBC dans cet outil pour que vous puissiez exécuter le processus EMD (Enterprise Metadata Discovery) JDBC.

Configuration du magasin d'événements

Vous devez configurer le magasin d'événements de la base de données pour permettre le traitement des événements entrants. Des exemples de script sont fournis pour configurer le magasin d'événements pour la base de données IBM DB2, Oracle ou Microsoft SQLServer, de la manière suivante :

- `WBIA_JDBC_EventStore_DB2.sql`
- `WBIA_JDBC_EventStore_Oracle.sql`
- `WBIA_JDBC_EventStore_MSSQL.sql`

Vous devez configurer des déclencheurs dans les tables utilisateur de sorte que les modifications apportées à ces tables puissent générer automatiquement des événements stockés dans le magasin d'événements.

Structure de fichiers installée

Une fois que vous avez installé l'adaptateur, vous pouvez afficher les fichiers et répertoires installés, qui se trouvent tous sous le répertoire d'installation.

Par exemple, si le répertoire d'installation de l'adaptateur est `c:\WebSphereBI`, le fichier `CWYBC_JDBC.rar` possède le chemin d'accès absolu suivant :
`c:\WebSphereBI.\adapter\JDBC\deploy\CWYBC_JDBC.rar`

Le fichier RAR (Resource Adapter Archive) contient les fichiers de l'adaptateur et de l'outil de reconnaissance de service d'entreprise.

Les plateformes UNIX et Windows partagent la même structure de répertoires et de fichiers installée à la différence près que la désignation du chemin d'accès aux répertoires n'est pas la même (barre oblique pour (/) pour UNIX et barre oblique inversée (\) pour Windows).

Structure des répertoires et des fichiers pour UNIX/Linux

Le tableau ci-après répertorie les répertoires et fichiers UNIX/Linux de WebSphere Adapter for JDBC. Ces répertoires et fichiers sont regroupés en catégories.

Catégorie de fichiers et répertoires	Répertoires et fichiers
Fichier RAR de l'adaptateur pour JDBC	adapter/deploy/CWYBC_JDBC.rar
Fichier zip du module d'extension IBM Support Assistant	adapter/ISAPugin/com.ibm.esupport.client.SS6FE6_RAJDBC.zip
Fichier ILTM (IBM Tivoli License Manager)	adapter/5724L77E060000.sys
Fichiers de messages	adapter/messages/CWYBC_JDBC_messages.tar
Fichiers de messages de classes Foundation	adapter/messages/CWYBS_AdapterFoundation_messages.tar
Fichier de remarques d'ICU4J	adapter/notices.txt
Exemple de fichier EAR d'application	adapter/samples/Apps/JDBCApp.ear
Exemples de script pour IBM DB2	adapter/samples/scripts/scripts_db2.sql
Exemples de script pour Oracle	adapter/samples/scripts_oracle.sql

Structure des répertoires et des fichiers pour Windows

Le tableau ci-après répertorie les répertoires et fichiers Windows de WebSphere Adapter for JDBC. Ces répertoires et fichiers sont regroupés en catégories.

Catégorie de fichiers et répertoires	Répertoires et fichiers
Fichier RAR de l'adaptateur pour JDBC	adapter\deploy\CWYBC_JDBC.rar
Fichier zip du module d'extension IBM Support Assistant	adapter\ISAPugin\com.ibm.esupport.client.SS6FE6_RAJDBC.zip

Catégorie de fichiers et répertoires	Répertoires et fichiers
Fichier ILTM (IBM Tivoli License Manager)	adapter\5724L77E060000.sys
Fichiers de messages	adapter\messages\CWYBC_JDBC_messages.zip
Fichiers de messages de classes Foundation	adapter\messages\CWYBS_AdapterFoundation_messages.zip
Fichier de remarques d'ICU4J	adapter\notices.txt
Exemple de fichier EAR d'application	adapter\samples\Apps\JDBCApp.ear
Exemples de script pour IBM DB2	adapter\samples\scripts\scripts_db2.sql
Exemples de script pour Oracle	adapter\samples\scripts_oracle.sql

Création du projet d'adaptateur

Vous devez créer un projet d'adaptateur dans WID (WebSphere Integration Developer). Générez ensuite les objets métier et les blocs de construction de service à l'aide de l'assistant ESD (WebSphere Integration Developer) de WebSphere. Définissez les propriétés de configuration. Enfin, déployez le projet d'adaptateur sur le serveur d'applications.

Avant de déployer l'adaptateur, vous devez installer les produits suivants :

- WebSphere Integration Developer, Version 6.0
- WebSphere Process Server : à l'aide de la console d'administration de WebSphere Process Server, déployez le projet d'adaptateur sur le serveur d'applications et reconfigurez les valeurs des propriétés.

Reportez-vous aux instructions d'installation d' WebSphere Process Server à l'adresse suivante : <http://www.ibm.com/software/integration/wps>.

Les adaptateurs WebSphere ne peuvent être installés que sur des ordinateurs dotés d'un système d'exploitation Windows ou Linux. A partir de là, ils peuvent être déployés sur des systèmes UNIX. Chaque adaptateur est installé sous forme de fichier RAR (Resource Adapter Archive).

En principe, le déploiement d'un adaptateur se déroule de la même manière que celui de tout autre composant sur WebSphere Process Server. Pour plus d'informations sur le déploiement des composants sur WebSphere Process Server, reportez-vous au guide de l'utilisateur de WebSphere Integration Developer, à l'adresse suivante : <http://www.ibm.com/software/integration/wid>.

Pour créer le projet d'adaptateur, vous devez effectuer les tâches suivantes :

- Créez un projet pour l'adaptateur
- Ajoutez des bibliothèques fournisseur
- Générez des objets métier et des blocs de construction de service
- Définissez les valeurs des propriétés de configuration, puis sauvegardez les valeurs et les artefacts dans un nouveau module Business Integration
- Déployez le projet d'adaptateur sur le serveur et démarrez l'application

Création d'un projet pour l'adaptateur

La première tâche du déploiement de l'adaptateur consiste à créer un projet de connecteur J2EE pour l'adaptateur. Le fichier RAR (Resource Adapter Archive) de l'adaptateur pour JDBC doit être importé dans WID (WebSphere Integration Developer). Ceci définit le projet dans votre espace de travail de WebSphere Integration Developer.

1. Démarrez WebSphere Integration Developer

Pour plus de détails, reportez-vous au guide de l'utilisateur de WebSphere Integration Developer, à l'adresse suivante : <http://www.ibm.com/software/integration/wid>.

2. Importez le fichier RAR

Dans WebSphere Integration Developer, accédez à la perspective Business Integration. Cliquez sur **Fichier > Importer**. Dans la fenêtre de sélection, choisissez la source d'importation **Fichier RAR** et cliquez sur **Suivant**.

Dans la fenêtre Importation du connecteur, sélectionnez l'emplacement du fichier RAR à l'aide du bouton **Parcourir**. Un nom de projet apparaît automatiquement dans la zone **Connector project**, mais vous pouvez le modifier si vous le souhaitez.

Désélectionnez la case **Ajouter un module à un projet EAR**.

Cliquez sur **Terminer** pour importer le fichier RAR. Un projet de connecteur J2EE est créé dans l'espace de travail.

3. Ajoutez le pilote JDBC au projet de connecteur

Vous devez ajouter le pilote JDBC au projet de connecteur pour qu'il soit intégré au fichier EAR (Enterprise Application Archive) à déployer sur le serveur d'applications. Cette opération peut être effectuée une fois que vous avez importé le fichier RAR ou après que vous ayez installé le projet EAR sur le serveur d'applications.

Une fois que vous avez importé le fichier RAR, vous devez ajouter le fichier JAR dans le dossier approprié de l'espace de travail. Par exemple, le fichier JAR peut être ajouté à l'emplacement suivant : `C:\workspace\CWYBC_JDBC\connectorModule`

En revanche, si vous souhaitez ajouter le pilote JDBC au projet de connecteur après avoir installé le projet EAR sur le serveur d'applications, après avoir installé l'application, vous devez ajouter le fichier JAR au sous-répertoire RAR du répertoire installedApps de WebSphere Process Server. Pour plus de détails sur l'installation de l'application sur le serveur d'applications, voir «Déploiement du projet d'adaptateur».

Tâches associées

«Déploiement du projet d'adaptateur», à la page 53

Votre fichier de projet correspond à un projet de connecteur J2EE Connector dans l'espace de travail de WebSphere. Vous devez l'exporter dans votre système de fichiers local sous forme de fichier EAR (Enterprise Application Archive). Vous devez ensuite fournir les informations d'authentification pour la sécurité du connecteur JCA avant de pouvoir télécharger et installer le fichier EAR du projet sur le serveur d'applications.

Ajout de bibliothèques fournisseur

Une fois que vous avez créé le projet dans WID (WebSphere Integration Developer), vous devez ajouter une référence au pilote JDBC de votre projet.

Ajoutez le pilote JDBC au chemin de compilation Java

Dans la fenêtre **Configuration** de WebSphere Integration Developer, cliquez avec le bouton droit sur **Projet de connecteur**. Sélectionnez **Propriétés**.

Pour ajouter le fichier JAR externe, cliquez sur le bouton **Chemin de compilation Java**. Sélectionnez l'onglet **Bibliothèques**, puis cliquez sur **Ajouter des fichiers JAR externes**. Dans la fenêtre Système de fichiers, accédez à **Pilote JDBC** et sélectionnez le fichier JAR.

Génération d'objets métier

Pour générer des objets métier, vous devez d'abord définir les propriétés de connexion. Vous exécutez ensuite une requête sur les objets de base de données afin de pouvoir sélectionner ceux nécessaires pour la description du service. Vous devez ensuite spécifier des valeurs pour les propriétés de sélection des fichiers d'importation et d'exportation. Enfin, vous définissez les propriétés de configuration et sauvegardez les artefacts et les valeurs des propriétés dans un nouveau module Business Integration.

Avant de commencer les tâches de génération des objets métier, vous pouvez consulter les détails de cette procédure dans les sections relatives à l'importation des métadonnées, la recherche des fonctionnalités du système et les descriptions des données.

Importation de métadonnées

L'assistant ESD (Enterprise Service Discovery) de WebSphere Adapter pour JDBC permet de rechercher des objets dans une base de données et de générer des objets métier à partir des objets sélectionnés. Le support ESD (Enterprise Service Discovery) inclut également la génération des constructions de service qui permettent d'exécuter l'adaptateur comme un composant SCA (Service Component Architecture).

Les objets à partir desquels les objets métier peuvent être créés incluent les tables, vues, procédures stockées et synonymes/noms d'usage. L'adaptateur permet de rechercher les objets en générant une liste de tous les schémas de la base de données. Chaque schéma contient une liste des tables, vues, procédures stockées et synonymes/noms d'usage que vous pouvez sélectionner et pour lesquels vous pouvez demander à l'assistant ESD (Enterprise Service Discovery) de générer les objets métier correspondants. L'assistant analyse les métadonnées des objets et génère les attributs dans l'objet métier. Les attributs sont générés en fonction des noms de colonne de tous les objets de base de données. Pour récapituler ses fonctions, l'assistant ESD (Enterprise Service Discovery) :

- Génère un objet métier correspondant à un objet de base de données.
- Génère les propriétés de l'objet métier correspondant aux propriétés de l'objet de base de données.
- Définit les informations spécifiques à l'application sur l'objet métier.
- Fournit des descriptions de service (de communications entrantes et sortantes) utilisées pour générer les fichiers d'importation, d'exportation et les fichiers WSDL (Web Services Description Language).

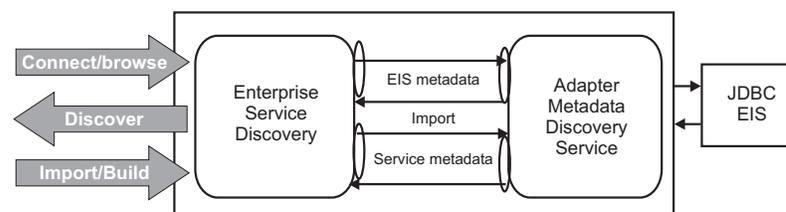


Diagramme ESD de niveau supérieur

Recherche des fonctionnalités du système

L'adaptateur recherche des objets métier dans une base de données en analysant cette dernière pour identifier les schémas. Il crée ensuite une liste de tous les objets de la base de données et l'affiche sous forme d'arborescence.

Les schémas sont affichés comme les noeuds de niveau supérieur de l'arborescence. Les noeuds de schéma ne peuvent pas être sélectionnés pour génération.

Chaque schéma contient des noeuds intitulés Tables, Vues, Procédures stockées et Synonymes/Noms d'usage. Ces noeuds ne peuvent pas être sélectionnés pour génération. Les objets répertoriés sous ces noeuds sont les noms des tables, les vues, les procédures stockées et les synonymes/noms d'usage. Ces noeuds sont marqués comme pouvant être sélectionnés pour génération. Si aucun synonyme, table, vue ou procédure stockée n'existe pour un schéma particulier, aucun de ces éléments n'est répertorié.

Vous pouvez spécifier des propriétés de filtre si vous souhaitez restreindre la liste des schémas affichés dans l'arborescence ; sinon, tous les schémas sont affichés. Le paragraphe «Propriétés de filtre et de noeud» de la section «Référence» décrit les propriétés à spécifier.

Vous pouvez sélectionner un noeud de tables, de vues, de procédures stockées ou de synonymes/noms d'usage, puis cliquer sur **Filtre** ; l'assistant ESD (Enterprise Service Discovery) lance alors une requête sur une propriété ObjectNameFilter. Vous pouvez utiliser la propriété ObjectNameFilter pour filtrer la liste des objets de base de données à afficher. Le paragraphe «Propriétés de filtre et de noeud» de la section «Référence» décrit la propriété ObjectNameFilter.

Référence associée

«Propriétés de filtre et de noeud», à la page 88

Lors de la reconnaissance des objets métier, vous pouvez spécifier des propriétés de filtre si vous souhaitez restreindre la liste des schémas affichés dans l'arborescence ; sinon, tous les schémas sont affichés. Vous pouvez définir la propriété de noeud pour restreindre la liste des objets de base de données à afficher.

Sélection et génération d'objets :

Pour générer des objets métier, vous devez sélectionner des noeuds d'objet de base de données. L'assistant ESD (Enterprise Service Discovery) génère ensuite des objets métier pour les objets des noeuds sélectionnés.

Vous pouvez sélectionner plusieurs noeuds d'objet de base de données. Lorsque vous spécifiez des propriétés de filtre, si vous cochez la case **Ajouter l'ASI de l'objet métier**, pour chaque noeud que vous sélectionnez, l'assistant ESD (Enterprise Service Discovery) vous demande les paramètres au niveau objet de StatusColumnName, de StatusValue et de l'association des procédures stockées.

Pour StatusColumnName, une liste des noms de colonne réels est affichée pour l'objet particulier ; à partir de cette liste, vous pouvez définir le paramètre StatusColumnName. Vous devez spécifier le paramètre StatusValue. Ces valeurs sont définies dans les informations ASI au niveau de l'objet métier.

Vous pouvez choisir d'associer des procédures stockées aux objets métier. Une liste de tous les types de procédure stockée pris en charge est affichée. Chaque type de procédure stockée contient une liste des procédures stockées disponibles dans la base de données. Une procédure stockée peut être affectée à un type de procédure stockée particulier.

Pour chaque procédure stockée affectée, une liste des paramètres d'entrée et de sortie de la procédure stockée est présentée. Chaque paramètre contient une liste des attributs de l'objet métier. Le type du paramètre est indiqué dans la description de la propriété. Vous pouvez sélectionner un attribut d'objet métier pour chaque paramètre de procédure stockée. Tous les types de procédure stockées auxquels une procédure stockée a été affectée sont ajoutés à l'ASI des instructions de l'objet métier. Voir «ASI des instructions.»

Une fois que vous avez sélectionné les objets de base de données, vous devez définir les valeurs des propriétés de sélection. L'assistant ESD (Enterprise Service Discovery) interroge les propriétés de sélection. Pour plus de détails sur ces propriétés, voir «Propriétés de sélection» dans la section «Référence».

Tâches associées

«Propriétés de sélection», à la page 89

Une fois que vous avez sélectionné les objets de base de données, vous devez définir les valeurs des propriétés de sélection.

Référence associée

«Informations d'instruction spécifiques à l'application», à la page 27

L'adaptateur met à jour les tables de base de données à l'aide de requêtes SQL ou de procédures stockées (groupes d'instructions SQL), comme indiqué dans les objets métier. Cette section décrit les procédures stockées et les éléments d'une définition de procédure stockée. Elle inclut un exemple de définition de procédure stockée.

Descriptions de données

Les descriptions de données correspondent à certaines des constructions de service générées par le processus ESD (Enterprise Service Discovery) pour permettre à l'adaptateur d'être exécuté en tant que composant SCA (Service Component Architecture).

La description des données inclut une définition de la structure et du contenu des objets métier de l'adaptateur qui est transmise entre l'application client et l'adaptateur lors de l'exécution. Cette description permet à l'application client de créer les objets de données appropriés pour les demandes et d'interpréter les objets de données renvoyés comme réponses. La description de données, générée à partir des composants de la base de données, est représentée sous forme de schéma XML.

- Les objets métier sont mappés avec une définition de type complexe.
- Les attributs de l'objet métier sont mappés avec des définitions de type d'élément.
- Les informations d'objet métier spécifiques à l'application sont contenues dans des annotations dans le type complexe.
- Les informations spécifiques à l'application de chaque propriété de l'objet métier sont contenues dans des annotations pour les types d'élément.

Remarque : Le modèle des propriétés spécifiques à l'application de l'objet métier et le niveau d'attribut sont définis dans le schéma des métadonnées de

l'adaptateur JDBC. Le nom du fichier de schéma est JDBCASI.xsd. Le fichier de schéma généré contient une référence à ce modèle dans ses annotations.

Schéma d'objet métier et ses informations ASI (informations spécifiques à l'application) :

Le schéma de l'objet métier est généré à partir des composants de base de données que vous sélectionnez. Chaque composant est converti en objet métier de niveau supérieur.

L'assistant ESD (Enterprise Service Discovery) génère le nom de l'objet métier au format *PréfixeNomSchémaNomObjet*,

- *Préfixe* correspond à la valeur spécifiée dans la propriété de connexion Préfixe. Le préfixe n'est pas requis ; s'il n'est pas spécifié, aucun préfixe n'est ajouté au nom de l'objet métier.
- *NomSchéma* correspondant au nom du schéma auquel l'objet appartient.
- *NomObjet* correspond au nom de la table, de la vue, de la procédure stockée ou du synonyme/nom d'usage.

Les caractères globalisés ne sont pas pris en charge dans le nom de l'objet métier.

L'assistant ESD (Enterprise Service Discovery) affecte à l'attribut ASI TableName une valeur au format *nomschéma.nomtable*. Il définit les informations ASI au niveau objet métier telles qu'elles sont répertoriées dans le tableau «Informations spécifiques à l'application d'objets métier (ASI)». Les opérations que vous sélectionnez sont définies dans l'objet métier. Tous les objets métier générés possèdent la même structure, quel que soit le type d'objet à partir duquel ils sont générés (table, vue, procédure stockée ou synonyme/nom d'usage). Ils ont tous des attributs basés sur les colonnes et les informations ASI TableName reçoivent le nom de l'objet.

Informations spécifiques à l'application d'objets métier (ASI)

Informations ASI des objets métier	Assistant de définition par reconnaissance de service d'entreprise	Informations complémentaires
TableName	Oui	Nom réel de la colonne
StatusColumnName	Oui	Spécifié lors de la sélection de l'objet
StatusValue	Oui	Spécifié lors de la sélection de l'objet

Tous les objets métier créés se trouvent au niveau supérieur. L'assistant ESD (Enterprise Service Discovery) ne crée pas d'objets métier récursifs (enfant). L'assistant ESD (Enterprise Service Discovery) génère également des graphiques métier pour tous les objets métier, car ils se trouvent tous au niveau supérieur. Le nom du graphique métier correspond au nom de l'objet métier suivi de «BG». Par exemple, si un objet métier s'intitule JDBCSchema1Customer, son graphique métier a pour nom JDBCSchema1CustomerBG. Les opérations définies dans l'objet métier le sont également dans le graphique métier.

Attributs d'objets métier et leurs informations ASI (informations spécifiques à l'application) :

Les attributs d'un objet métier sont générés à partir de la liste des colonnes de l'objet de base de données. L'assistant ESD (Enterprise Service Discovery) affecte le nom de la colonne au nom de l'attribut. Les caractères globalisés ne sont pas pris en charge dans les noms d'attribut. L'adaptateur ajoute le nom de l'attribut, son type et les informations spécifiques à l'application.

Les types renvoyés par les métadonnées JDBC sont mappés aux types d'attribut d'objet métier, comme indiqué dans le tableau «Types d'attribut des colonnes de métadonnées JDBC et des objets métier.» Seuls les types JDBC répertoriés sont pris en charge par l'adaptateur. Les colonnes dont les types ne sont pas répertoriés ne sont pas ajoutées à l'objet métier. Un message d'information est généré et indique par exemple : La colonne intitulée xxxx de la table yyyy n'est pas prise en charge et ne sera pas ajoutée à l'objet métier.

Types d'attribut des colonnes de métadonnées JDBC et des objets métier

Type de colonne de métadonnées JDBC	Type d'attribut d'objets métier
BIT	BOOLEAN
CHAR LONGVARCHAR VARCHAR	STRING
INTEGER NUMERIC SMALLINT TINYINT BIGINT	INTEGER
TIME TIMESTAMP DATE	DATE
DECIMAL	STRING
DOUBLE FLOAT	DOUBLE
REAL	FLOAT

Le tableau intitulé «Informations sur les attributs» répertorie les informations sur les attributs définies par l'assistant ESD (Enterprise Service Discovery) et décrit comment il est défini.

Informations sur les attributs

Informations sur les attributs	Définir par reconnaissance de service d'entreprise	Informations supplémentaires
Cardinalité	Non	
Nom	Oui	Nom de l'attribut. Activé pour les langues bidirectionnelles.
MinOccurs/MaxOccurs	Oui	Si la colonne ne correspond pas à une clé principale et qu'elle ne peut pas avoir la valeur null, l'attribut est requis et possède une valeur au moins égale à 1.

Informations sur les attributs	Définir par reconnaissance de service d'entreprise	Informations supplémentaires
Type	Oui	Défini dans la table «Types d'attribut des colonnes de métadonnées JDBC et des objets métier».

L'assistant ESD (Enterprise Service Discover) définit les informations ASI des attributs dans l'objet métier, comme illustré dans le tableau «Informations propres à l'application de l'attribut». Pour plus d'informations sur les informations ASI des attributs, voir «ASI des attributs simples.»

Informations propres à l'application de l'attribut

ASI des attributs	Définir par reconnaissance de service d'entreprise	Informations supplémentaires
ColumnName	Oui	Nom réel de la colonne. Activé pour les langues bidirectionnelles.
FixedChar	Non	Doit être mis à jour manuellement dans le fichier .xsd de l'objet métier. Utilisez le mode texte ou l'éditeur d'objets métier de WebSphere Integration Developer pour modifier le fichier. Une fois que vous avez mis à jour le fichier, vérifiez qu'aucune erreur de validation n'a été générée. Consultez un exemple d'attribut FixedChar de fichier .xsd dans la section «ASI des attributs simples.»
ForeignKey	Non	
OrderBy	Non	
PrimaryKey	Oui	Si la colonne correspond à une clé principale, l'attribut PrimaryKey a la valeur true.
UID	Non	

Si vous choisissez d'ajouter des procédures stockées aux objets métier, les informations ASI des instructions sont spécifiées dans le tableau «Informations d'instruction propres à l'application». Pour plus d'informations sur les types de procédure stockées valides, voir la section «Informations d'instruction propres à l'application».

Informations d'instruction propres à l'application

Élément ASI des instructions ou paramètres de procédure stockée	Assistant de définition par reconnaissance de service d'entreprise	Informations supplémentaires
Paramètres	Oui	Répertorie les paramètres des procédures stockées. Activé pour les langues bidirectionnelles.
PropertyName	Oui	Nom de l'attribut d'objet métier que vous sélectionnez. Activé pour les langues bidirectionnelles.
ResultSet	Non	Si la procédure stockée renvoie un jeu de résultats, vous devez affecter la valeur true à ce paramètre dans la définition de l'objet métier.
StoredProcedure	Oui	Nom de la procédure stockée. Activé pour les langues bidirectionnelles.
Type StoredProcedure	Oui	Choisissez une valeur dans la liste des types.
Type	Oui	Définissez le type de paramètre de procédure stockée (IP/OP/IO).

Génération d'objets métier hiérarchiques

L'assistant ESD (Enterprise Service Discovery) génère des objets métier à plat. Il n'utilise pas les contraintes de clé externe définies dans la base de données entre les différentes tables pour générer automatiquement les relations. Les liens doivent être créés manuellement. Vous pouvez mettre à jour les définitions d'objet métier en mode texte ou à l'aide de l'éditeur d'objets métier.

Vous trouverez ci-après un exemple de fichier de définition .xsd pour les objets métier enfant à cardinalité simple et multiple. L'élément custInfoObj est un objet métier enfant à cardinalité simple et l'élément addressObj, un objet métier enfant à cardinalité multiple.

```
<element name="addressObj" minOccurs="1" type="Address:Address"
maxOccurs="unbounded">
  <annotation>
    <appinfo source="WBI">
      <pasi:JDBCAttributeTypeMetadata xmlns:pasi=
"urn:app:jdbc:asi">
        <pasi:Ownership>true</pasi:Ownership>
      </pasi:JDBCAttributeTypeMetadata>
    </appinfo>
  </annotation>
</element>
<element name="custInfoObj" minOccurs="0" type=
"CustInfo:CustInfo" maxOccurs="1">
  <annotation>
    <appinfo source="WBI">
      <pasi:JDBCAttributeTypeMetadata xmlns:pasi=
"urn:app:jdbc:asi">
        <pasi:Ownership>false</pasi:Ownership>
```

```
</pasi:JDBCAttributeTypeMetadata>
</appinfo>
</annotation>
</element>
```

Référence associée

«Informations d'attribut spécifiques à l'application», à la page 29

La présente rubrique décrit les informations d'attribut spécifiques (ASI) pour l'application et répertorie les paramètres pris en charge et leur description.

«Informations d'instruction spécifiques à l'application», à la page 27

L'adaptateur met à jour les tables de base de données à l'aide de requêtes SQL ou de procédures stockées (groupes d'instructions SQL), comme indiqué dans les objets métier. Cette section décrit les procédures stockées et les éléments d'une définition de procédure stockée. Elle inclut un exemple de définition de procédure stockée.

Définition des propriétés de connexion

Une fois que vous avez créé le projet d'adaptateur, vous devez initialiser l'assistant ESD (Enterprise Service Discovery) pour l'adaptateur pour JDBC et définir les valeurs des propriétés de connexion de votre instance de base de données.

1. Initialisez ESD (Initialize Service Discovery)

Dans WebSphere Integration Developer, accédez à la perspective **Intégration métier**. Cliquez avec le bouton droit sur la page **Intégration métier** contenant le projet de connecteur JDBC, qui est sélectionné. Dans le menu en incrustation, sélectionnez **Nouveau > Enterprise Service Discovery**.

Dans la fenêtre Sélection d'un adaptateur de ressources de service d'entreprise, sélectionnez l'option de votre adaptateur, puis cliquez sur **Suivant**. Si vous avez déjà importé le fichier RAR, vous pouvez effectuer l'importation en cliquant sur le bouton **Importer l'adaptateur de ressources** de cette fenêtre.

2. Définissez les valeurs des propriétés de connexion

Vous devez définir les valeurs des propriétés de connexion à la fonction de recherche des métadonnées permettant de se connecter à l'instance EIS cible pour rechercher et créer la description de service. Pour activer le traitement des données de script bidirectionnel, vous devez activer la conversion bidirectionnelle et définir les valeurs des propriétés bidirectionnelles.

Dans la fenêtre Configuration des paramètres de l'agent de reconnaissance, entrez les valeurs des propriétés de configuration des connexions. Pour plus de détails sur ces propriétés, voir «Propriétés de connexion à la fonction de recherche des métadonnées» dans la section «Référence».

Pour activer la fonctionnalité bidirectionnelle, cochez la case en regard de **BiDi Transformation**. Définissez ensuite les valeurs des propriétés bidirectionnelles. Pour plus de détails sur ces propriétés, voir «Propriétés de connexion bidirectionnelles» dans la section «Référence».



Fenêtre Configuration des paramètres de l'agent de reconnaissance

3. Sélectionnez les options de journalisation

Si vous cliquez sur **Afficher les paramètres avancés**, les **options de journalisation** apparaissent. Les **options de journalisation** ne permettent de configurer les fonctions de journalisation et de trace que pour le processus ESD (Enterprise Service Discovery). Toutefois, les niveaux de journalisation et de trace sont les mêmes que pour l'adaptateur. Pour plus de détails sur les niveaux de journalisation et de trace de l'adaptateur, voir «Activation de la journalisation» et «Activation de la fonction de trace».

Sous **Options de journalisation**, entrez l'emplacement de sortie du fichier journal ou recherchez-le. Sélectionnez les niveaux de journalisation et de trace d'ESD (Enterprise Service Discovery). Cliquez sur **Suivant**.

Le service de reconnaissance utilise les propriétés de connexion pour préparer une arborescence des métadonnées qui est affichée pour la sélection et l'exploration des objets.

Tâches associées

«Activation de la journalisation», à la page 57

WebSphere Adapter pour JDBC conserve un fichier journal que vous pouvez afficher pour déterminer le statut du traitement des événements. Tous les événements et erreurs liés à l'adaptateur sont suivis par le fichier journal ; chaque événement est spécifié avec la date et l'heure. L'adaptateur consignait un message d'erreur chaque fois qu'il rencontre une erreur ou un avertissement, il est recommandé de commencer par le consulter pour rechercher les éventuels incidents.

«Activation de la fonction de trace», à la page 59

La fonction de trace détermine le niveau des erreurs ou des avertissements

consignés dans le fichier journal de l'adaptateur. Vous pouvez effectuer la trace des messages relatifs au traitement de l'adaptateur en définissant un niveau de suivi.

Référence associée

«Propriétés de connexion à la fonction de recherche des métadonnées», à la page 86

Le processus ESD (Enterprise Service Discovery) requiert ces propriétés pour se connecter au système EIS (Enterprise Information System) afin de rechercher et créer la description de service.

«Propriétés de connexion bidirectionnelles», à la page 87

Ces propriétés permettent à l'assistant ESD (Enterprise Service Discovery) d'appliquer la conversion bidirectionnelle appropriée aux données transmises au système EIS (Enterprise Information System).

Recherche d'objets de base de données

Après avoir configuré les propriétés de connexion, vous pouvez exécuter une requête sur les objets de base de données. Vous pouvez explorer l'arborescence des métadonnées pour comprendre la structure des objets du système EIS (Enterprise Information System) et sélectionner les objets requis pour la description de service.

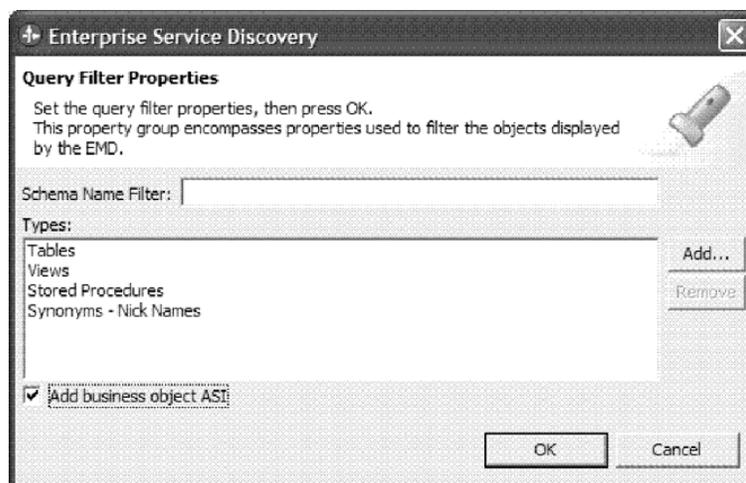
Avant d'exécuter la requête, vous pouvez spécifier les propriétés de filtre pour restreindre la liste des schémas, des noeuds ou des objets affichés dans l'arborescence.

1. Spécifiez les propriétés de filtre

Dans la fenêtre Recherche et reconnaissance des services d'entreprise, cliquez sur le bouton **Editer la requête**. Dans la fenêtre en incrustation Propriétés du filtre de requête, entrez votre texte dans la zone de propriété **Filtre Nom de schéma**. Les schémas commençant par la chaîne spécifiée sont affichés. Sélectionnez les schémas que vous souhaitez utiliser.

La zone de propriété **Types** répertorie les entrées : tables, vues, procédures stockées et synonymes/noms d'usage. Vous pouvez ajouter des noeuds à cette liste ou en supprimer.

Dans la fenêtre Propriétés du filtre de requête, vous pouvez sélectionner l'option **Ajouter ASI de l'objet métier**. Chaque fois que vous ajoutez un objet lors de l'exécution de la requête des métadonnées à l'étape 2, la fenêtre Paramètres de configuration pour (*nom de l'objet*) apparaît alors pour que vous puissiez spécifier les informations ASI.

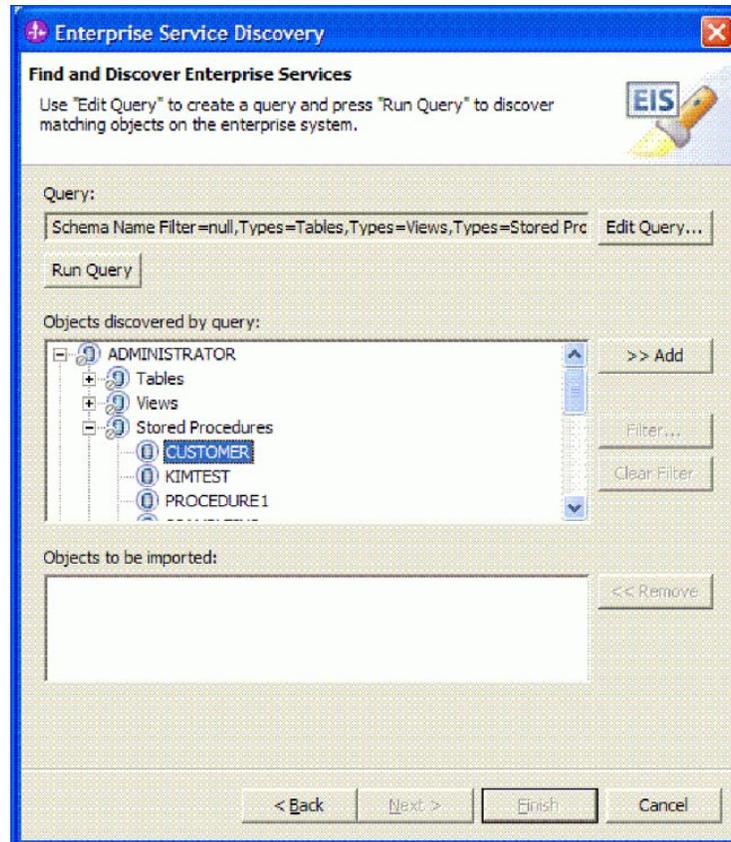


Fenêtre Propriétés du filtre de requête

2. Exécutez la requête des métadonnées

a. Affichez les objets identifiés par le biais d'une requête

Dans la fenêtre Recherche et reconnaissance des services d'entreprise, cliquez sur **Exécuter la requête**. Les objets sont affichés dans la sous-fenêtre supérieure.



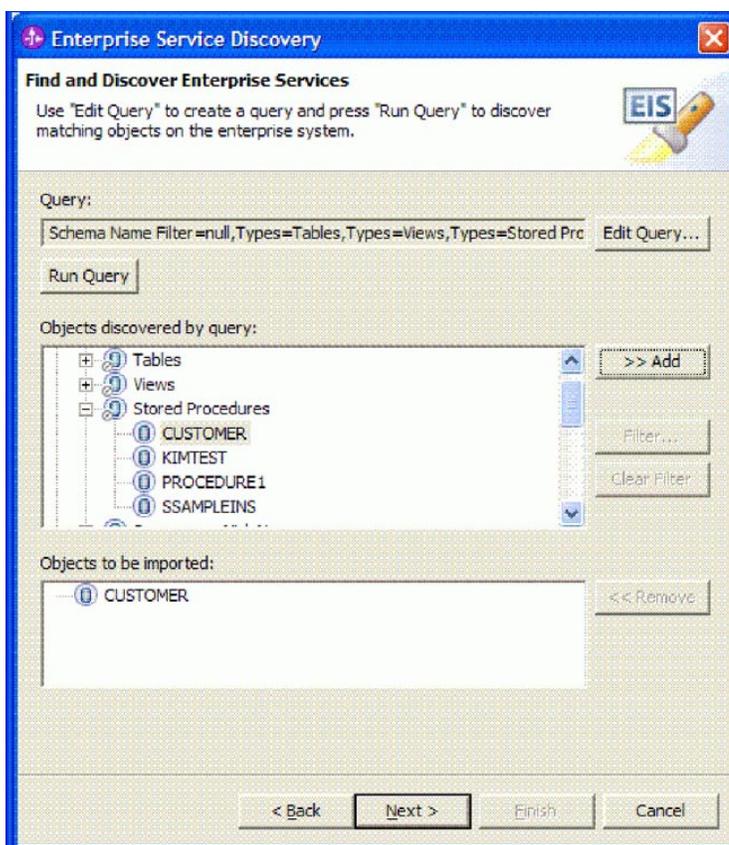
Fenêtre Recherche et reconnaissance des services d'entreprise

b. Filtrage d'objets

Vous pouvez filtrer les objets de la même manière que vous avez filtré les schémas. Sélectionnez un noeud de tables, de vues, de procédures stockées ou de synonymes/noms d'usage. Cliquez sur **Filtre**. L'assistant (Enterprise Service Discovery) recherche un **filtre de noms d'objet** pour filtrer la liste des objets de base de données à afficher pour ce noeud. Entrez du texte ; seuls les objets de base de données commençant par la chaîne spécifiée sont affichés.

c. Sélectionnez les objets à importer

Sélectionnez un objet et cliquez sur le bouton **Ajouter** pour sélectionner les objets à importer. Les objets sélectionnés apparaissent dans la sous-fenêtre inférieure. Pour supprimer un objet sélectionné, mettez-le en évidence et cliquez sur **Supprimer**.



L'objet Customer a été sélectionné pour l'importation

d. Ajoutez les informations ASI des objets métier

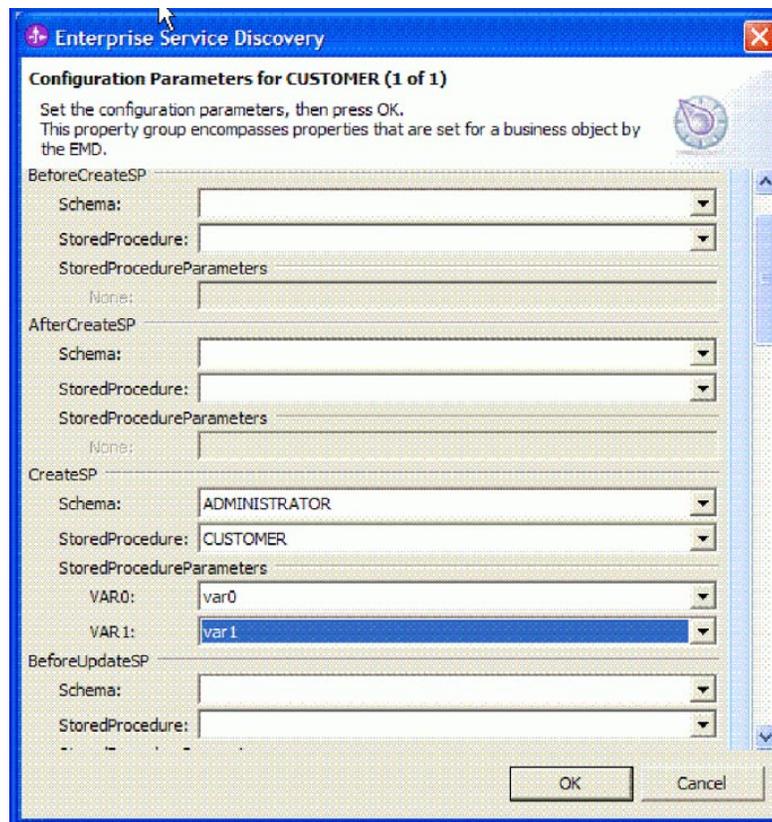
Si vous avez coché l'option **Ajouter ASI de l'objet métier** dans la fenêtre Propriétés du filtre de requête, la fenêtre Paramètres de configuration pour (*nom de l'objet*) apparaît chaque fois que vous ajoutez un objet, afin que vous puissiez spécifier les informations ASI. Cette fenêtre est illustrée ci-après avec les paramètres ASI des instructions de l'objet **CUSTOMER**.

Sélectionnez l'instruction à associer à la procédure stockée. Si par exemple la procédure stockée crée un enregistrement dans une table, ajoutez les détails correspondants au paramètre de configuration **CreateSP**. Entrez les valeurs correspondant respectivement au nom des paramètres **Schema**, **StoredProcedure** et **StoredProcedureParameters**.

Cliquez sur **OK** une fois que vous avez terminé d'entrer les informations ASI.

Si vous sélectionnez plusieurs objets à ajouter, la fenêtre du premier objet apparaît. Une fois que vous avez entré les informations ASI et cliqué sur **OK**, la fenêtre de l'objet suivant apparaît et ainsi de suite. Pour plus de détails sur les informations ASI au niveau objet, des instructions et des attributs, voir «Informations spécifiques à l'application».

Cliquez sur **Suivant**.



Paramètres de configuration de la fenêtre CUSTOMER

Tâches associées

«Déploiement et configuration du scénario 2», à la page 69

Dans le scénario 2, vous définissez les propriétés de connexion de l'adaptateur et générez des objets métier. Vous exportez votre projet dans un fichier EAR (Enterprise Application Archive), déployez le projet sur le serveur d'applications et réinitialisez les propriétés de configuration.

Référence associée

«Informations spécifiques à l'application», à la page 25

Les informations spécifiques à l'application dans les définitions d'objet métier fournissent à l'adaptateur les instructions dépendantes de l'application concernant le traitement des objets métier. L'adaptateur analyse les informations spécifiques à l'application depuis les attributs ou l'instruction d'un objet métier ou depuis l'objet métier lui-même pour générer des requêtes pour les opérations de création, de mise à jour, d'extraction ou de suppression.

Définition des propriétés de sélection

Une fois que vous avez sélectionné les objets de base de données, vous devez spécifier les valeurs des propriétés de sélection pour les fichiers d'importation et d'exportation.

Pour plus de détails sur les propriétés de sélection, reportez-vous à la section «Référence».

1. Spécifiez NameSpace

La propriété NameSpace correspond par défaut à l'espace de nom par défaut de tous les objets métier. Cette valeur par défaut apparaît dans la fenêtre Configuration des objets. Il s'agit de l'espace de nom du schéma de métadonnées, JDBCASI.xsd.

L'espace de nom précède le nom de l'objet métier pour que les schémas d'objet métier restent logiquement séparés.

2. Sélectionnez Service Type.

Sélectionnez **Inbound** ou **Outbound** comme type de service.

3. Sélectionnez Operations

Dans la fenêtre Configuration des objets, la zone **Operations** répertorie les opérations prises en charge par le type de service sélectionné. Si vous souhaitez ajouter certaines opérations à la liste, cliquez sur le bouton **Ajouter**. Par exemple, vous pouvez supprimer une opération, puis décider de l'inclure. Dans la fenêtre Ajouter, sélectionnez des opérations dans la liste et cliquez sur **OK**. Une fois que vous avez terminé, cliquez sur **Suivant**.

Les opérations spécifiées sont définies pour l'ensemble des objets métier générés.

4. Définissez MaxRecords

Entrez le nombre maximal d'enregistrements à extraire pour une opération RetrieveAll.

5. Spécifiez BOLocation

Entrez le chemin d'accès à l'emplacement dans lequel les fichiers .xsd générés doivent être stockés.

Tâches associées

«Propriétés de sélection», à la page 89

Une fois que vous avez sélectionné les objets de base de données, vous devez définir les valeurs des propriétés de sélection.

«Déploiement et configuration du scénario 2», à la page 69

Dans le scénario 2, vous définissez les propriétés de connexion de l'adaptateur et générez des objets métier. Vous exportez votre projet dans un fichier EAR (Enterprise Application Archive), déployez le projet sur le serveur d'applications et réinitialisez les propriétés de configuration.

Sauvegarde du projet d'adaptateur

Une fois que vous avez spécifié les propriétés de sélection, configurez les propriétés utilisées par l'adaptateur pour configurer un canal de communication vers une application de base de données spécifique. Il s'agit des propriétés de l'adaptateur de ressources, de la fabrique de connexions J2C, de la spécification d'activation J2C, ainsi que des propriétés bidirectionnelles. Vous devez également créer un module Business Integration dans lequel tous les artefacts et les valeurs des propriétés seront sauvegardés.

1. Spécifiez le nom du nouveau module

Dans la fenêtre Génération des artefacts illustrée ci-après, si aucun nom de module n'apparaît dans la zone **Module**, vous devez cliquer sur **Nouveau**.

Dans la fenêtre en incrustation Nouveau module, entrez le nom du module et cliquez sur **Terminer**.

2. Spécifiez le dossier de la description de service

Dans la fenêtre Génération des artefacts, spécifiez ou recherchez un dossier du nouveau module dans lequel la description de service sera sauvegardée.

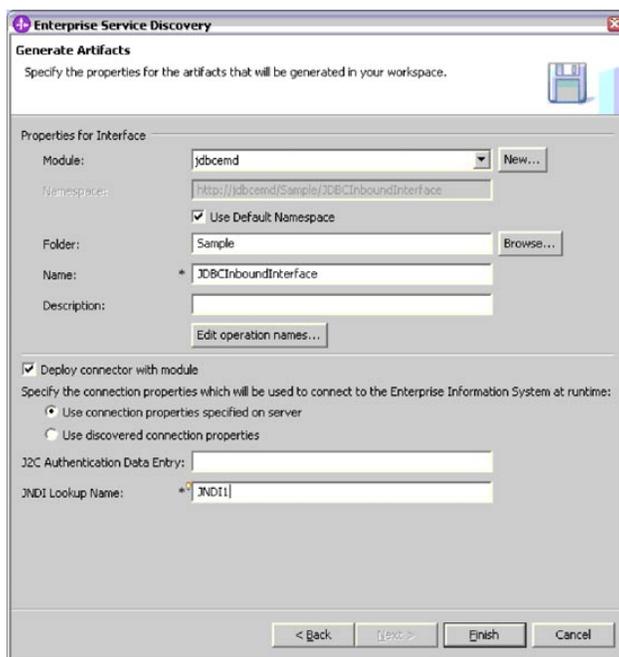
Suivant le type de service sélectionné, le nom de fichier d'importation ou d'exportation apparaît dans la zone **Nom**. Vous pouvez ajouter un commentaire dans la zone **Description**.

Si vous cliquez sur **Modifier les opérations**, une fenêtre en incrustation apparaît avec les noms par défaut de toutes les opérations des objets métier que vous créez. Les noms par défaut sont constitués du nom de l'opération, suivi du nom de l'objet métier.

Remarque : La case **Déployer le connecteur avec le module** doit être cochée pour que le fichier RAR du projet de connecteur soit inclus dans le fichier EAR que vous déployez sur le serveur d'applications.

3. Définissez les valeurs des propriétés de configuration

L'option **Utiliser les propriétés de connexion indiquées sur le serveur** peut être utilisée ultérieurement, une fois que vous avez déployé le projet, chaque fois que vous souhaitez créer des objets métier. Elle indique que vous souhaitez utiliser les propriétés qui se trouvent déjà sur le serveur d'applications. La zone **Saisie des données d'authentification J2C** permet de spécifier un alias d'authentification, si un alias est déjà configuré sur le serveur d'applications. Le **nom JNDI à rechercher** permet de désigner la fabrique de connexions qui sera utilisée sur le serveur d'applications.



Fenêtre Génération des artefacts

Pour définir les valeurs des propriétés, sélectionnez **Utiliser les propriétés reconnues de connexion**.

Pour le traitement des événements entrants, seules les zones des propriétés de la spécification d'activation J2C et de l'adaptateur de ressources apparaissent. Pour le traitement des événements sortants, seules les zones des propriétés de la fabrique de connexions J2C et de l'adaptateur de ressources apparaissent. Si vous avez activé la conversion bidirectionnelle, les zones des propriétés bidirectionnelles apparaissent pour le traitement des événements entrants et sortants. Les zones des propriétés requises sont marquées d'un astérisque. Pour plus de détails sur ces propriétés de configuration, reportez-vous à la section «Référence».

Important : Si vous spécifiez les propriétés de la spécification d'activation J2C à ce stade à l'aide de l'assistant ESD (Enterprise Service Discovery), les paramètres des propriétés ne peuvent pas être modifiés ultérieurement. Une fois que vous avez déployé le projet comme nouvelle application sur le serveur d'applications, vous ne pouvez plus mettre à jour les propriétés de la spécification d'activation J2C à l'aide de la console d'administration de WebSphere Process Server, bien que vous puissiez modifier d'autres propriétés de configuration à l'aide de cette dernière. Si vous essayez de mettre à

jour les propriétés ultérieurement, l'adaptateur ne reconnaîtra pas les valeurs mises à jour. Pour définir les propriétés de la spécification d'activation J2C après avoir installé le projet sur le serveur d'applications, ne définissez pas ces valeurs maintenant.

Référence associée

«Propriétés de configuration», à la page 75

Ce groupe de propriétés contient les attributs utilisés par l'adaptateur pour configurer un canal de communication vers une application de base de données spécifique.

Génération de liaisons de référence

Les liaisons de référence permettent aux autres composants SCA (Service Component Architecture) de WebSphere Business Integration d'accéder à l'adaptateur. Vous créez une référence à l'adaptateur depuis le module du projet afin de lier l'adaptateur à d'autres processus de serveur.

La génération de liaisons de référence n'est requise qu'en environnement de test. Elle n'est pas nécessaire lors du déploiement de l'adaptateur en environnement de production.

1. Créez un composant SCA

Accédez à la perspective **Intégration métier** de WebSphere Integration Developer. Dans la page **Intégration métier**, cliquez avec le bouton droit sur le module **JDBCEND** et sélectionnez **Ouvrir avec > Editeur d'assemblage**.

La fenêtre Diagramme d'assemblage apparaît avec le composant d'importation du module dans la vue. Pour créer un composant, cliquez sur la deuxième icône (Importer) du cadre de gauche (vertical) de la fenêtre Diagramme d'assemblage. Un nouveau menu d'icônes apparaît.

Cliquez sur l'icône située en bas du nouveau menu d'icônes (son info-bulle indique **Références autonomes**). Le curseur se transforme en icône de positionnement.

Cliquez sur la palette pour ajouter le nouveau composant dans la fenêtre du diagramme d'assemblage.

2. Créez une référence autonome

Cliquez sur le composant d'importation du module et faites-le glisser sur le nouveau composant. Une connexion est représentée entre le composant d'importation et le nouveau composant et la fenêtre Ajouter une connexion est affichée.

Dans la fenêtre Ajouter une connexion, cliquez sur **OK**. Une autre fenêtre Ajouter une connexion apparaît et vous demande si vous souhaitez convertir l'interface WSDL en langage Java. Cliquez sur **Non**.

Le nouveau composant Référence autonome est affiché dans la fenêtre **Diagramme d'assemblage** avec une "connexion" qui la relie au composant d'importation du module.

Cliquez sur **Fichier > Enregistrer** pour sauvegarder le diagramme d'assemblage. Pour plus d'informations, reportez-vous au guide de l'utilisateur de WebSphere Integration Developer, à l'adresse suivante : <http://www.ibm.com/software/integration/wid>.

Déploiement du projet d'adaptateur

Votre fichier de projet correspond à un projet de connecteur J2EE Connector dans l'espace de travail de WebSphere. Vous devez l'exporter dans votre système de fichiers local sous forme de fichier EAR (Enterprise Application Archive). Vous devez ensuite fournir les informations d'authentification pour la sécurité du connecteur JCA avant de pouvoir télécharger et installer le fichier EAR du projet sur le serveur d'applications.

1. Exporter le projet vers un fichier EAR

Dans WebSphere Integration Developer, vous devez exporter votre projet dans un fichier EAR. Dans la sous-fenêtre Exportation de la fenêtre Sélection, sélectionnez **Fichier EAR** dans la liste et cliquez sur **Suivant**.

Dans la fenêtre Exportation EAR, sélectionnez le module Intégration métier dans la zone du **Projet EAR**. Le nom du module, **JDBCemd**, possède maintenant le suffixe **App**. Entrez l'emplacement dans lequel le fichier EAR doit être créé ou recherchez-le. Cochez toutes les cases pour inclure tout ce que vous avez créé dans le fichier EAR. Cliquez sur **Terminer**.

2. Entrez les informations d'authentification

Ouvrez la console d'administration de WebSphere Process Server. Dans le panneau de gauche, cliquez sur **Sécurité > Sécurité globale**. Dans la sous-fenêtre Configuration de la fenêtre **Sécurité globale**, accédez à l'en-tête **Authentification**, dans la partie droite. Cliquez sur **Configuration JAAS**. Cliquez ensuite sur le lien **Données d'authentification J2C**.

Dans la fenêtre Sécurité globale > Entrées des données d'authentification de J2EE Connector Architecture (J2C), cliquez sur **Nouveau**. Entrez un nom d'alias dans la zone **Alias**. Entrez un ID utilisateur et un mot de passe permettant de se connecter à la base de données. Cliquez sur **OK**. Cliquez sur **Enregistrer** pour sauvegarder les informations d'authentification.

3. Installez le fichier EAR sur le serveur d'applications

A l'aide de la console d'administration, dans le panneau de gauche, cliquez sur **Applications > Installation d'une nouvelle application**. Cliquez sur **Parcourir**, sélectionnez le fichier EAR et cliquez sur **Suivant**.

Continuez de cliquer sur **Suivant** jusqu'à ce que vous atteigniez l'étape 7. **Mapper des références de ressources sur des ressources**. Sélectionnez l'alias d'authentification que vous avez créé précédemment, cochez la case **Utiliser la méthode par défaut**, puis cliquez sur **Appliquer**.

Dans les fenêtres qui suivent, cliquez sur **Suivant** ou sur **Continuer** jusqu'à ce que la sous-fenêtre Récapitulatif des options d'installation apparaisse. Cliquez sur **Terminer**. Le message Application *nom* installée avec succès apparaît.

Cliquez sur **Enregistrer dans la configuration principale** pour sauvegarder vos modifications. Cliquez ensuite sur **Enregistrer** dans la sous-fenêtre Applications d'entreprise.

Tâches associées

«Création d'un projet pour l'adaptateur», à la page 36

La première tâche du déploiement de l'adaptateur consiste à créer un projet de connecteur J2EE pour l'adaptateur. Le fichier RAR (Resource Adapter Archive) de l'adaptateur pour JDBC doit être importé dans WID (WebSphere Integration Developer). Ceci définit le projet dans votre espace de travail de WebSphere Integration Developer.

Configuration de l'adaptateur sur le serveur

Une fois que le fichier EAR (Enterprise Application Archive) du projet d'adaptateur a été installé sur le serveur d'applications, vous pouvez, si vous le souhaitez, reconfigurer les propriétés utilisées par l'adaptateur pour configurer un canal de communication vers une application de base de données spécifique. Vous pouvez ensuite démarrer l'application d'adaptateur configurée.

Normalement, les propriétés de configuration sont définies à l'aide de l'assistant ESD (Enterprise Service Discovery) lorsque vous créez votre projet d'adaptateur. Vous pouvez redéfinir ces propriétés à l'aide de la console d'administration de WebSphere Process Server ou vérifier si elles sont alimentées conformément aux valeurs que vous avez définies dans l'assistant ESD (Enterprise Service Discovery).

1. Ouvrez le projet d'adaptateur

Le nom de votre projet d'adaptateur apparaît dans la sous-fenêtre

Configuration de la console d'administration de WebSphere Process Server.

Sous l'en-tête **Articles liés**, sélectionnez **Modules de connecteurs**. Le nom du fichier RAR du projet est affiché. Cochez la case en regard du nom de fichier.

Sous **Propriétés supplémentaires**, sélectionnez **Adaptateur de ressources**.

2. Reconfigurez les propriétés

a. Editez les valeurs des propriétés des fabriques de connexions J2C

Dans la console d'administration, votre projet d'adaptateur est affiché sous **Propriétés générales > Nom**. Dans la partie droite de la fenêtre, sous **Propriétés supplémentaires**, cliquez sur **Fabriques de connexions J2C**.

Le nom du projet d'adaptateur est affiché avec le nom JNDI spécifié dans le projet EJB. Cochez la case en regard de la fabrique de connexions de votre adaptateur.

Les propriétés et les valeurs de la fabrique de connexions J2C apparaissent. Editez les valeurs des propriétés si besoin est. Pour plus de détails sur ces propriétés, voir les «propriétés des fabriques de connexions J2C.»

b. Définissez les valeurs des propriétés de la spécification d'activation J2C

Suivez l'étape 2a pour définir les valeurs des propriétés de la spécification d'activation J2C selon vos besoins, si vous ne les avez pas déjà définies.

Important : Si vous avez déjà spécifié les propriétés de la spécification d'activation J2C à l'aide de l'assistant ESD (Enterprise Service Discovery), vous ne pouvez pas en modifier les valeurs maintenant. Une fois que vous avez déployé le projet comme nouvelle application sur le serveur d'applications, vous ne pouvez plus mettre à jour les propriétés de la spécification d'activation J2C à l'aide de la console d'administration de WebSphere Process Server. Si vous essayez de mettre à jour les valeurs de ces propriétés, l'adaptateur ne les reconnaîtra pas. Si vous n'avez pas défini les propriétés de la spécification d'activation J2C avant d'installer le projet sur le serveur d'applications, vous pouvez les définir maintenant.

Pour définir ces propriétés, sous **Propriétés supplémentaires**, cliquez sur **Spécifications d'activation J2C**. Pour plus de détails sur ces propriétés, voir les «propriétés de la spécification d'activation J2C.»

c. Editez les valeurs des propriétés de l'adaptateur de ressources

Sous **Propriétés supplémentaires**, cliquez sur **Propriétés personnalisées**.

Une liste des propriétés de l'adaptateur de ressources et de leurs valeurs est

affichée. Editez-les si besoin est. Voir les propriétés et leur description dans «Propriétés de l'adaptateur de ressources».

Remarque : Dans la liste **Propriétés personnalisées**, au niveau de l'adaptateur, recherchez la propriété **DatabaseVendor**. Cette propriété est requise pour démarrer l'application. Spécifiez une valeur pour cette propriété si elle n'en a pas et sauvegardez-la.

3. Démarrez l'application de l'adaptateur

Dans le panneau de gauche de la console d'administration, cliquez sur **Applications > Applications d'entreprise**. Dans le panneau **Applications d'entreprise**, cochez la case en regard du nom de votre application, puis cliquez sur **Démarrer**. Un message indique que l'application a démarré. Si des incidents se produisent lorsque vous tentez de démarrer l'application, recherchez la description de ces erreurs dans le fichier journal de l'adaptateur.

Pour plus d'informations sur des messages spécifiques, voir Messages dans le centre de documentation d'IBM WebSphere Adapters. Pour plus de détails sur les fonctions de journalisation et de trace, voir «Identification et résolution des incidents».

Tâches associées

«Identification et résolution des incidents»

Vous pouvez contacter le service de support logiciel d'IBM pour signaler un incident. Pour cela, veuillez suivre les instructions de la présente section. Le statut de traitement d'un événement peut être déterminé en activant la fonction de journalisation de l'adaptateur. Vous pouvez définir le niveau de trace, afin de déterminer le niveau des erreurs ou avertissements consignés dans le fichier journal de l'adaptateur. Les instructions de la présente section indiquent comment activer l'infrastructure CEI (Common Event Infrastructure) pour votre adaptateur. Elle décrit également le mode d'accès aux données de grand système.

Référence associée

«Propriétés des fabriques de connexions J2C», à la page 79

Les propriétés des fabriques de connexions J2C permettent de configurer une instance de système EIS (Enterprise Information System) cible. Ces propriétés affectent le traitement des communications sortantes et correspondent à l'interface ManagedConnectionFactory de la spécification JCA (J2EE Connector Architecture).

«Propriétés de la spécification d'activation J2C», à la page 82

Les propriétés de la spécification d'activation J2C activent les noeuds finaux de message. Ces propriétés correspondent à l'interface ActivationSpec de la spécification JCA (J2EE Connector Architecture Specification).

«Propriétés de l'adaptateur de ressources», à la page 75

Ces propriétés de configuration sont définies au niveau de l'adaptateur de ressources.

Identification et résolution des incidents

Vous pouvez contacter le service de support logiciel d'IBM pour signaler un incident. Pour cela, veuillez suivre les instructions de la présente section. Le statut de traitement d'un événement peut être déterminé en activant la fonction de journalisation de l'adaptateur. Vous pouvez définir le niveau de trace, afin de déterminer le niveau des erreurs ou avertissements consignés dans le fichier journal de l'adaptateur. Les instructions de la présente section indiquent comment activer l'infrastructure CEI (Common Event Infrastructure) pour votre adaptateur. Elle décrit également le mode d'accès aux données de grand système.

Tâches associées

«Configuration de l'adaptateur sur le serveur», à la page 54

Une fois que le fichier EAR (Enterprise Application Archive) du projet d'adaptateur a été installé sur le serveur d'applications, vous pouvez, si vous le souhaitez, reconfigurer les propriétés utilisées par l'adaptateur pour configurer un canal de communication vers une application de base de données spécifique. Vous pouvez ensuite démarrer l'application d'adaptateur configurée.

Pour contacter le centre de support logiciel IBM

Le centre de support logiciel IBM vous aide à résoudre les problèmes liés aux défauts des produits.

Pour que vous puissiez faire appel au centre de support logiciel IBM, votre société doit disposer d'un contrat de maintenance logicielle IBM en cours de validité et vous devez être autorisé à soumettre des problèmes à IBM. Le type de contrat de maintenance logicielle requis dépend du type de produit dont vous disposez :

- Pour les produits logiciels distribués par IBM (y compris, mais non limités à, Tivoli, Lotus, et Rational, ainsi que DB2 et les produits WebSphere s'exécutant sous Windows ou UNIX), souscrivez à Passport Advantage de l'une des manières suivantes :
 - **En ligne** : Accédez à la page Web du programme Passport Advantage et cliquez sur How to Enroll.
 - **Par téléphone** : Pour connaître le numéro de téléphone pour votre pays, accédez à la page Contacts du manuel IBM Software Support Handbook, disponible sur le Web et cliquez sur votre zone géographique.
- Pour les produits logiciels IBM eServer (y compris, mais non limités à, DB2 et les produits WebSphere s'exécutant sous zSeries, pSeries, et iSeries), vous pouvez acquérir un contrat de maintenance logicielle en contactant directement un ingénieur commercial ou un partenaire commercial IBM. Pour plus d'informations sur le support disponible pour les logiciels eServer, accédez à la page Web IBM Technical Support Advantage.

Si vous avez un doute concernant le type de contrat de maintenance logicielle dont vous avez besoin, appelez le 1-800-IBMSERV (1-800-426-7378) aux Etats-Unis ou, pour les autres pays, accédez à la page Contacts du manuel IBM Software Support Handbook disponible sur le Web et cliquez sur le nom de votre zone géographique afin de connaître le numéro de téléphone de votre centre de support local.

Pour prendre contact avec le service de support logiciel IBM, procédez comme suit :

- Déterminez l'incidence professionnelle du problème.
 - Décrivez le problème et collectez toutes les informations relatives à votre système.
 - Soumettez le problème au centre de support logiciel IBM.
1. Déterminez l'incidence professionnelle du problème. Lorsque vous signalez un problème au support technique IBM, vous devez lui indiquer un niveau de gravité. Par conséquent, vous devez comprendre et évaluer l'incidence du problème signalé sur votre entreprise. Utilisez les critères suivants :

Gravité	Description
Gravité 1	Incidence critique sur l'entreprise : Vous ne pouvez pas utiliser le programme, ce qui a une incidence critique sur votre activité. Cette condition requiert une solution immédiate.

Gravité	Description
Gravité 2	Incidence significative sur votre entreprise : Le programme est utilisable, mais ses fonctionnalités sont très limitées.
Gravité 3	Incidence limitée sur votre entreprise : Le programme est utilisable, mais les fonctionnalités moins importantes (non essentielles pour votre activité) ne sont pas disponibles.
Gravité 4	Incidence minimale sur votre entreprise : Le problème a peu d'incidence sur votre activité ou une solution appropriée a été mise en oeuvre.

2. Décrivez le problème et collectez toutes les informations relatives à votre système. Lorsque vous expliquez un problème à IBM, soyez aussi précis que possible. Fournissez toutes les informations appropriées concernant votre système pour que les spécialistes du support logiciel IBM puissent vous aider à résoudre efficacement le problème. Pour gagner du temps, sachez répondre aux questions suivantes :
 - Quelles versions de logiciel exécutiez-vous lorsque le problème est survenu ?
 - Disposez-vous de fichiers journaux, de données de trace et de messages relatifs aux symptômes du problème ? Le support logiciel IBM va vraisemblablement vous poser ces questions.
 - Pouvez-vous recréer le problème ? Si oui, quelles opérations ont causé la défaillance ?
 - Des modifications ont-elles été apportées au système (par exemple, au matériel, au système d'exploitation, au logiciel de réseau) ?
 - Utilisez-vous actuellement une solution palliative pour ce problème ? Si oui, vous devrez l'expliquer lorsque vous signalerez le problème.
3. Soumettez le problème au centre de support logiciel IBM. Vous pouvez soumettre le problème selon une des méthodes suivantes :
 - **En ligne** : Accédez à la page Submit and track problems du site IBM Software Support. Saisissez les informations requises dans l'outil de soumission des problèmes approprié.
 - **Par téléphone** : Pour connaître le numéro de téléphone pour votre pays, accédez à la page Contacts du manuel IBM Software Support Handbook, disponible sur le Web et cliquez sur votre zone géographique.

Si le problème soumis concerne un défaut logiciel ou une documentation incomplète ou inexacte, le centre de support logiciel IBM crée un APAR (Authorized Program Analysis Report). Celui-ci décrit le problème en détail.

Dans la mesure du possible, le centre de support logiciel IBM vous fournira une solution palliative à mettre en oeuvre jusqu'à ce que l'APAR soit résolu et qu'un correctif soit proposé. IBM publie quotidiennement les APAR résolus dans les pages Web de support des produits IBM pour que les autres utilisateurs ayant rencontré le même problème puissent bénéficier des mêmes solutions.

Activation de la journalisation

WebSphere Adapter pour JDBC conserve un fichier journal que vous pouvez afficher pour déterminer le statut du traitement des événements. Tous les événements et erreurs liés à l'adaptateur sont suivis par le fichier journal ; chaque événement est spécifié avec la date et l'heure. L'adaptateur consignait un message d'erreur chaque fois qu'il rencontre une erreur ou un avertissement, il est recommandé de commencer par le consulter pour rechercher les éventuels incidents.

L'adaptateur communique avec la base de données via un pilote JDBC et les requêtes SQL ou les appels de procédure stockée peuvent générer un message d'exception SQL. Ces messages sont capturés et consignés et un message ResourceException est généré. Les messages sont renvoyés en fonction des erreurs qui se produisent lors du traitement des demandes ou des erreurs du magasin d'événements. Les erreurs du traitement des événements sont traitées par le gestionnaire d'événements.

Remarque : Lorsque vous créez et connectez simultanément une opération entrante et une opération sortante à la même instance d'adaptateur via la même propriété AdapterID, vous devez utiliser le même fichier journal, car un seul fichier est créé au moment du déploiement de l'adaptateur sur WebSphere Process Server. Si les opérations entrante et sortante sont liées à deux adaptateurs distincts et configurés avec une propriété AdapterID différente, un fichier journal séparé est créé pour chaque opération.

La journalisation de l'adaptateur est activée par l'intermédiaire de la console d'administration de WebSphere Process Server. Suivez les étapes ci-après pour activer la journalisation.

1. Démarrez les fonctions de journalisation et de trace
 Démarrez la console d'administration de WebSphere Process Server. Dans la console d'administration, sélectionnez **Résolution des incidents** → **Journaux et trace**.
2. Spécifiez le niveau de détail du journal
 Cliquez sur **Composant** pour spécifier un niveau de détail de journalisation pour chaque composant ou sur **Groupes** pour en spécifier un pour un groupe de composants prédéfini.
3. Sélectionnez le niveau de journalisation
 Le tableau "Niveaux de journalisation" décrit les différents niveaux de journalisation qui peuvent être définis via la console d'administration.

Remarque : Pour afficher les événements du journal qui se trouvent sous le niveau Détails, vous devez activer Diagnostics du service de trace. Les événements du journal qui se trouvent au niveau Détails ou au-dessus peuvent être visualisés dans le journal SystemOut, le journal IBM Service (s'il est activé), ou Diagnostics du service de trace (s'il est activé).

Niveaux de journalisation

Niveau	Indicateur	Description
Fatal	F	La tâche ne peut pas se poursuivre. Le composant ne peut pas fonctionner.
Grave	E	La tâche ne peut pas se poursuivre. Le composant peut toujours fonctionner. Inclut également les conditions indiquant une erreur fatale imminente, c'est-à-dire les situations qui indiquent que les ressources sont sur le point d'être épuisées.

Niveau	Indicateur	Description
Avertissement	W	Erreur potentielle ou imminente. Inclut également les conditions indiquant un échec progressif (par exemple, une perte de ressources potentielle).
Audit	A	Événement important affectant les ressources ou l'état du serveur.
Info	I	Informations générales décrivant la progression générale des tâches.
Configuration	C	Statut ou modification de la configuration.
Détails	D	Informations générales détaillant la progression des sous-tâches.

4. Sauvegardez les informations de journalisation

Cliquez sur **Appliquer** pour sauvegarder vos modifications.

Tâches associées

«Définition des propriétés de connexion», à la page 44

Une fois que vous avez créé le projet d'adaptateur, vous devez initialiser l'assistant ESD (Enterprise Service Discovery) pour l'adaptateur pour JDBC et définir les valeurs des propriétés de connexion de votre instance de base de données.

Activation de la fonction de trace

La fonction de trace détermine le niveau des erreurs ou des avertissements consignés dans le fichier journal de l'adaptateur. Vous pouvez effectuer la trace des messages relatifs au traitement de l'adaptateur en définissant un niveau de suivi.

Vous pouvez configurer les niveaux de trace par l'intermédiaire de la console d'administration de WebSphere Process Server. Suivez la procédure ci-après pour activer et définir les niveaux de trace.

1. Démarrez la fonction de trace

Démarrez la console d'administration de WebSphere Process Server. Sélectionnez **Résolution des incidents** → **Journaux et trace**.

2. Sélectionnez le niveau de trace

Le tableau "Niveaux de trace" décrit les différents niveaux de trace qui peuvent être définis via la console d'administration.

Niveaux de trace

Niveau	Indicateur	Description
Précis	1	Trace générale. Inclut les actions étendues effectuées par l'adaptateur, telles que l'établissement d'une connexion au système EIS (Enterprise Information System), la conversion d'un événement du système EIS en un objet métier (uniquement les valeurs de clé), le traitement d'un objet métier (uniquement les valeurs de clé).
Plus précis	2	Trace détaillée fournissant des informations plus précises sur la logique effectuée par l'adaptateur et notamment les divers appels de données spécifiques à l'application au système EIS (Enterprise Information System) et les éventuels paramètres et valeurs renvoyées.
Le plus précis	3	Niveau le plus détaillé ; il doit inclure les valeurs des entrées de méthode / les valeurs de sortie / les valeurs renvoyées. Des vidages d'objet métier complets doivent être inclus. A ce niveau, tous les détails requis pour déboguer les incidents doivent être fournis.

3. Sauvegardez les informations de trace

Cliquez sur **Appliquer** pour sauvegarder vos modifications.

Tâches associées

«Définition des propriétés de connexion», à la page 44

Une fois que vous avez créé le projet d'adaptateur, vous devez initialiser l'assistant ESD (Enterprise Service Discovery) pour l'adaptateur pour JDBC et définir les valeurs des propriétés de connexion de votre instance de base de données.

Activation de Common Event Infrastructure (CEI)

Cette rubrique décrit comment activer CEI (Common Event Infrastructure) pour l'adaptateur.

Vous devez publier le fichier de définitions d'événements des adaptateurs IBM WebSphere dans le catalogue de CEI avant de pouvoir définir ces définitions d'événements. Pour plus d'instructions sur la façon de procéder, voir la

documentation CEI sur le site Web de WebSphere Process Server à l'adresse <http://www.ibm.com/software/integration/wps>.

1. Démarrez la console d'administration WebSphere.
2. Accédez à **Résolution des incidents** → **Journalisation et trace** et sélectionnez le <nom de votre serveur>.
3. Vous verrez de nombreuses options pour les propriétés générales. Sélectionnez l'option de **modification du niveau de détail de la journalisation**, puis sélectionnez **com.ibm.j2ca.*** pour les composants JCA. Dans cette section, il existe un sous-composant pour chaque type d'adaptateur :
 - com.ibm.j2ca.flatfile.* (WebSphere Adapter for Flat Files)
 - com.ibm.j2ca.jdbc.* (WebSphere Adapter pour JDBC)
 - com.ibm.j2ca.peoplesoft.* (WebSphere Adapter pour PeopleSoft)
 - com.ibm.j2ca.sap.* (WebSphere Adapter pour SAP)
 - com.ibm.j2ca.siebel.* (WebSphere Adapter pour Siebel)
4. Sélectionnez le composant qui correspond à votre adaptateur. Chaque composant d'adaptateur a deux sous-composants, un pour la journalisation et un pour CEI. Ils se présentent comme suit :
 - *nom sous-composant.log.id adaptateur*
 - *nom sous-composant.cei.id adaptateur*

Par exemple, com.ibm.j2ca.siebel.cei.<AdapterID1>. Pour chaque instance d'un adaptateur déployé, le système indique un ID distinct.
5. Sélectionnez l'ID d'adaptateur CEI que vous voulez activer.
6. Dans le menu déroulant, vous avez le choix entre les options suivantes :
 - off : désactive CEI
 - fine : active CEI avec un contenu d'événement défini sur Vide
 - finer : active CEI avec un contenu d'événement défini sur Sommaire
 - finest : active CEI avec un contenu d'événement défini sur Complet
 - all : même action que finest

Pour plus d'informations sur la signification des différents niveaux de contenu d'événement (Vide, Sommaire et Complet) et pour plus d'informations sur l'utilisation du modèle Common Base Event model et de Common Event Infrastructure, voir la documentation du Centre de documentation de WebSphere Process Server à l'adresse <http://www.ibm.com/software/integration/wps>

Accès aux données de l'ordinateur principal

L'adaptateur JDBC prend en charge l'accès aux données d'ordinateur principal via le composant IBM WebSphere Information Integrator Classic Federation pour z/OS. Ce produit comprend des applications Web et distribuées ayant des connectivités d'accès en lecture/écriture aux bases de données de grand système.

Ce produit offre un accès SQL à performances élevées et permet de fédérer les sources de données de grand système. Vous pouvez sélectionner sur le bureau les outils et applications dont vous avez besoin pour accéder en toute transparence aux informations stratégiques situées sur un ordinateur principal.

Si vous rencontrez des difficultés lors de l'utilisation de l'adaptateur pour JDBC avec IBM WebSphere Information Integrator Classic Federation sur la plateforme IBM z/OS, reportez-vous aux notes techniques décrivant la configuration requise

dans les pages Web de support des logiciels IBM, à l'adresse suivante :
<http://www-306.ibm.com/software/integration/wbiadapters/support>.

Utilisation des exemples de fichier et de l'exemple d'application

Des exemples des artefacts suivants sont fournis pour le processus ESD (Enterprise Service Discovery) : des fichiers d'entrée et de sortie de SCA (Service Component Architecture) et un fichier WSDL (Web Services Description Language) correspondant. En outre, un exemple d'application est fourni pour vous aider à vous familiariser avec l'utilisation de l'adaptateur lors de l'établissement de communications entre les applications J2EE et les systèmes EIS (Enterprise Information System).

Exemples de fichier d'importation, d'exportation et WSDL

Des exemples de fichier d'importation et d'exportation SCA (Service Component Architecture), ainsi qu'un fichier WSDL (Web Services Description Language) correspondant sont fournis. Ces artefacts sont générés lors du processus ESD (Enterprise Service Discovery).

Exemple de fichier d'exportation (entrant)

```
<?xml version="1.0" encoding="UTF-8"?>
<scdl:Export xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:_="http://JDBCEMD/inbound"
xmlns:eis="http://www.ibm.com/xmlns/prod/websphere/scdl/eis/6.0.0"
xmlns:scdl="http://www.ibm.com/xmlns/prod/websphere/scdl/6.0.0"
xmlns:wSDL="http://www.ibm.com/xmlns/prod/websphere/scdl/wSDL/6.0.0"
name="inbound/JDBCInboundInterface">
  <interfaces>
    <interface xsi:type="wSDL:WSDLPortType" portType="_:JDBCInboundInterface"/>
  </interfaces>
  <esbBinding xsi:type="eis:EISExportBinding"
dataBindingType="com.ibm.j2ca.extension.emd.runtime.WBIDataBindingImpl">
    <resourceAdapter name="JDBCEMDApp.IBM JDBC Adapter"
type="com.ibm.j2ca.jdbc.JDBCResourceAdapter">
      <properties/>
      </resourceAdapter>
      <connection type="com.ibm.j2ca.jdbc.inbound.JDBCActivationSpec"
selectorType="com.ibm.j2ca.extension.emd.runtime.WBIFunctionSelectorImpl">
        <properties>
          <BONamespace>http://www.ibm.com/xmlns/prod/websphere/j2ca/jdbc</BONamespace>
          <jdbcDriverClass>COM.ibm.db2.jdbc.app.DB2Driver</jdbcDriverClass>
          <databaseURL>jdbc:db2:somedb<databaseURL>
          <password>abcdefg</password>
          <userName>db2admin</userName>
        </properties>
      </connection>
      <methodBinding method="createDb2adminCustomer"
nativeMethod="emitCreateAfterImageDb2adminCustomer"/>
      <methodBinding method="updateDb2adminCustomer"
nativeMethod="emitUpdateAfterImageDb2adminCustomer"/>
      <methodBinding method="deleteDb2adminCustomer"
nativeMethod="emitDeleteAfterImageDb2adminCustomer"/>
    </esbBinding>
  </scdl:Export>
```

Exemple de description de service d'importation (sortante)

```
<?xml version="1.0" encoding="UTF-8"?>
<scdl:Import xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:_="http://JDBCEMD/outbound"
xmlns:eis="http://www.ibm.com/xmlns/prod/websphere/scdl/eis/6.0.0"
xmlns:scdl="http://www.ibm.com/xmlns/prod/websphere/scdl/6.0.0"
xmlns:wSDL="http://www.ibm.com/xmlns/prod/websphere/scdl/wSDL/6.0.0"
name="outbound/JDBCOutboundInterface">
  <interfaces>
    <interface xsi:type="wSDL:WSDLPortType" portType="_:JDBCOutboundInterface"/>
  </interfaces>
  <esbBinding xsi:type="eis:EISImportBinding"
dataBindingType="com.ibm.j2ca.extension.emd.runtime.WBIDataBindingImpl">
```

```

    <resourceAdapter name="JDBCCEMApp.IBM JDBC Adapter"
type="com.ibm.j2ca.jdbc.JDBCResourceAdapter">
    <properties/>
</resourceAdapter>
    <connection type="com.ibm.j2ca.jdbc.JDBCManagedConnectionFactory"
interactionType="com.ibm.j2ca.jdbc.JDBCInteractionSpec">
    <properties>
        <databaseURL>jdbc:db2:somedb</databaseURL>
        <jdbcDriverClass>COM.ibm.db2.jdbc.app.DB2Driver</jdbcDriverClass>
        <password>abcdefg</password>
        <userName>db2admin</userName>
    </properties>
</connection>
    <methodBinding method="createDb2adminCustomer">
    <interaction>
        <properties>
            <functionName>Create</functionName>
        </properties>
    </interaction>
</methodBinding>
    <methodBinding method="updateDb2adminCustomer">
    <interaction>
        <properties>
            <functionName>Update</functionName>
        </properties>
    </interaction>
</methodBinding>
    <methodBinding method="deleteDb2adminCustomer">
    <interaction>
        <properties>
            <functionName>Delete</functionName>
        </properties>
    </interaction>
</methodBinding>
    <methodBinding method="retrieveDb2adminCustomer">
    <interaction>
        <properties>
            <functionName>Retrieve</functionName>
        </properties>
    </interaction>
</methodBinding>
    <methodBinding method="retrieveallDb2adminCustomer">
    <interaction>
        <properties>
            <functionName>RetrieveAll</functionName>
        </properties>
    </interaction>
</methodBinding>
    <methodBinding method="applychangesDb2adminCustomer">
    <interaction>
        <properties>
            <functionName>ApplyChanges</functionName>
        </properties>
    </interaction>
</methodBinding>
</esbBinding>
</scdl:Import>

```

Exemple de fichier WSDL

```

<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions name="CUSTOMER"
    targetNamespace="http://test/j2c/jdbc/customer"
    xmlns:tns="http://test/j2c/jdbc/customer"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:datans="http://test/j2c/jdbc/customer">
    <wsdl:types>

```

```

<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:import namespace="http://test/j2c/jdbc/customer"
    schemaLocation="CUSTOMER.xsd">
  </xsd:import>
</xsd:schema>
</wsdl:types>
<wsdl:message name="CUSTOMERRequest">
  <wsdl:part name="request" element="datans:CUSTOMER"></wsdl:part>
</wsdl:message>
<wsdl:portType name="Customer">
  <wsdl:operation name="updateCustomer">
    <wsdl:input message="tns:CustomerRequest"></wsdl:input>
    <wsdl:output message="tns:CustomerRequest"></wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="createCustomer">
    <wsdl:input message="tns:CustomerRequest"></wsdl:input>
    <wsdl:output message="tns:CustomerRequest"></wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="retrieveCustomer">
    <wsdl:input message="tns:CustomerRequest"></wsdl:input>
    <wsdl:output message="tns:CustomerRequest"></wsdl:output>
  </wsdl:operation>
</wsdl:portType>
</wsdl:definitions>

```

Exemple : Mise à jour d'une application de base de données

Des exemples de fichier sont fournis avec IBM WebSphere Adapter pour JDBC pour que vous puissiez vous entraîner à mettre à jour des tables spécifiques dans une application de base de données. Deux scénarios pas-à-pas sont fournis, l'un concernant l'intégrateur d'applications, l'autre l'intégrateur de données. Suivant votre rôle, vous pouvez vous entraîner à générer des objets métier, à déployer l'adaptateur et à configurer ce dernier pour communiquer entre les applications J2EE et les systèmes EIS (Enterprise Information System).

Exemples de scénario d'application

Scénario	Description	Public visé
Scénario 1	<ul style="list-style-type: none"> Fournit des artefacts déjà générés et illustre la manière dont l'adaptateur traite les objets métier. Ce scénario ne nécessite pas l'utilisation de l'assistant ESD (Enterprise Service Discovery) pour générer des artefacts. Destiné aux personnes chargées d'assembler des composants d'application en une solution et de préparer cette dernière pour la tester et la déployer. 	Intégrateur d'applications
Scénario 2	<ul style="list-style-type: none"> Illustre la manière d'utiliser l'assistant ESD (Enterprise Service Discovery) pour rechercher des composants de base de données et développer les objets métier traités par l'adaptateur. Destiné aux personnes qui possèdent les mêmes responsabilités que l'intégrateur d'applications, mais qui sont également chargées d'activer l'accès à une plage de sources de données pour les développeurs d'applications. 	Intégrateur de données

Pour plus de détails sur les rôles de l'intégrateur d'applications et de l'intégrateur de données, voir la rubrique «Public visé» de la «Présentation du produit.»

Concepts associés

«Public visé», à la page 2

Cette rubrique définit les utilisateurs des produits WebSphere Adapter et détaille les compétences qu'ils doivent avoir.

Obtention du fichier du projet pour le scénario 1

Pour le scénario 1, l'exemple inclut un ensemble complet de fichiers pour que vous n'ayez pas à les générer. En environnement de travail réel, une personne doit générer les artefacts et configurer l'adaptateur à l'aide de l'assistant (Enterprise Service Discovery).

Avant d'utiliser les scénarios fournis, vous devez installer l'adaptateur et extraire le package des exemples du dossier «Sample.»

Pour plus d'informations sur l'installation de l'adaptateur, reportez-vous à la rubrique Installation d'adaptateurs IBM WebSphere.

Procurez-vous le fichier du projet

Accédez au dossier «jdbc/samples/Apps» et recherchez le fichier JDBCApp.ear.

Le fichier EAR (Enterprise Application Archive) de ce projet contient les fichiers suivants :

- Une instance configurée de l'adaptateur, déployée par défaut sur le serveur d'applications hôte local : CWYBC_JDBC.rar
- Un module SCA (Service Component Architecture) avec divers artefacts SCA :
 - JDBCInboundInterface.export
 - JDBCInboundInterface.wsdl
 - JDBCOutboundInterface.import
 - JDBCOutboundInterface.wsdl
- Un fichier WSDL : CustomerInterface.wsdl
- Des objets métier :
 - InboundRtasserCustomer.xsd
 - InboundRtasserAddress.xsd
 - InboundRtasserCustInfo.xsd
 - OutboundRtasserCustomer.xsd
 - OutboundRtasserAddress.xsd
 - OutboundRtasserCustInfo.xsd
 - InboundRtasserCustomerBG.xsd
 - InboundRtasserAddressBG.xsd
 - InboundRtasserCustInfoBG.xsd
 - OutboundRtasserCustomerBG.xsd
 - OutboundRtasserAddressBG.xsd
 - OutboundRtasserCustInfoBG.xsd
 - InboundRtasserCustomerContainer.xsd
 - InboundRtasserAddressContainer.xsd

- InboundRtasserCustInfoContainer.xsd
- OutboundRtasserCustomerContainer.xsd
- OutboundRtasserAddressContainer.xsd
- OutboundRtasserCustInfoContainer.xsd
- Des scripts SQL permettant de créer des tables de base de données :
 - CreateCustomerTable.sql
 - CreateAddressTable.sql
 - CreateCustInfoTable.sql

Vous pouvez ensuite passer à la tâche de déploiement et de configuration.

Obtention du fichier du projet pour le scénario 2

Pour le scénario 2, l'exemple inclut un ensemble minimal de fichiers pour que vous puissiez vous entraîner à générer un ensemble complet d'artefacts à l'aide de l'assistant ESD (Enterprise Service Discovery).

Avant d'utiliser l'un des scénarios fournis, vous devez installer l'adaptateur et extraire le package des exemples du dossier «Sample.»

Pour plus d'informations sur l'installation de l'adaptateur, reportez-vous à la rubrique Installation d'adaptateurs IBM WebSphere.

Procurez-vous le fichier du projet

Accédez au dossier «jdbc/samples/Apps» et recherchez le fichier JDBCApp.ear.

Le fichier EAR (Enterprise Application Archive) de ce projet contient les fichiers répertoriés ci-après. Ces fichiers sont des exemples des artefacts que vous créerez dans ce scénario lorsque vous générerez des artefacts et configurerez l'adaptateur pour le déploiement et l'utilisation à l'aide de l'assistant ESD (Enterprise Service Discovery). Lorsque vous suivez ce scénario, vous pouvez afficher les fichiers pour les comparer à votre sortie. Il s'agit d'exemples de fichier d'importation, d'exportation, d'objet métier et WSDL :

- JDBCOutboundInterface.import
- JDBCInbound Interface.export
- JDBCOutboundInterface.wsdl
- JDBCInboundInterface.wsdl
- InboundRtasserCustomer.xsd
- InboundRtasserAddress.xsd
- InboundRtasserCustInfo.xsd
- OutboundRtasserCustomer.xsd
- OutboundRtasserAddress.xsd
- OutboundRtasser.CustInfo.xsd
- InboundRtasserCustomerBG.xsd
- InboundRtasserAddressBG.xsd
- InboundRtasserCustInfoBG.xsd
- OutboundRtasserCustomerBG.xsd
- OutboundRtasserAddressBG.xsd
- OutboundRtasserCustInfoBG.xsd
- InboundRtasserCustomerContainer.xsd

- InboundRtasserAddressContainer.xsd
- InboundRtasserCustInfoContainer.xsd
- OutboundRtasserCustomerContainer.xsd
- OutboundRtasserAddressContainer.xsd
- OutboundrtasserCustInfoContainer.xsd

Pour plus d'informations sur les objets métier, voir «Présentation des objets métier».

Vous pouvez ensuite passer à la tâche de déploiement et de configuration.

Déploiement et configuration du scénario 1

Pour le scénario 1, l'exemple fournit tous les artefacts pour que vous n'ayez pas à les créer à l'aide de l'assistant (Enterprise Service Discovery).

L'exemple fournit une instance de l'adaptateur déjà configurée. Dans la procédure ci-après, vous devez mettre à jour les informations ASI des objets métier conformément à votre environnement. En outre, vous devez modifier les valeurs des propriétés de configuration conformément à l'instance de base de données de votre système EIS (Enterprise Information System).

1. Démarrez WebSphere Integration Developer

Pour plus de détails, reportez-vous au guide de l'utilisateur de WebSphere Integration Developer, à l'adresse suivante : <http://www.ibm.com/software/integration/wid>.
2. Créez le module Intégration métier

Dans WebSphere Integration Developer, accédez à la perspective Intégration métier. Cliquez avec le bouton droit sur Intégration métier. Cliquez sur **Nouveau > Module**. Dans la fenêtre Nouveau module, entrez **JDBC** pour la zone **Nom du module**. Cliquez sur **Terminer**.
3. Importez le fichier RAR (Resource Adapter Archive)
 - a. Cliquez sur **Fichier > Importer**. Dans la fenêtre de sélection, choisissez la source d'importation **Fichier RAR** et cliquez sur **Suivant**. Le fichier CWYBC_JDBC.rar est alors importé dans WebSphere Integration Developer.
 - b. Dans la fenêtre Importation du connecteur, dans la zone **Fichier du connecteur**, recherchez le chemin d'accès et le nom du module créé à l'étape précédente (par exemple, *répertoire d'installation*\jdbc\deploy\CWYBC_JDBC.rar). Vérifiez que la case **Ajouter un module à un projet EAR** est cochée. Dans la zone **Projet EAR**, sélectionnez **JDBCApp**. Cliquez sur **Terminer**.
 - c. Dans la perspective Intégration métier de la sous-fenêtre du même nom, cliquez avec le bouton droit sur le projet **JDBC** et sélectionnez **Ouvrir l'éditeur de dépendances**. Cliquez sur la flèche pour développer la section **J2EE**. Cliquez sur **Ajouter**. Dans la fenêtre **Sélection d'un projet J2EE**, sélectionnez **CWBC_JDBC** et cliquez sur **OK**. Dans le menu, cliquez ensuite sur **Fichier > Enregistrer**.
4. Ajoutez le pilote JDBC au chemin de compilation Java

Vous devez ajouter une référence au pilote JDBC de votre projet.

Dans la perspective Java de WebSphere Integration Developer, cliquez avec le bouton droit sur le **projet de connecteur CWBC_JDBC**. Sélectionnez **Propriétés**.

Pour ajouter le fichier JAR externe, cliquez sur le bouton **Chemin de compilation Java**. Sélectionnez l'onglet **Bibliothèques**, puis cliquez sur **Ajouter des fichiers JAR externes**. Dans la fenêtre Système de fichiers, accédez à **Pilote JDBC** et sélectionnez le fichier JAR.

Cliquez sur **OK** pour sauvegarder vos modifications.

5. Ajoutez des artefacts d'objet métier au module Intégration métier
Vous devez décompresser le fichier JDBCApp.ear.
Vous devez ensuite extraire le fichier JAR contenu dans le fichier EAR (Enterprise Application Archive). Extrayez le fichier JDBC.jar du fichier JDBCAPP.ear dans le répertoire \workspace\JDBC.
6. Modifiez les informations ASI des objets métier
 - a. Dans la perspective Intégration métier, cliquez avec le bouton droit sur le module **JDBC** et cliquez sur **Régénérer**.
 - b. Cliquez sur le signe + en regard de **Types de données**. Cliquez avec le bouton droit sur l'objet métier **InboundRtasserCustomer.xsd**.
 - c. Cliquez sur **Ouvrir avec > Editeur d'objets métier**. Sélectionnez l'onglet **Propriétés**. Sélectionnez l'onglet **Informations sur l'application**.
 - d. Dans la sous-fenêtre Propriétés ASI, remplacez la valeur de jdbcasi:TableName (**RTASSER.CUSTOMER**) par *schema.CUSTOMER* en utilisant le nom du schéma de votre base de données où la table CUSTOMER a été créée. Dans le menu, cliquez sur **Fichier > Enregistrer**.
Répétez les étapes 6a à 6d pour les objets métier suivants :
 - InboundRtasserAddress
 - InboundRtasserCustInfo
 - OutboundRtasserCustomer
 - OutboundRtasserAddress
 - OutboundRtasserCustInfo
7. Entrez les informations d'authentification
 - a. Vous devez créer un alias d'authentification, à l'aide de la console d'administration de WebSphere Process Server. Ouvrez la console d'administration. Dans le panneau de gauche, cliquez sur **Sécurité > Sécurité globale**. Dans la sous-fenêtre Configuration de la fenêtre **Sécurité globale**, accédez à l'en-tête **Authentification**, dans la partie droite. Cliquez sur **Configuration JAAS**. Cliquez ensuite sur le lien **Données d'authentification J2C**.
 - b. Dans la fenêtre Sécurité globale > Entrées des données d'authentification de J2EE Connector Architecture (J2C), cliquez sur **Nouveau**. Entrez un nom d'alias dans la zone **Alias**. Entrez un ID utilisateur et un mot de passe permettant de se connecter à la base de données. Cliquez sur **OK**. Cliquez ensuite sur **Enregistrer** pour sauvegarder les informations d'authentification.
8. Modifiez les valeurs des propriétés de configuration
 - a. Vous devez modifier les valeurs des propriétés de configuration conformément à votre instance de base de données. Dans WebSphere Integration Developer, dans la perspective Intégration métier de la sous-fenêtre du même nom, cliquez sur le **module JDBC**. Cliquez deux fois sur **Inbound/JDBCInboundInterface**.
 - b. Dans la page **Propriétés** de l'éditeur d'associations, cliquez sur l'onglet **Association**, puis sur l'onglet **Connexion**.

- c. Développez **ActivationSpecProperties**. Affectez aux propriétés `UserName`, `Password`, `DatabaseURL` et `JDBCClass` les valeurs appropriées pour la spécification d'activation J2C, en fonction de votre configuration de base de données.
- d. Dans la page **Connection**, développez **Authentication Properties**. Dans la zone de **Saisie des données d'authentification J2C**, entrez le nom de l'alias d'authentification créé à l'étape précédente. Dans le menu, cliquez sur **Fichier > Enregistrer**.

Vous devez répéter cette procédure pour l'objet `Outbound/JDBCOutboundInterface`. Mettez à jour les valeurs des propriétés de la fabrique de connexions J2C (également appelée fabrique de connexions gérées) : `UserName`, `Password`, `DatabaseURL` et `JDBCClass`.

- 9. Déployez le fichier du projet sur le serveur d'applications

Vous devez exporter le fichier EAR de votre projet de WebSphere Integration Developer vers WebSphere Process Server.

- a. Exportez le fichier EAR

Dans le menu WebSphere Integration Developer, cliquez sur **Fichier > Importer**. Sélectionnez le fichier EAR, puis cliquez sur **Suivant**. Dans la fenêtre Exportation EAR, accédez à la zone **Projet EAR** et entrez **JDBCApp**. Définissez le chemin d'accès de destination et le nom du fichier. Cochez les trois cases et cliquez sur **Terminer**.

- b. Installez le fichier EAR sur le serveur d'applications

A l'aide de la console d'administration, sélectionnez **Applications > Installation d'une nouvelle application**. Cliquez sur **Parcourir**, sélectionnez le fichier EAR et cliquez sur **Suivant**.

Continuez de cliquer sur **Suivant** jusqu'à ce que vous atteigniez l'étape 7. **Mapper des références de ressources sur des ressources**. Sélectionnez l'alias d'authentification que vous avez créé précédemment, cochez la case, puis cliquez sur **Appliquer**.

Dans les écrans qui suivent, cliquez sur **Suivant** ou sur **Continuer** jusqu'à ce que la fenêtre **Récapitulatif des options d'installation** apparaisse. Cliquez ensuite sur **Terminer**.

Le message **Application SampleApp installée avec succès** apparaît. Cliquez sur **Enregistrer dans la configuration principale** pour sauvegarder vos modifications. Cliquez ensuite sur **Enregistrer**.

Vous pouvez alors exécuter l'exemple d'application.

Déploiement et configuration du scénario 2

Dans le scénario 2, vous définissez les propriétés de connexion de l'adaptateur et générez des objets métier. Vous exportez votre projet dans un fichier EAR (Enterprise Application Archive), déployez le projet sur le serveur d'applications et réinitialisez les propriétés de configuration.

- 1. Démarrez WebSphere Integration Developer

Pour plus de détails, reportez-vous au guide de l'utilisateur de WebSphere Integration Developer, à l'adresse suivante : <http://www.ibm.com/software/integration/wid>.

- 2. Créez le module Intégration métier

Dans WebSphere Integration Developer, accédez à la perspective Intégration métier. Cliquez avec le bouton droit sur Intégration métier. Sélectionnez **Nouveau > Module**. Dans la fenêtre Nouveau module, entrez JDBC pour la zone **Nom du module**. Cliquez sur **Terminer**.

3. Importez le fichier RAR de l'adaptateur

Dans WebSphere Integration Developer, sélectionnez **Fichier > Importer** pour importer le fichier RAR de l'adaptateur. Le fichier RAR est `jdbc/deploy/CWYBC_JDBC.rar`.

Dans la fenêtre **Sélectionner**, sélectionnez **Fichier RAR** comme source d'importation.

Dans la fenêtre Importation du connecteur, dans la zone **Fichier du connecteur**, recherchez le chemin d'accès et le nom du module créé à l'étape précédente (par exemple, *répertoire d'installation/jdbc/deploy/CWYBC_JDBC.rar*).

Vérifiez que la case **Ajouter un module à un projet EAR** est cochée. Dans la zone **Projet EAR**, sélectionnez **JDBCApp**. Cliquez sur **Terminer**.
4. Ajoutez le pilote JDBC au chemin de compilation Java

Vous devez ajouter une référence au pilote JDBC de votre projet.

Dans la perspective **Java** de WebSphere Integration Developer, cliquez avec le bouton droit sur le **projet de connecteur CWBC_JDBC**. Sélectionnez **Propriétés**.

Pour ajouter le fichier JAR externe, cliquez sur **Chemin de compilation Java**. Sélectionnez l'onglet **Bibliothèques**, puis cliquez sur **Ajouter des fichiers JAR externes**. Dans la fenêtre Système de fichiers, accédez à **Pilote JDBC** et sélectionnez le fichier JAR.

Cliquez sur **OK** pour sauvegarder vos modifications.
5. Entrez les informations d'authentification

Ouvrez la console d'administration de WebSphere Process Server. Dans le panneau de gauche, sélectionnez **Sécurité > Sécurité globale**. Dans la sous-fenêtre Configuration de la fenêtre **Sécurité globale**, accédez à l'en-tête **Authentification**, dans la partie droite. Cliquez sur **Configuration JAAS**. Cliquez ensuite sur le lien **Données d'authentification J2C**.

Dans la fenêtre Sécurité globale > Entrées des données d'authentification de J2EE Connector Architecture (J2C), cliquez sur **Nouveau**. Entrez un nom d'alias dans la zone **Alias**. Entrez un ID utilisateur et un mot de passe permettant de se connecter à la base de données. Cliquez sur **OK**. Cliquez sur **Enregistrer** pour sauvegarder les informations d'authentification.
6. Définissez les propriétés de connexion de l'adaptateur
 - a. Dans la page **Intégration métier** de WebSphere Integration Developer contenant le projet de connecteur JDBC, cliquez sur le cadre avec le bouton droit. Dans le menu en incrustation, sélectionnez **Nouveau > Enterprise Service Discovery**.
 - b. Dans la fenêtre Sélection d'un adaptateur de ressources de service d'entreprise, sélectionnez **JDBC EMD Adapter** et cliquez sur **fichier RAR**.
 - c. Dans la fenêtre Configuration des paramètres de l'agent de reconnaissance, entrez les informations de votre base de données dans les zones des propriétés de configuration des connexions. L'exemple ne requiert que quatre valeurs de propriété de connexion. Pour plus de détails sur ces propriétés, voir «Propriétés de connexion» dans la section «Référence». Le tableau suivant illustre des exemples de valeur pour initialiser l'agent de reconnaissance.

Propriétés de connexion à la fonction de recherche des métadonnées

Propriété	Exemples de valeur
UserName	db2admin

Propriété	Exemples de valeur
Password	xxxxxxxxxxx
DatabaseURL	jdbc:db2:mytestdb
JDBCdriverClass	COM.ibm.db2.jdbc.app.DB2Driver

Cliquez sur **Suivant**.

7. Recherchez des objets de base de données.

Après avoir configuré les propriétés de connexion, vous pouvez exécuter une requête sur les objets de base de données. Vous pouvez explorer l'arborescence des métadonnées pour comprendre la structure des objets du système EIS (Enterprise Information System) et sélectionner les objets requis pour la description de service.

A ce stade, vous pouvez spécifier des propriétés de filtre pour restreindre la liste des schémas affichés dans l'arborescence. Reportez-vous à la procédure de spécification des propriétés de filtre dans «Recherche d'objets de base de données».

Dans la fenêtre Recherche et reconnaissance des services d'entreprise, cliquez sur **Exécuter la requête**.

Les objets de métadonnées sont affichés dans la sous-fenêtre supérieure. Sélectionnez l'objet source de la génération, puis cliquez sur **Ajouter**. L'objet sélectionné apparaît dans la sous-fenêtre inférieure. Vous pouvez supprimer un objet sélectionné en cliquant sur **Supprimer**.

Une fois que vous avez terminé de sélectionner des objets, cliquez sur **Suivant**.

8. Spécifiez les propriétés de sélection.

Une fois que vous avez sélectionné les objets de base de données, vous devez spécifier les valeurs des propriétés de sélection pour les fichiers d'importation et d'exportation.

La propriété Namespace est définie par défaut pour tous les objets métier.

La sélection du type de service (entrant ou sortant) et des opérations correspondantes est décrite dans «Définition des propriétés de sélection.» Une fois que vous avez spécifié ces propriétés de sélection, cliquez sur **Suivant**.

9. Sauvegardez le projet d'adaptateur

Le nom de module **JDBC** apparaît automatiquement dans la fenêtre Génération des artefacts. Ne modifiez pas la valeur associée à la case à cocher. Cliquez sur **Nouveau** pour créer un module Intégration métier dans lequel les artefacts seront sauvegardés.

Dans la fenêtre Nouveau module, le nom du module apparaît. Un nom de dossier par défaut est sélectionné. Il s'agit du dossier dans lequel les descriptions de service seront sauvegardées.

Dans la zone **Saisie des données d'authentification J2C**, entrez le nom de l'alias d'authentification créé précédemment.

Cliquez sur **Terminer**.

Vous pouvez comparer votre sortie aux exemples de fichier. Dans la perspective Intégration métier de la sous-fenêtre Intégration métier, cliquez sur **+**, en regard du module **JDBC** pour le développer. Cliquez sur le signe **+** en regard du dossier **Types de données** pour afficher la liste de tous les objets métier.

10. Déployez le fichier du projet sur le serveur d'applications

Votre fichier de projet correspond à un projet de connecteur J2EE dans l'espace de travail de WebSphere Integration Developer. Vous devez l'exporter dans votre système de fichiers local. Vous devez ensuite fournir les informations d'authentification pour la sécurité du connecteur JCA avant de pouvoir télécharger et installer le fichier EAR sur le serveur d'applications.

a. Exportez le fichier EAR

Dans l'écran Exportation de WebSphere Integration Developer, sélectionnez le fichier EAR **JDBCApp** et cliquez sur **Suivant**.

Dans la fenêtre Exportation EAR, vous devez sélectionner le module Intégration métier et l'emplacement dans lequel le fichier EAR doit être créé. Sélectionnez le module dans la zone **Projet EAR**. Entrez l'emplacement dans lequel le fichier EAR doit être créé ou recherchez-le. Cochez toutes les cases, puis cliquez sur **Terminer**.

b. Installez le fichier EAR sur le serveur d'applications

A l'aide de la console d'administration, dans la sous-fenêtre de gauche, cliquez sur **Applications > Installation d'une nouvelle application**.

Cliquez sur **Parcourir**, sélectionnez le fichier EAR et cliquez sur **Suivant**.

Continuez de cliquer sur **Suivant** jusqu'à ce que vous atteigniez l'étape 7. **Mapper des références de ressources sur des ressources**. Sélectionnez l'alias d'authentification que vous avez créé précédemment, cochez la case **Utiliser la méthode par défaut**, puis cliquez sur le bouton **Appliquer**.

Dans les écrans qui suivent, cliquez sur **Suivant** ou sur **Continuer** jusqu'à ce que la fenêtre **Récapitulatif des options d'installation** apparaisse. Cliquez ensuite sur **Terminer**.

Le message **Application SampleApp installée avec succès** apparaît.

Cliquez sur **Enregistrer dans la configuration principale** pour sauvegarder vos modifications. Cliquez ensuite sur **Enregistrer** dans la sous-fenêtre Applications d'entreprise.

11. Modifiez les valeurs des propriétés de configuration

Les propriétés d'adaptateur Username, Password, DatabaseURL et JDBCClass sont déjà remplies par défaut, mais vous devez les reconfigurer conformément à votre instance de base de données, à l'aide de la console d'administration.

Pour le traitement des événements entrants, vous devez modifier les propriétés de la spécification d'activation J2C. Pour le traitement des événements sortants, vous devez modifier les propriétés de la fabrique de connexions J2C. L'exemple ne requiert pas que vous reconfiguriez les valeurs des autres propriétés. Vous pouvez également modifier les propriétés de configuration de l'adaptateur de ressources selon vos besoins. Pour plus de détails sur ces propriétés, voir les propriétés de «configuration» dans la section «Référence».

a. Ouvrez le fichier du projet d'adaptateur

Dans la console d'administration, sous l'onglet **Configuration** de la fenêtre Applications d'entreprise, cliquez sur le nom de votre projet d'adaptateur. Sous l'en-tête **Articles liés**, sélectionnez **Modules de connecteurs**. Le nom du fichier RAR du projet est affiché. Cochez la case en regard du nom de fichier.

Dans la page **Configuration**, sous l'en-tête **Propriétés supplémentaires**, sélectionnez **Adaptateur de ressources**.

b. Reconfigurez les propriétés

Votre projet d'adaptateur est affiché sous **Nom** dans la sous-fenêtre Propriétés générales. Sous **Propriétés supplémentaires**, cliquez sur l'une

des catégories de propriétés (spécification d'activation J2C, fabrique de connexions J2C ou propriétés personnalisées) pour éditer les valeurs de propriété. Les propriétés personnalisées correspondent aux propriétés de l'adaptateur de ressources.

Vous pouvez alors exécuter l'exemple d'application.

Tâches associées

«Recherche d'objets de base de données», à la page 46

Après avoir configuré les propriétés de connexion, vous pouvez exécuter une requête sur les objets de base de données. Vous pouvez explorer l'arborescence des métadonnées pour comprendre la structure des objets du système EIS (Enterprise Information System) et sélectionner les objets requis pour la description de service.

«Définition des propriétés de sélection», à la page 49

Une fois que vous avez sélectionné les objets de base de données, vous devez spécifier les valeurs des propriétés de sélection pour les fichiers d'importation et d'exportation.

Référence associée

«Propriétés de connexion», à la page 86

Les propriétés de connexion permettent à l'assistant ESD (Enterprise Service Discovery) de se connecter à l'instance de système EIS (Enterprise Information System) cible.

«Propriétés de configuration», à la page 75

Ce groupe de propriétés contient les attributs utilisés par l'adaptateur pour configurer un canal de communication vers une application de base de données spécifique.

Exécution de l'exemple d'application

Dans cette tâche, vous exécutez l'exemple d'application de l'adaptateur pour tester le traitement d'un événement lors des opérations de communication entrante. Vous pouvez vérifier la précision des valeurs de sortie de l'événement.

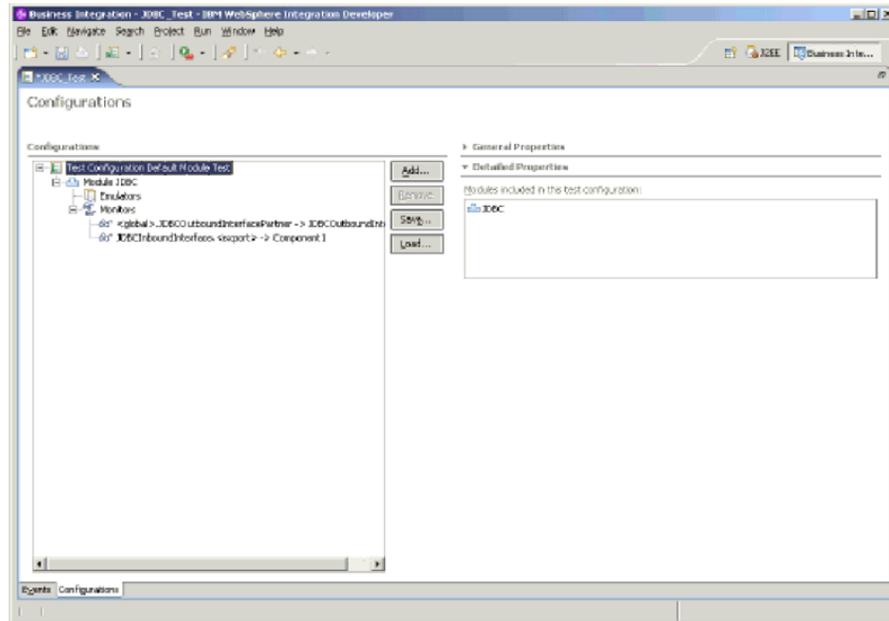
L'exemple est exécuté à l'aide du client de test de WID (WebSphere Integration Developer).

1. Sélectionnez le module Intégration métier et le serveur

Dans la perspective Intégration métier de WebSphere Integration Developer, sélectionnez le module **JDBC** dans la page Intégration métier. Cliquez avec le bouton droit sur **Tester > Connecter**.

Le client de test de WebSphere Integration Developer affiche la fenêtre des événements. Dans la partie droite, le **test de module par défaut** est affiché dans la zone **Configuration**. La zone **Module** contient **JDBC**.

Cliquez sur l'onglet **Configurations** dans la partie inférieure gauche. Dans la fenêtre Configurations, cliquez sur le signe + en regard de **Moniteurs** pour vérifier que le moniteur **JDBCInboundInterface <export> Component1** existe pour l'exportation. Si vous avez effectué le scénario 1, ce fichier de moniteur a été généré automatiquement. Si vous avez effectué le scénario 2, ce fichier a été créé à l'étape 9, «Enregistrer le projet d'adaptateur.»



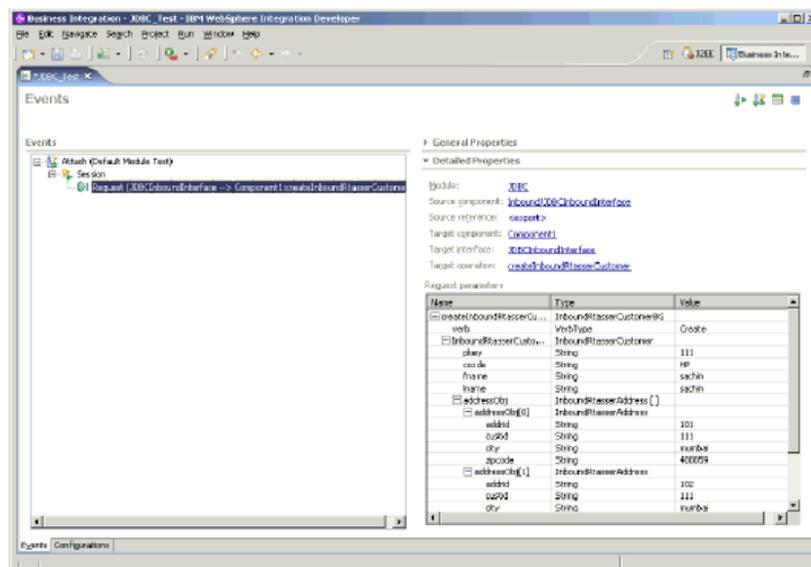
Cliquez sur l'onglet **Événements** dans la partie inférieure gauche, puis sur **Continuer**. Dans la fenêtre en incrustation vous invitant à sélectionner l'emplacement du déploiement, sélectionnez le serveur d'applications pour exécuter l'application. Sélectionnez **WebSphere Process Server v6.0**, puis cliquez sur **Terminer**. La fenêtre en incrustation **Démarrage du client de test d'intégration** apparaît.

2. Testez le traitement des événements

Pour tester le traitement des événements, vous pouvez insérer manuellement un événement dans la table des événements de l'exemple d'application. Utilisez l'exemple suivant pour vous aider à insérer l'événement dans le magasin d'événements à l'aide d'une instruction SQL :

```
INSERT INTO magasinévénements_wbia_jdbc (clé_objet, nom_objet,
fonction_objet,
priorité_événement, état_événement)
VALUES ('11234', 'InboundRtasserCustomerBG', 'Create', 1, 0)
```

Lorsque le composant surveillé reçoit l'événement, une entrée apparaît dans la fenêtre Événements. Reportez-vous à l'exemple de fenêtre Événements ci-après.



Référence

Cette section décrit les propriétés de configuration, de connexion et de sélection d'objets.

Concepts associés

«Gestion de transactions», à la page 15

L'adaptateur pour JDBC prend en charge les transactions locales et XA.

Propriétés de configuration

Ce groupe de propriétés contient les attributs utilisés par l'adaptateur pour configurer un canal de communication vers une application de base de données spécifique.

Les types de propriété sont les suivants :

- Adaptateur de ressources
- Fabrique de connexions J2C
- Spécification d'activation J2C

Les propriétés bidirectionnelles sont décrites pour chacun de ces types de propriété.

Tâches associées

«Sauvegarde du projet d'adaptateur», à la page 50

Une fois que vous avez spécifié les propriétés de sélection, configurez les propriétés utilisées par l'adaptateur pour configurer un canal de communication vers une application de base de données spécifique. Il s'agit des propriétés de l'adaptateur de ressources, de la fabrique de connexions J2C, de la spécification d'activation J2C, ainsi que des propriétés bidirectionnelles. Vous devez également créer un module Business Integration dans lequel tous les artefacts et les valeurs des propriétés seront sauvegardés.

«Déploiement et configuration du scénario 2», à la page 69

Dans le scénario 2, vous définissez les propriétés de connexion de l'adaptateur et générez des objets métier. Vous exportez votre projet dans un fichier EAR (Enterprise Application Archive), déployez le projet sur le serveur d'applications et réinitialisez les propriétés de configuration.

Propriétés de l'adaptateur de ressources

Ces propriétés de configuration sont définies au niveau de l'adaptateur de ressources.

Le tableau «Propriétés de l'adaptateur de ressources» décrit ces propriétés de configuration. Les propriétés bidirectionnelles de l'adaptateur de ressources devant être configurées lors de l'activation d'une conversion de texte bidirectionnel sont décrites dans un tableau séparé de la présente section.

Propriétés de l'adaptateur de ressources

Propriété	Type	Globalisée	Description
BONamespace	String (Chaîne)	Non	Espace de nom des définitions d'objet métier à utiliser par cet adaptateur. Cette valeur doit être extraite de la valeur que vous avez spécifiée lors de la procédure de reconnaissance des métadonnées. Cette propriété est requise.
DatabaseVendor	String (Chaîne)	Non	Indique quel SGBDR est utilisé par l'adaptateur pour un traitement spécifique. Définissez la valeur DB2, Oracle ou MSSQLServer selon que vous une base de données IBM DB2, Oracle ou Microsoft SQL Server. Si vous utilisez une autre base de données, choisissez le nom de cette dernière. Si vous utilisez une base de données autre que celle par défaut, vérifiez que le pilote chargé est approprié. Si cette propriété a la valeur Others, l'adaptateur détermine quelle base de données utiliser en localisant le pilote. Il est indispensable d'indiquer une valeur pour que l'adaptateur fonctionne correctement.
LogFileName	String (Chaîne)	Non	Chemin d'accès complet du fichier journal. Cette propriété est requise.

Propriété	Type	Globalisée	Description
LogNumberOfFiles	Integer (nombre entier)	Non	Nombre de fichiers journaux à utiliser. Lorsqu'un fichier journal atteint sa taille maximale, un autre fichier journal est créé. Si aucune valeur n'est spécifiée, la valeur 1 est utilisée.
LogMaxFileSize	Integer (nombre entier)	Non	Taille des fichiers journaux en kilo-octets. Si aucune valeur n'est spécifiée, les fichiers ne sont pas limités en taille.
PingQuery	String (Chaîne)	Non	Requête SQL permettant de tester si la connexion à la base de données est valide.
QueryTimeOut	Integer (nombre entier)	Oui	Ce paramètre spécifie la durée maximale (en secondes) du traitement de toutes les instructions SQL par la requête. Si aucune valeur n'est indiquée, cela signifie qu'aucun délai d'expiration n'est défini pour la requête. Si la requête prend plus de temps que le nombre de secondes indiqué, la base de données génère une exception SQL qui est enregistrée. Le message associé est consigné dans le fichier journal.

Propriété	Type	Globalisée	Description
ReturnDummy BOforSP	Boolean (booléen)	Non	Permet de renvoyer des paramètres de sortie même lorsque le jeu de résultats est vide. Dans le cas de la propriété RetrieveSP, un jeu de résultats est renvoyé. Si le jeu de résultats est vide, aucun objet métier n'est créé, et les paramètres de sortie renvoyés par l'appel de procédure ne peuvent pas être récupérés. Si la valeur de cette propriété est true, un objet métier fictif comprenant des valeurs issues des paramètres d'entrée-sortie indiqués dans les attributs correspondants est renvoyé. La valeur par défaut est false.
TraceFileName	String (Chaîne)	Non	Chemin d'accès complet du fichier de trace. Cette propriété est requise.
TraceNumberOfFiles	Integer (nombre entier)	Non	Nombre de fichiers de trace à utiliser. Lorsqu'un fichier de trace atteint sa taille maximale, un autre fichier de trace est créé. Si aucune valeur n'est spécifiée, la valeur 1 est utilisée.
TraceFileSizeMax	Integer (nombre entier)	Non	Taille des fichiers de trace en kilo-octets. Si aucune valeur n'est spécifiée, les fichiers ne sont pas limités en taille.

Le tableau suivant décrit les propriétés bidirectionnelles de l'adaptateur de ressources devant être configurées lors de l'activation d'une conversion de texte bidirectionnel.

Propriétés bidirectionnelles de l'adaptateur de ressources

Propriété	Type	Description
BiDiContextEIS	String (Chaîne)	Définit le format bidirectionnel des données de contenu de toutes les instances d'exécution d'objets métier.
BiDiContextMetadata	String (Chaîne)	Définit le format bidirectionnel des propriétés de métadonnées ou de configuration enregistrées dans le descripteur de déploiement.
BiDiContextSkip	Boolean (booléen)	Indicateur servant à contrôler l'appel du support bidirectionnel au niveau de l'adaptateur de ressources. Si la valeur true est définie, l'appel du support n'a pas lieu. Si elle vaut false, le support est appelé.
BiDiContextSpecialFormat	String (Chaîne)	Spécifie la catégorie de propriétés sujettes à un traitement spécial pour toutes les propriétés de connecteur.
BiDiContextTurnBiDiOff	Boolean (booléen)	Indicateur servant à contrôler l'appel de transformation bidirectionnelle au niveau de l'adaptateur JDBC. Ce paramètre prévaut sur l'ensemble des paramètres bidirectionnels, quel que soit le niveau. En d'autres termes, lorsque la valeur true est définie, aucun support bidirectionnel ni contrôle de consultation n'est effectué. Si la valeur définie est false, l'appel du support bidirectionnel a lieu.

Tâches associées

«Configuration de l'adaptateur sur le serveur», à la page 54

Une fois que le fichier EAR (Enterprise Application Archive) du projet d'adaptateur a été installé sur le serveur d'applications, vous pouvez, si vous le souhaitez, reconfigurer les propriétés utilisées par l'adaptateur pour configurer un canal de communication vers une application de base de données spécifique. Vous pouvez ensuite démarrer l'application d'adaptateur configurée.

Propriétés des fabriques de connexions J2C

Les propriétés des fabriques de connexions J2C permettent de configurer une instance de système EIS (Enterprise Information System) cible. Ces propriétés affectent le traitement des communications sortantes et correspondent à l'interface ManagedConnectionFactory de la spécification JCA (J2EE Connector Architecture).

Une fabrique de connexions J2C gère le regroupement des connexions. Elle fournit des informations de configuration sur la connectivité des communications sortantes d'une application à une instance d'application JDBC par l'intermédiaire de l'adaptateur.

Le tableau intitulé «Propriétés des fabriques de connexions J2C» définit les propriétés de configuration JDBC relatives à une fabrique de connexions J2C. Les deux tableaux suivants de la présente section définissent les propriétés bidirectionnelles relatives à la fabrique de connexions J2C, dont la configuration est requise uniquement lors de l'activation d'une conversion de texte bidirectionnel.

Propriétés des fabriques de connexions J2C

Propriété	Type	Globalisée	Description
AutoCommit	Boolean (booléen)	Non	Valeur de la propriété AutoCommit à définir lors de la connexion.
DatabaseURL	String (Chaîne)	Oui	URL de base de données permettant de se connecter à la base de données. Activée pour une utilisation avec les langues bidirectionnelles.
JDBCDriverClass	String (Chaîne)	Non	Classe du pilote JDBC utilisé pour se connecter à la base de données.
Password	String (Chaîne)	Oui	Mot de passe de l'utilisateur correspondant. Activée pour une utilisation avec les langues bidirectionnelles.
UserName	String (Chaîne)	Oui	Nom d'utilisateur pour la connexion au système EIS (Enterprise Information System). Activée pour une utilisation avec les langues bidirectionnelles.
XADataSourceName	String (Chaîne)	Non	Nom de la source de données XA permettant d'établir une connexion XA à la base de données.

Propriété	Type	Globalisée	Description
XADatabaseName	String (Chaîne)	Oui	Nom de la base de données utilisée pour la connexion XA. Cette propriété doit être définie si vous utilisez une base de données DB2.

Propriétés bidirectionnelles de la fabrique de connexions J2C

Propriété	Type	Description
BiDiContextEIS	String (Chaîne)	Définit le format bidirectionnel des données de contenu de toutes les instances d'exécution d'objets métier prises en charge par l'adaptateur pour la connexion spécifique à l'EIS.
BiDiContextMetadata	String (Chaîne)	Définit le format bidirectionnel des propriétés de métadonnées ou de configuration d'une connexion spécifique à l'EIS.
BiDiContextSkip	Boolean (booléen)	Indicateur servant à contrôler l'appel du support bidirectionnel au niveau de l'adaptateur de ressources. Si la valeur true est définie, l'appel du support n'a pas lieu. Si elle vaut false, le support est appelé.
BiDiContextSpecialFormat	String (Chaîne)	Spécifie la catégorie de propriétés sujettes à un traitement spécial pour toutes les propriétés de connecteur liées à la connexion spécifique à l'EIS.
Propriétés BiDi de l'URL de base de données		*DatabaseURLEIS, DatabaseURLSpecialFormat, DatabaseURLSkip
Propriétés BiDi du mot de passe		*PasswordEIS, PasswordSpecialFormat, PasswordSkip
Propriétés BiDi du nom d'utilisateur		*UserNameEIS, UserNameSpecialFormat, UserNameSkip

*Les trois propriétés bidirectionnelles (BiDi) associées à chaque propriété bidirectionnelle prise en charge. Ces propriétés sont décrites dans le tableau ci-après.

Propriété BiDi	Type	Description
BiDiContextCP<property_Name>EIS	String (Chaîne)	Définit le format bidirectionnel des propriétés d'une propriété de configuration de connecteur spécifique, pour une connexion spécifique à l'EIS.
BiDiContextCP<property_Name>SpecialFormat	String (Chaîne)	Indicateur servant à contrôler l'appel du support bidirectionnel au niveau de la propriété de configuration du connecteur. Si la valeur true est définie, l'appel du support n'a pas lieu. Si elle vaut false, le support est appelé.
BiDiContextCP<property_Name>Skip	Boolean (booléen)	Définit le format spécial des propriétés d'une propriété de configuration de connecteur spécifique, pour une connexion spécifique à l'EIS.

Concepts associés

«Gestion de transactions», à la page 15

L'adaptateur pour JDBC prend en charge les transactions locales et XA.

Tâches associées

«Configuration de l'adaptateur sur le serveur», à la page 54

Une fois que le fichier EAR (Enterprise Application Archive) du projet d'adaptateur a été installé sur le serveur d'applications, vous pouvez, si vous le souhaitez, reconfigurer les propriétés utilisées par l'adaptateur pour configurer un canal de communication vers une application de base de données spécifique. Vous pouvez ensuite démarrer l'application d'adaptateur configurée.

Propriétés de la spécification d'activation J2C

Les propriétés de la spécification d'activation J2C activent les noeuds finaux de message. Ces propriétés correspondent à l'interface ActivationSpec de la spécification JCA (J2EE Connector Architecture Specification).

Le tableau intitulé «Propriétés de la spécification d'activation J2C» décrit les propriétés de configuration spécifiques à l'adaptateur JDBC relatives à l'activation des noeuds finaux de message. Un tableau distinct de la présente section décrit les propriétés bidirectionnelles relatives à la spécification d'activation J2C, dont la configuration est requise uniquement lors de l'activation d'une conversion de texte bidirectionnel.

Propriétés de la spécification d'activation J2C

Propriété	Type	Globalisée	Description
AutoCreateEDT	Boolean (booléen)	Oui	Indicateur déterminant si l'adaptateur doit créer la table EDT automatiquement si elle n'existe pas déjà. La valeur par défaut est true.

Propriété	Type	Globalisée	Description
DatabaseURL	String (Chaîne)	Oui	URL spécifique au pilote pour la création d'une connexion au système EIS. Activée pour une utilisation avec les langues bidirectionnelles.
Type de distribution	String (Chaîne)	Oui	Cette propriété détermine l'ordre selon lequel les événements sont publiés. Les valeurs possibles sont ORDERED ou UNORDERED. Les événements <i>ordonnés</i> sont publiés à raison d'un à la fois, tandis que les événements <i>non ordonnés</i> sont tous publiés simultanément. La valeur par défaut est ORDERED.
EDTDatabaseName	String (Chaîne)	Oui	Nom de la base de données de la table de distribution des événements.
EDTDriverName	String (Chaîne)	Oui	Nom du pilote de base de données XA utilisé pour se connecter à la table de distribution des événements pour les événements entrants. Exemple : com.ibm.db2j.DB2jXADataSource. Si aucune valeur n'est spécifiée, le gestionnaire d'événements ne peut pas effectuer de reprise sur incident.
EDTSchemaName	String (Chaîne)	Oui	Nom du schéma de la table de distribution des événements.
EDTTableName	String (Chaîne)	Oui	Nom de la table représentant la table de distribution des événements.
EDTUserName	String (Chaîne)	Oui	Nom d'utilisateur requis pour se connecter à la base de données de la table de distribution des événements.
EDTUserPassword	String (Chaîne)	Oui	Mot de passe requis pour se connecter à la base de données de la table de distribution des événements.
EventOrderBy	String (Chaîne)	Non	Ordre suivant lequel les événements sont extraits et traités. Les valeurs attendues sont des noms de colonne de la table des événements, séparés par des virgules. Cette propriété est activée pour une utilisation avec les langues bidirectionnelles.
EventTableName	String (Chaîne)	Oui	Tableau du système EIS contenant les événements générés par le système EIS pour le traitement des événements entrants.
JDBCDriverClass	String (Chaîne)	Non	Classe de pilote JDBC utilisée pour la connexion au système EIS.
Password	String (Chaîne)	Oui	Mot de passe permettant d'autoriser l'utilisateur à extraire les événements du système EIS. Cette propriété est activée pour une utilisation avec les langues bidirectionnelles.

Propriété	Type	Globalisée	Description
PollPeriod	Entier égal ou supérieur à 0	Oui	Fréquence en millisecondes à laquelle le magasin d'événements EIS est interrogé pour rechercher les nouveaux événements entrants. Si cette fréquence est égale à 0, l'adaptateur n'attend pas entre les cycles. Le cycle d'interrogation est établi à une fréquence fixe, ce qui signifie que si une exécution du cycle d'interrogation est retardée (par exemple, si le cycle d'interrogation précédent dure plus longtemps que prévu), le cycle suivant est exécuté immédiatement pour rattraper le retard. Cette propriété est requise. La valeur par défaut est 500.
PollQuantity	Entier supérieur à 0	Oui	Cette propriété permet de déterminer le nombre d'événements à distribuer à chaque noeud final par cycle d'interrogation. Cette propriété est requise.
Username	String (Chaîne)	Oui	Nom d'utilisateur en vue de la connexion au système EIS pour les événements entrants. Activée pour une utilisation avec les langues bidirectionnelles.

Propriétés bidirectionnelles de la spécification d'activation J2C

Propriété	Type	Description
BiDiContextEIS	String (Chaîne)	Définit le format bidirectionnel des données de contenu de toutes les instances d'exécution d'objets métier prises en charge par l'adaptateur pour la connexion spécifique à l'EIS.
BiDiContextMetadata	String (Chaîne)	Définit le format bidirectionnel des propriétés de métadonnées ou de configuration d'une connexion spécifique à l'EIS.
BiDiContextSkip	Boolean (booléen)	Indicateur servant à contrôler l'appel du support bidirectionnel au niveau de l'adaptateur de ressources. Si la valeur true est définie, l'appel du support n'a pas lieu. Si elle vaut false, le support est appelé.
BiDiContextSpecialFormat	String (Chaîne)	Spécifie la catégorie de propriétés sujettes à un traitement spécial pour toutes les propriétés de connecteur liées à une connexion spécifique à l'EIS.

Propriété	Type	Description
Propriétés BiDi de l'URL de base de données		*DatabaseURLEIS, DatabaseURLSpecialFormat, DatabaseURLSkip
Propriétés BiDi du mot de passe		*PasswordEIS, PasswordSpecialFormat, PasswordSkip
Propriétés BiDi du nom d'utilisateur		*UserNameEIS, UserNameSpecialFormat, UserNameSkip
Propriétés BiDi du nom de la table d'événements		*EventTableNameEIS, EventTableNameSpecialFormat, EventTableNameSkip
EventOrderBy		*EventOrderByEIS, EventOrderBySpecialFormat, EventOrderBySkip
EDTTableName		EDT_BiDiFormat, EDT_BiDiSkip, EDTURL_BiDiSpecialFormat, EDTURL_BiDiSkip
EDTUserName		EDT_BiDiFormat, EDT_BiDiSkip, EDTURL_BiDiSpecialFormat, EDTURL_BiDiSkip
EDTUserPassword		EDT_BiDiFormat, EDT_BiDiSkip, EDTURL_BiDiSpecialFormat, EDTURL_BiDiSkip
EDTDatabaseName		EDT_BiDiFormat, EDT_BiDiSkip, EDTURL_BiDiSpecialFormat, EDTURL_BiDiSkip

*Les trois propriétés bidirectionnelles (BiDi) associées à chaque propriété bidirectionnelle prise en charge. Ces propriétés sont décrites dans le tableau ci-après.

Propriété BiDi	Type	Description
BiDiContextCP<property_Name>EIS	String (Chaîne)	Définit le format bidirectionnel des propriétés d'une propriété de configuration de connecteur spécifique, pour une connexion spécifique à l'EIS.
BiDiContextCP<property_Name>SpecialFormat	String (Chaîne)	Indicateur servant à contrôler l'appel du support bidirectionnel au niveau de la propriété de configuration du connecteur. Si la valeur true est définie, l'appel du support n'a pas lieu. Si elle vaut false, le support est appelé.

Propriété BiDi	Type	Description
BiDiContextCP<property_Name>Skip	Boolean (booléen)	Définit le format spécial des propriétés d'une propriété de configuration de connecteur spécifique, pour une connexion spécifique à l'EIS.

Tâches associées

«Configuration de l'adaptateur sur le serveur», à la page 54

Une fois que le fichier EAR (Enterprise Application Archive) du projet d'adaptateur a été installé sur le serveur d'applications, vous pouvez, si vous le souhaitez, reconfigurer les propriétés utilisées par l'adaptateur pour configurer un canal de communication vers une application de base de données spécifique. Vous pouvez ensuite démarrer l'application d'adaptateur configurée.

Propriétés de connexion

Les propriétés de connexion permettent à l'assistant ESD (Enterprise Service Discovery) de se connecter à l'instance de système EIS (Enterprise Information System) cible.

Tâches associées

«Déploiement et configuration du scénario 2», à la page 69

Dans le scénario 2, vous définissez les propriétés de connexion de l'adaptateur et générez des objets métier. Vous exportez votre projet dans un fichier EAR (Enterprise Application Archive), déployez le projet sur le serveur d'applications et réinitialisez les propriétés de configuration.

Propriétés de connexion à la fonction de recherche des métadonnées

Le processus ESD (Enterprise Service Discovery) requiert ces propriétés pour se connecter au système EIS (Enterprise Information System) afin de rechercher et créer la description de service.

A l'aide de ces propriétés, le service de reconnaissance prépare une arborescence des métadonnées à afficher pour la sélection et l'exploration des objets.

Le tableau intitulé «Propriétés de connexion à la fonction de recherche des métadonnées» décrit ces propriétés.

Propriétés de connexion à la fonction de recherche des métadonnées

Propriété	Obligatoire	Globalisée	Description
URL de la base de données	Oui	Oui	URL de la base de données cible de la connexion. Cette URL peut varier d'un pilote à un autre. Spécifiez cette propriété conformément aux instructions du pilote. Cette propriété est activée pour une utilisation avec les langues bidirectionnelles.

Propriété	Obligatoire	Globalisée	Description
JDBCDriverClass	Oui	Non	Nom du pilote de base de données utilisé pour se connecter à la base de données.
Password	Oui	Oui	Mot de passe de l'utilisateur correspondant. Cette propriété est activée pour une utilisation avec les langues bidirectionnelles.
Prefix	Non	Oui	Préfixe à ajouter au nom de l'objet métier. Cette propriété est activée pour une utilisation avec les langues bidirectionnelles.
UserName	Oui	Oui	Nom d'utilisateur pour la connexion à la base de données. Cette propriété est activée pour une utilisation avec les langues bidirectionnelles.

Tâches associées

«Définition des propriétés de connexion», à la page 44

Une fois que vous avez créé le projet d'adaptateur, vous devez initialiser l'assistant ESD (Enterprise Service Discovery) pour l'adaptateur pour JDBC et définir les valeurs des propriétés de connexion de votre instance de base de données.

Propriétés de connexion bidirectionnelles

Ces propriétés permettent à l'assistant ESD (Enterprise Service Discovery) d'appliquer la conversion bidirectionnelle appropriée aux données transmises au système EIS (Enterprise Information System).

Le tableau ci-après répertorie les propriétés bidirectionnelles utilisées lors de la configuration des connexions.

Propriétés de connexion bidirectionnelles

Propriété	Valeur	Position de la lettre	Description
BiDi OrderingSchema	Implicite Visuel	1	Logique (Implicite). La valeur par défaut est Implicite. Visuel
BiDi Direction	GàD DàG contextuel_GàD contextuel_DàG	2	De gauche à droite. GàD (L) est la valeur par défaut. De droite à gauche Contextuel de gauche à droite Contextuel de droite à gauche

Propriété	Valeur	Position de la lettre	Description
Permutation symétrique bidirectionnelle	Oui Non	3	La permutation symétrique est activée. Oui (Y) est la valeur par défaut. La permutation symétrique est désactivée
BiDi Shaping	nominal mis en forme initial intermédiaire final isolé	4	Nominal (N). Il s'agit de la valeur par défaut. Lettre mise en forme suivant sa position. Lettre dans sa mise en forme initiale. Lettre dans sa mise en forme intermédiaire. Lettre dans sa mise en forme finale. Lettre dans sa mise en forme isolée.
BiDi NumericShaping	nominal national contextuel	5	Nominal (N). Il s'agit de la valeur par défaut. Hindi (National) Contextuel

Tâches associées

«Définition des propriétés de connexion», à la page 44

Une fois que vous avez créé le projet d'adaptateur, vous devez initialiser l'assistant ESD (Enterprise Service Discovery) pour l'adaptateur pour JDBC et définir les valeurs des propriétés de connexion de votre instance de base de données.

Propriétés de sélection des objets

Ces propriétés sont utilisées lors du processus de sélection des objets métier. Elles incluent les propriétés de filtre de noeud et de sélection.

Propriétés de filtre et de noeud

Lors de la reconnaissance des objets métier, vous pouvez spécifier des propriétés de filtre si vous souhaitez restreindre la liste des schémas affichés dans l'arborescence ; sinon, tous les schémas sont affichés. Vous pouvez définir la propriété de noeud pour restreindre la liste des objets de base de données à afficher.

Le tableau intitulé «Propriétés de filtre» décrit les propriétés à utiliser pour filtrer les schémas.

Propriétés de filtre

Propriété de filtre	Type	Description
SchemaNameFilter	String (Chaîne)	Chaîne utilisée pour filtrer les schémas. Seuls les schémas commençant par la chaîne spécifiée sont affichés. Si cette propriété n'est pas définie, tous les schémas sont affichés.

Propriété de filtre	Type	Description
Types	Multi-value (valeurs multiples)	Répertorie les types d'objet de base de données : tables, vues, procédures stockées et synonymes/noms d'usage. Les valeurs sont définies par EMD (Enterprise Metadata Discovery). Vous pouvez sélectionner plusieurs valeurs.
DefineASI	Boolean (booléen)	Si la valeur est true, lorsque vous sélectionnez un objet pour la génération, vous êtes invités à définir les informations ASI de l'objet métier.

Le tableau intitulé «Propriété de noeud» décrit la propriété à utiliser pour filtrer les objets de base de données.

Propriété de noeud

Propriété de noeud	Type	Description
ObjectNameFilter	String (Chaîne)	Chaîne utilisée pour filtrer les objets de base de données. Seuls les objets de base de données commençant par le texte spécifié sont affichés. Si aucune valeur n'est spécifiée pour le paramètre ObjectNameFilter, tous les objets sont affichés.

Concepts associés

«Recherche des fonctionnalités du système», à la page 38

L'adaptateur recherche des objets métier dans une base de données en analysant cette dernière pour identifier les schémas. Il crée ensuite une liste de tous les objets de la base de données et l'affiche sous forme d'arborescence.

Propriétés de sélection

Une fois que vous avez sélectionné les objets de base de données, vous devez définir les valeurs des propriétés de sélection.

L'assistant ESD (Enterprise Service Discovery) interroge les propriétés de sélection. Le tableau intitulé «Propriétés de sélection» décrit ces propriétés.

Propriétés de sélection

Propriété de sélection	Type	Description
BOLocation	String (Chaîne)	Chemin d'accès à l'emplacement dans lequel les fichiers .xsd générés doivent être stockés. Défini comme le chemin d'accès relatif dans la description des données.
MaxRecords	Integer (nombre entier)	Nombre maximal d'enregistrements à extraire pour une opération RetrieveALL.
NameSpace	String (Chaîne)	Texte utilisé pour filtrer les schémas.

Propriété de sélection	Type	Description
Operations	Multi-value (valeurs multiples)	Il s'agit par défaut d'une liste des opérations prises en charge par l'adaptateur pour le type de service sélectionné. Sélectionnez une opération dans cette liste. Les opérations spécifiées sont définies pour tous les objets métier générés.
ServiceType	String (Chaîne)	Le type sélectionné peut être le type entrant ou sortant. Les valeurs sont définies par EMD (Enterprise Metadata Discovery).

La propriété Namespace correspond par défaut à l'espace de nom par défaut de tous les objets métier. Il s'agit de l'espace de nom <http://www.ibm.com/xmlns/prod/websphere/j2ca/jdbc>.

L'espace de nom précède le nom de l'objet métier pour que les schémas d'objet métier restent logiquement séparés. Par exemple : <http://www.ibm.com/xmlns/prod/websphere/j2ca/jdbc/Schema1Customer>.

Concepts associés

«Sélection et génération d'objets», à la page 38

Pour générer des objets métier, vous devez sélectionner des noeuds d'objet de base de données. L'assistant ESD (Enterprise Service Discovery) génère ensuite des objets métier pour les objets des noeuds sélectionnés.

Tâches associées

«Définition des propriétés de sélection», à la page 49

Une fois que vous avez sélectionné les objets de base de données, vous devez spécifier les valeurs des propriétés de sélection pour les fichiers d'importation et d'exportation.

Remarques

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 ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM EMEA Director of Licensing
IBM Europe Middle-East Africa
Tour Descartes
La Défense 5
2, avenue Gambetta
92066 - Paris-La Défense CEDEX
France

Pour le Canada, veuillez adresser votre courrier à :

IBM Director of Commercial Relations
IBM Canada Ltd.
3600 Steeles Avenue East
Markham, Ontario
L3R 9Z7
Canada

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". IBM DECLINE TOUTE RESPONSABILITE, EXPLICITE OU IMPLICITE, RELATIVE AUX INFORMATIONS QUI Y SONT CONTENUES, Y COMPRIS EN CE QUI CONCERNE LES GARANTIES DE VALEUR MARCHANDE OU D'ADAPTATION A VOS BESOINS. 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. Il 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 logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Corporation
577 Airport Blvd., Suite 800
Burlingame, CA 94010
Etats-Unis

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.

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 DE COPYRIGHT :

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.

Si vous visualisez ces informations en ligne, il se peut que les photographies et illustrations en couleur n'apparaissent pas à l'écran.

Informations sur les interfaces de programmation

Les informations sur les interfaces de programmation ont pour objectif de vous aider à créer des logiciels d'application 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.

Cependant, ces informations peuvent également contenir des informations sur le diagnostic, la modification et le réglage. Ces informations vous permettent d'exécuter le débogage de votre logiciel d'application.

Avertissement : N'utilisez pas les informations de diagnostic, de modification et d'optimisation en guise d'interface de programmation car elles peuvent être modifiées sans préavis.

Marques et marques de service

Les termes qui suivent sont des marques d'International Business Machines Corporation aux Etats-Unis et/ou dans certains autres pays :

i5/OS
IBM
le logo IBM
AIX
AIX 5L
CICS
CrossWorlds
DB2
DB2 Universal Database
Domino
HelpNow
IMS
Informix
iSeries
Lotus
Lotus Notes
MQIntegrator
MQSeries

MVS
Remarques
OS/400
Passport Advantage
pSeries
Redbooks
SupportPac
WebSphere
z/OS

Java ainsi que tous les logos et toutes les marques incluant Java sont des marques de Sun Microsystems, Inc. aux Etats-Unis et/ou dans certains autres pays.

Microsoft, Windows, Windows NT et le logo Windows sont des marques de Microsoft Corporation aux Etats-Unis et/ou dans certains autres pays.

Intel, le logo Intel, Intel Inside, le logo Intel Inside, Intel Centrino, le logo Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium et Pentium sont des marques d'Intel Corporation ou de ses filiales aux Etats-Unis et/ou dans d'autres pays.

UNIX est une marque enregistrée de The Open Group 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.

Les autres noms de sociétés, de produits et de services peuvent appartenir à des tiers.

Ce produit inclut des composants logiciels développés par le projet Eclipse (<http://www.eclipse.org/>).



Adaptateurs WebSphere, Version 6.0

IBM