

IBM WebSphere Commerce



# Guide d'utilisation de Catalog Manager

*Version 54*



IBM WebSphere Commerce



# Guide d'utilisation de Catalog Manager

*Version 54*

**Important**

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations figurant à la section «Remarques» à la page 117.

**Première édition – mai 2002**

LE PRESENT DOCUMENT EST LIVRE "EN L'ETAT". IBM DECLINE TOUTE RESPONSABILITE, EXPRESSE OU IMPLICITE, RELATIVE AUX INFORMATIONS QUI Y SONT CONTENUES, Y COMPRIS EN CE QUI CONCERNE LES GARANTIES DE QUALITE 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.

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 2002. Tous droits réservés.

© **Copyright International Business Machines Corporation 2002. All rights reserved.**

# Table des matières

|  |            |
|--|------------|
| <b>Avant de commencer</b> . . . . .                    | <b>vii</b> |
| Conventions utilisées dans le présent manuel . . . . . | vii        |
| A qui est destiné ce manuel . . . . .                  | viii       |
| Où trouver plus d'informations . . . . .               | viii       |

## **Partie 1. Présentation de Catalog Manager** . . . . . **1**

## **Partie 2. Transformation, chargement et extraction de données** . . . . . **3**

|   |          |
|---|----------|
| <b>Chapitre 1. Introduction</b> . . . . . | <b>5</b> |
| Utilitaires . . . . .                     | 7        |
| Outils administratifs . . . . .           | 8        |

### **Chapitre 2. Transformation de texte** . . . . . **9**

|   |    |
|---|----|
| Lancement de l'outil de transformation de texte . . . . .   | 10 |
| Utilisation de la vue d'édition de schéma de texte . . . . .  | 11 |
| Création d'un fichier schéma . . . . .  | 11 |
| Ouverture d'un fichier schéma . . . . .   | 11 |
| Sauvegarde d'un fichier schéma . . . . .  | 11 |
| Modification d'un fichier schéma . . . . .  | 12 |
| Ajout d'un élément . . . . .  | 12 |
| Suppression d'un élément . . . . .  | 12 |
| Remplacement d'un élément . . . . .   | 12 |
| Déplacement d'un élément d'une ligne vers le haut . . . . .   | 12 |
| Déplacement d'un élément d'une ligne vers le bas . . . . .  | 12 |
| Ajout d'un attribut . . . . .   | 12 |
| Suppression d'un attribut . . . . .   | 12 |
| Remplacement d'un attribut . . . . .  | 12 |
| Déplacement d'un attribut d'une ligne vers le haut . . . . .  | 12 |
| Déplacement d'un attribut d'une ligne vers le bas . . . . .   | 13 |
| Modification de la structure d'un fichier schéma . . . . .  | 13 |
| Préparation d'un fichier schéma pour convertir des données d'un format XML en un format variable délimité par caractère . . . . . | 13 |
| Utilisation de la vue d'édition de commande de transformation . . . . .   | 14 |
| Création d'une instruction . . . . .  | 14 |
| Suppression d'une instruction . . . . .   | 14 |
| Modification ou remplacement d'une instruction . . . . .  | 14 |
| Déplacement d'une instruction d'une ligne vers le haut . . . . .  | 14 |
| Déplacement d'une instruction d'une ligne vers le bas . . . . .   | 15 |
| Effacement d'une instruction . . . . .  | 15 |
| Utilisation de la vue du processus de transformation . . . . .  | 15 |

### **Chapitre 3. Transformation de données XML** . . . . . **17**

|  |    |
|--|----|
| Lancement de l'éditeur XSL Editor . . . . .                                | 18 |
| Gestion des zones de constitution de règle de mappage . . . . .            | 19 |
| Création d'une zone de constitution de règle de mappage . . . . .          | 19 |
| Modification d'une zone de constitution de règle de mappage . . . . .      | 19 |
| Suppression d'une zone de constitution de règle de mappage . . . . .       | 19 |
| Utilisation de l'éditeur XSL . . . . .                                     | 20 |
| Création d'une règle de mappage . . . . .                                  | 20 |
| Mappage d'un élément sur un élément . . . . .                              | 20 |
| Mappage d'un attribut sur un attribut . . . . .                            | 20 |
| Création d'une expression de mappage personnalisée . . . . .               | 20 |
| Suppression d'une règle de mappage . . . . .                               | 21 |
| Exécution d'une transformation XML . . . . .                               | 21 |
| Personnalisation de la table des règles de mappage . . . . .               | 21 |
| Affichage d'une expression de valeur ou d'une règle XSL complète . . . . . | 22 |

### **Chapitre 4. Génération d'un fichier DTD et d'un schéma** . . . . . **23**

|   |    |
|---|----|
| Configuration de DTD Generator . . . . .                      | 23 |
| Génération d'un fichier DTD . . . . .                         | 24 |
| Génération d'un schéma et d'un fichier XML détaillé . . . . . | 26 |

### **Chapitre 5. Résolution d'identificateurs** **27**

|   |    |
|---|----|
| Configuration de l'utilitaire ID Resolver . . . . .                                     | 27 |
| Définition de la façon dont ID Resolver gère les horodatages . . . . .                  | 28 |
| Définition de la façon dont ID Resolver gère les archivages . . . . .                   | 28 |
| Définition de la façon dont ID Resolver gère les pilotes de base de données . . . . .   | 29 |
| Détermination du mode de traitement des données . . . . .                               | 29 |
| Choix de la méthode de chargement . . . . .   | 30 |
| Choix de la méthode de mise à jour . . . . .  | 30 |
| Choix de la méthode mixte . . . . .   | 31 |
| Utilisation des techniques de résolution d'ID . . . . .                                 | 31 |
| Spécification d'un fichier de propriétés à l'aide de l'utilitaire ID Resolver . . . . . | 31 |
| Utilisation d'un fichier de propriétés pour générer des identificateurs . . . . .       | 32 |
| Utilisation d'un fichier de propriétés avec clés composées . . . . .                    | 34 |
| Utilisation d'un fichier de propriétés avec clés primaires en cascade . . . . .         | 35 |
| Utilisation de la résolution d'alias interne . . . . .                                  | 36 |
| Exemple partiel d'utilisation de la résolution d'ID d'alias interne . . . . .           | 36 |

|   |    |
|---|----|
| Utilisation de la résolution d'index à entrées<br>uniques . . . . .       | 37 |
| Exemple partiel de résolution d'index à<br>entrées uniques . . . . .      | 37 |
| Chargement de données dans la table MEMBER . . . . .                      | 38 |
| Création d'une relation externe à l'aide de la table<br>REFKEYS . . . . . | 39 |
| Identification et résolution des erreurs . . . . .                        | 40 |

## **Chapitre 6. Chargement de données . . . 41**

|   |    |
|---|----|
| Configuration du module Loader . . . . .  | 41 |
| Non prise en compte des éléments du fichier<br>d'entrée . . . . .   | 42 |
| Insertion de la valeur NULL dans une colonne  | 42 |
| Chargement de données d'horodatage et de date   | 42 |
| Chargement des horodatages en cours . . . . .   | 43 |
| Exemple de chargement d'horodatages en<br>cours . . . . .   | 45 |
| Exemples d'ajout de durées à des horodatages<br>en cours . . . . .  | 45 |
| Gestion de files d'attente d'événements . . . . .   | 46 |
| Exécution avec différents logiciels de base de<br>données et systèmes d'exploitation . . . . .              | 47 |
| Substitution d'un composant . . . . .   | 50 |
| Utilisation de la synchronisation de l'espace de<br>recherche de l'Assistant produit . . . . .              | 50 |
| Personnalisation de la synchronisation de<br>l'espace de recherche de l'Assistant produit. . . . .          | 54 |
| Détermination de la façon de traiter les données en<br>utilisant le programme de chargement Loader. . . . . | 55 |
| Choix de la méthode de chargement . . . . .   | 55 |
| Choix de la méthode d'importation . . . . .   | 55 |
| Choix de la méthode d'importation SQL. . . . .  | 56 |
| Autres considérations . . . . .   | 56 |
| Chargement de documents volumineux . . . . .  | 57 |
| Conseils d'identification et de résolution des<br>incidents . . . . .                                       | 58 |

## **Chapitre 7. Extraction de données . . . 59**

|  |    |
|--|----|
| Création d'un filtre d'extraction . . . . .      | 59 |
| Configuration de l'utilitaire Extractor. . . . . | 62 |

## **Chapitre 8. Utilisation du programme de consignment du module Loader . . . 63**

|   |    |
|---|----|
| Configuration de la journalisation dans votre<br>environnement pour les systèmes Windows NT,<br>Windows 2000, AIX, Linux et Solaris . . . . . | 63 |
| Exemple de définition de la variable<br>CLASSPATH . . . . .   | 63 |
| Exemple de spécification de la propriété système<br>com.ibm.wca.logging.configFile. . . . .   | 63 |
| Personnalisation de la journalisation pour le module<br>Loader . . . . .  | 64 |
| Questionnaires . . . . .  | 65 |
| Filtres . . . . .   | 66 |
| Formats . . . . .   | 67 |
| Exemple : WCALoggerConfig.xml et WCALogger.dtd  | 67 |
| WCALoggerConfig.xml . . . . .   | 67 |
| WCALogger.dtd . . . . .   | 68 |

## **Chapitre 9. Utilisation de la fonction de signalisation des erreurs du module Loader . . . . . 71**

## **Chapitre 10. Configuration des instructions et scripts du module Loader . . . . . 73**

---

## **Partie 3. Utilisation de Web editor 75**

### **Chapitre 11. Configuration de Web editor . . . . . 77**

|   |    |
|---|----|
| Configuration de Web editor . . . . .   | 78 |
| Modification du fichier webeditor.properties. . . . .   | 78 |
| Changement d'emplacement des fichiers<br>temporaires . . . . .                                | 80 |
| Création d'un fichier de description de formulaire<br>XML à l'aide de DTD Generator . . . . . | 80 |
| Personnalisation de la description de formulaire<br>XML. . . . .                              | 82 |
| Modification des noms de formulaire. . . . .  | 85 |
| Modification d'une description de zone . . . . .  | 85 |
| Ajout d'un menu déroulant . . . . .   | 86 |
| Ajout d'aide sur les zones . . . . .  | 87 |
| Personnalisation des résultats de la recherche et<br>de la liste de travail . . . . .         | 87 |
| Modification du fichier weProcessList . . . . .   | 90 |
| Modification du fichier webeditor.xml . . . . .   | 91 |

### **Chapitre 12. Gestion des catalogues . . . 93**

|   |    |
|---|----|
| Ajout d'un enregistrement à une table à l'aide de<br>Web editor . . . . .           | 93 |
| Modification d'un enregistrement dans une table à<br>l'aide de Web editor . . . . . | 94 |
| Suppression d'un enregistrement d'une table à l'aide<br>de Web editor . . . . .     | 95 |

---

## **Partie 4. Informations de référence sur les instructions . . . . . 97**

### **Chapitre 13. Instruction DTD Generate 99**

|   |     |
|---|-----|
| Instruction DTD Generate pour Windows, AIX,<br>Linux et Solaris . . . . . | 99  |
| Instruction DTD Generate pour iSeries . . . . .                           | 100 |

### **Chapitre 14. Instruction Extract. . . . 103**

|   |     |
|---|-----|
| Instruction Extract pour Windows, AIX, Linux et<br>Solaris. . . . . | 103 |
| Instruction Extract pour iSeries . . . . .                          | 104 |

### **Chapitre 15. Instruction ID Resolve 105**

|   |     |
|---|-----|
| Instruction ID Resolve pour Windows, AIX, Linux<br>et Solaris . . . . . | 105 |
| Instruction ID Resolve pour iSeries . . . . .                           | 107 |

### **Chapitre 16. Instruction Load . . . . 109**

|  |     |
|--|-----|
| Instruction Load pour Windows, AIX, Linux et Solaris . . . . . | 109 |
| Instruction Load pour iSeries . . . . .                        | 111 |

**Chapitre 17. Instruction Text Transform . . . . . 113**

|  |     |
|--|-----|
| Instruction Text Transform pour Windows, AIX, Linux et Solaris . . . . . | 113 |
| Instruction Text Transform pour iSeries . . . . .                        | 114 |

**Chapitre 18. Instruction XML Transform . . . . . 115**

|   |     |
|---|-----|
| Instruction XML Transform pour Windows, AIX, Linux et Solaris . . . . . | 115 |
| Instruction XML Transform pour iSeries . . . . .                        | 116 |

**Remarques . . . . . 117**

|                   |     |
|-------------------|-----|
| Marques . . . . . | 120 |
|-------------------|-----|



---

## Avant de commencer

Le *Guide d'utilisation d'IBM WebSphere Commerce Catalog Manager* fournit des informations sur WebSphere Commerce Catalog Manager. Il traite plus particulièrement des sujets suivants :

- Transformation, chargement et extraction de données à l'aide des utilitaires et des outils de Catalog Manager
- Utilisation de Catalog Manager Web editor pour gérer les données des catalogues
- Instructions de Catalog Manager

---

## Conventions utilisées dans le présent manuel

Le présent manuel respecte les conventions suivantes :

|   |  |
|---|--|
| <b>Police de caractères gras</b>  | indique des instructions ou des éléments d'interface graphique, tels que des noms de zone, des boutons ou des options de menu.       |
| Police non proportionnelle  | indique des exemples de texte que vous devez entrer tels quels, comme des noms de fichier ou des chemins de répertoire, par exemple. |
| <i>Police de caractères italiques</i>   | indique des éléments mis en évidence et des variables que vous pouvez remplacer par vos propres valeurs.                             |
|   | indique des informations spécifiques de WebSphere Commerce pour Windows.   |
|  | indique des informations spécifiques de WebSphere Commerce pour Windows NT.  |
|  | indique des informations spécifiques de WebSphere Commerce pour Windows 2000.  |
|  | indique des informations spécifiques de WebSphere Commerce pour AIX.   |
|  | indique des informations spécifiques de WebSphere Commerce pour Solaris.   |
|  | indique des informations spécifiques de WebSphere Commerce pour Linux.   |
|  | indique des informations spécifiques de WebSphere Commerce pour IBM @server iSeries (précédemment dénommé AS/400).                   |
|  | indique des informations spécifiques de la base de données universelle DB2.  |
|  | indique des informations spécifiques de la base de données Oracle.   |
|  | indique des informations spécifiques de WebSphere Commerce Business Edition.   |
|  | indique des informations spécifiques de WebSphere Commerce Professional Edition.   |

---

## A qui est destiné ce manuel

Le présent manuel est destiné aux développeurs de sites, aux administrateurs et aux collaborateurs qui doivent savoir utiliser WebSphere Commerce Catalog Manager, ainsi qu'aux autres utilisateurs professionnels qui doivent comprendre son fonctionnement.

Ce guide s'adresse en particulier aux développeurs de magasins et aux administrateurs de sites WebSphere Commerce qui doivent savoir utiliser les diverses fonctions de WebSphere Commerce Catalog Manager.

Les développeurs de magasins et administrateurs de sites WebSphere Commerce qui réalisent des extensions de programme doivent avoir de solides connaissances dans les domaines suivants :

- Technologie des bases de données
- Architecture de composants Enterprise JavaBeans
- Langage HTML
- Java
- Technologie JavaServer Pages
- VisualAge for Java, Enterprise Edition, version 3.5 ou suivante
- Langage XML (Extensible Markup Language)

Les gestionnaires de catégories et les responsables produits doivent également lire le présent manuel pour apprendre comment communiquer les demandes de catalogues, de produits et d'articles aux développeurs de magasins. Ils peuvent également apprendre dans ce manuel à utiliser Web editor pour mettre à jour le catalogue du magasin.

---

## Où trouver plus d'informations

Le présent guide d'utilisation est disponible au format Adobe PDF sur le site Web de WebSphere Commerce. Veuillez consulter le site Web WebSphere Commerce Technical Library pour obtenir les dernières versions du présent document, ainsi que d'autres informations sur WebSphere Commerce :

-  [http://www.ibm.com/software/webservers/commerce/wc\\_be/lit-tech-general.html](http://www.ibm.com/software/webservers/commerce/wc_be/lit-tech-general.html)
-  [http://www.ibm.com/software/webservers/commerce/wc\\_pe/lit-tech-general.html](http://www.ibm.com/software/webservers/commerce/wc_pe/lit-tech-general.html)

---

## Partie 1. Présentation de Catalog Manager

WebSphere Commerce Catalog Manager offre des outils de développement génériques qui remplissent diverses fonctions pouvant être regroupées dans l'ordre voulu pour résoudre des difficultés particulières en matière de gestion de catalogues. Il est en outre suffisamment flexible pour pouvoir traiter les personnalisations apportées au schéma WebSphere Commerce.

Catalog Manager permet de regrouper des informations provenant de plusieurs sources sur un seul système WebSphere Commerce consolidé et de remapper toutes ces données en un format de définition de catalogue et de produit standard en utilisant des fichiers XML comme mode de gestion des informations standard.

Catalog Manager vous donne les moyens d'effectuer les tâches suivantes :

- Importation de données provenant de diverses sources, sous forme de fichiers ASCII et XML, sur le serveur WebSphere Commerce
- Conversion des données du format ASCII au format XML et inversement
- Remappage des données d'un format XML à un autre
- Regroupement de données provenant de divers flux en entrée dans une seule base de données globale
- Création, modification et suppression de données via une interface de navigateur Web

Catalog Manager comprend les composants suivants :

- Module Loader

Ce module se compose principalement d'utilitaires destinés à préparer et charger des données dans une base de données WebSphere Commerce. Vous pouvez utiliser le module Loader pour charger de grandes quantités de données et pour mettre à jour des données dans votre base de données WebSphere Commerce.

Le module Loader vous permet également d'effectuer les tâches suivantes :

- Extraction des données d'une base de données sous la forme d'un document XML
  - Conversion des données XML dans d'autres formats XML
  - Conversion des données d'un format variable délimité par caractère en un format de données XML
- Outils administratifs de Catalog Manager  
Catalog Manager comprend également les deux outils suivants, dotés d'une interface utilisateur facilitant l'administration de ses fonctions :
    - Outil de transformation de texte
    - XSL Editor
  - Catalog Manager Web editor  
Web editor vous permet de créer ou supprimer des données de catalogue, ou encore d'apporter des modifications à ces dernières, via un navigateur Web.



---

## **Partie 2. Transformation, chargement et extraction de données**



---

## Chapitre 1. Introduction

Le module Loader de Catalog Manager se compose principalement d'utilitaires destinés à préparer et charger des données dans une base de données WebSphere Commerce. Vous pouvez utiliser le module Loader pour charger de grandes quantités de données et pour mettre à jour des données dans votre base de données WebSphere Commerce. Le module Loader vous permet également d'effectuer les tâches suivantes :

- Extraction des données d'une base de données sous la forme d'un document XML
- Conversion des données XML dans d'autres formats XML
- Conversion des données d'un format variable délimité par caractère en un format de données XML

Catalog Manager comprend également les deux outils suivants, dotés d'une interface utilisateur facilitant l'administration de ses fonctions :

- Outil de transformation de texte
- XSL Editor

---

## Utilitaires

Le module WebSphere Commerce Loader se compose d'utilitaires destinés à préparer et charger des données dans une base de données WebSphere Commerce. Vous pouvez utiliser ces utilitaires pour charger des quantités de données variables et pour mettre à jour des données dans votre base de données WebSphere Commerce.

Le processus de chargement comprend les étapes nécessaires pour placer des données dans votre base de données WebSphere Commerce :

1. Génération d'un fichier DTD à l'aide de l'utilitaire de génération de DTD (DTD Generator)
2. Résolution d'identificateurs dans les fichiers d'entrée à l'aide de l'utilitaire de résolution d'ID (ID Resolver)
3. Chargement des données à l'aide de l'utilitaire de chargement (Loader)

Le module Loader comprend également des utilitaires destinés à extraire des données d'une base de données sous forme de document XML et à assurer la conversion des données XML dans d'autres formats XML.

Le module Loader se compose des utilitaires suivants :

- **Text Transformer**

Text Transformer convertit les données d'un format variable délimité par caractère en un format de données XML.

Pour plus d'informations, reportez-vous au Chapitre 2, «Transformation de texte» à la page 9.

- **XML Transformer**

XML Transformer modifie, regroupe et remet en correspondance les données d'un document XML avec d'autres formats XML afin de permettre à d'autres utilisateurs ou systèmes d'utiliser ces données au besoin.

Pour plus d'informations, reportez-vous au Chapitre 3, «Transformation de données XML» à la page 17.

- **DTD Generator**

La définition de type de document (DTD) spécifie les éléments structurels ainsi que les définitions de code de formatage qui peuvent être utilisés dans le fichier de données XML. Par exemple, un fichier DTD peut répertorier des éléments à utiliser dans un document et spécifier les attributs pouvant être associés à chacun d'eux.

DTD Generator est un utilitaire qui crée un fichier DTD devant être utilisé par le module Loader en fonction du schéma de base de données. Ce fichier DTD décrit les tables et colonnes dans lesquelles le module Loader peut importer des données. DTD Generator peut également créer un schéma et un document XML détaillé à utiliser avec Catalog Manager Web editor.

DTD Generator génère un fichier DTD en fonction de la base de données cible à laquelle vos données doivent être conformes. Ce fichier DTD est utilisé tout au long du processus de chargement. DTD Generator ne doit être exécuté qu'une seule fois.

Pour plus d'informations, reportez-vous au Chapitre 4, «Génération d'un fichier DTD et d'un schéma» à la page 23.

- **ID Resolver**

ID Resolver est un utilitaire qui génère des identificateurs pour les éléments XML qui en ont besoin. Si votre contenu XML fournit déjà des identificateurs, il est inutile d'exécuter ID Resolver.

ID Resolver met à jour un jeu d'éléments XML avec leurs identificateurs associés. Cette étape est essentielle, puisque les fichiers XML du module Loader sont directement mis en correspondance avec le schéma de la base de données cible. A ce titre, ils doivent disposer d'identificateurs.

ID Resolver comprend une fonction de signalisation des erreurs, qui génère un document d'exception en cas d'erreur.

Pour plus d'informations, reportez-vous au Chapitre 5, «Résolution d'identificateurs» à la page 27.

- **Loader**

Le module Loader utilise un fichier XML valide et syntaxiquement correct comme entrée pour charger des données dans la base de données. Les éléments du document XML sont mappés sur les noms de table de la base de données et les attributs d'éléments sur des colonnes. Le module Loader est le moyen le plus répandu de charger des données dans un système.

Il permet de mettre à jour une table au niveau des colonnes. Il vous permet également de supprimer des données d'une base de données.

L'exemple ci-dessous représente un extrait d'une entrée XML valide et syntaxiquement correcte dans le module Loader :

```
<ADDRBOOK
  ADDRBOOK_ID="11801"
  MEMBER_ID="100"
  DISPLAYNAME="Amis"
  DESCRIPTION="Tous mes amis"
  TYPE="P"
/>
```

Dans l'exemple ci-dessus, ADDRBOOK est le nom de la table et les colonnes à mettre à jour sont indiquées par les attributs de l'élément ADDRBOOK.

Le module Loader comprend les fonctions suivantes :

– **Signalisation des erreurs**

Le module Loader comprend une fonction de signalisation des erreurs, qui génère un document d'exception en cas d'erreur.

– **Synchronisation des espaces de recherche de l'Assistant produit**

Les espaces de recherche de l'Assistant produit sont créés par extraction d'informations du catalogue WebSphere Commerce et présentation de ces informations sous une forme adaptée aux recherches. Les espaces de recherche et le catalogue ne sont plus synchronisés lorsque le catalogue est mis à jour. Pour éviter tout retard dans la synchronisation des espaces de recherche et du catalogue, vous pouvez activer la fonction de synchronisation des espaces de recherche de l'Assistant produit, offerte par le module Loader.

Lorsque cette fonction est activée, l'utilitaire de chargement (Loader) planifie les instructions de type search-space appropriées en ajoutant des informations d'instruction à la table SCHCONFIG du planificateur de travaux, ainsi qu'à la table SCHSTATUS des instances du planificateur de travaux, comme indiqué ci-dessous.

| Table      | Mode        | Instruction Search-Space         | Condition   |
|------------|-------------|----------------------------------|---|
| CATENTRY   | Mise à jour | UpdateSearchSpaces               | Mise à jour d'indicateur de publication   |
|            | Suppression | RemoveProductFromAllSearchSpaces | Suppression de produit  |
| CATENTDESC | Ajout       | UpdateSearchSpaces               | Ajout d'une description/langue  |
|            | Mise à jour | UpdateSearchSpaces               | Mise à jour d'indicateur de publication<br>Mise à jour des données de description |
|            | Suppression | UpdateSearchSpaces               | Suppression d'une description/langue  |
| LISTPRICE  | Ajout       | UpdateSearchSpaces               | Ajout de prix catalogue   |
|            | Mise à jour | UpdateSearchSpaces               | Mise à jour de prix catalogue   |
|            | Suppression | UpdateSearchSpaces               | Suppression de prix catalogue   |
| ATTRVALUE  | Ajout       | UpdateSearchSpaces               | Ajout d'une valeur définie par l'utilisateur                                      |
|            | Mise à jour | UpdateSearchSpaces               | Mise à jour d'une valeur définie par l'utilisateur                                |
|            | Suppression | UpdateSearchSpaces               | Suppression d'une valeur définie par l'utilisateur                                |
| CATENTATTR | Ajout       | UpdateSearchSpaces               | Ajout d'attribut  |
|            | Mise à jour | UpdateSearchSpaces               | Mise à jour d'attribut  |
|            | Suppression | UpdateSearchSpaces               | Suppression d'attribut  |
| CATGPENREL | Ajout       | AddProductsToSearchSpace         | Ajout d'un produit à une catégorie  |
|            | Suppression | RemoveProductsFromSearchSpace    | Suppression d'un produit d'une catégorie  |

Pour plus d'informations, reportez-vous au Chapitre 6, «Chargement de données» à la page 41.

- **Extractor**

Extractor adresse une requête à une base de données pour extraire des sous-ensembles de données sélectionnés de la base de données et les placer dans un document XML. Les données à extraire de la base de données sont spécifiées à l'aide d'un document XML à filtre d'extraction.

Le fonctionnement d'Extractor est l'inverse de celui du module Loader. Vous utilisez Extractor pour extraire des sous-ensembles de données spécifiques de la base de données WebSphere Commerce sous forme de fichiers XML. Vous pouvez extraire des données sur des produits liés aux prochaines vacances, par exemple, ou bien extraire des informations d'une base de données consolidée pour les utiliser dans d'autres systèmes.

Pour plus d'informations, reportez-vous au Chapitre 7, «Extraction de données» à la page 59.

Le module Loader comprend également une fonction de journalisation. Chaque utilitaire du module Loader crée des messages pour signaler un succès, un échec ou une erreur, ainsi que pour fournir des informations de trace du programme.

---

## Outils administratifs

Catalog Manager comprend les outils suivants, destinés à faciliter l'administration de ses fonctions :

### **Outil de transformation de texte**

L'outil de transformation de texte aide l'administrateur à préparer les informations requises pour procéder à la conversion de données d'un format variable délimité par caractère en un format de données XML.

### **XSL Editor**

L'outil de transformation XML utilise le langage XSL pour définir les règles de transformation d'un fichier XML en un autre fichier XML. La fonction de mappage de l'éditeur XSL Editor offre à l'administrateur une interface visuelle permettant d'établir une association entre un élément d'un fichier DTD (définition de type de document) source et un élément d'un fichier DTD cible.

---

## Chapitre 2. Transformation de texte

Catalog Manager vous donne la possibilité de convertir des données en sortie au format ASCII provenant d'autres outils (tels que des tableurs) en données au format XML pouvant être entrées dans la base de données WebSphere Commerce.

L'outil de transformation de texte prépare les informations requises pour procéder à la conversion de données d'un format variable délimité par caractère en un format de données XML. Les vues suivantes sont fournies :

1. La vue d'édition de schéma de texte vous permet de créer et de modifier le fichier schéma XML à utiliser dans le cadre d'une transformation.
2. La vue d'édition de commande de transformation vous permet de créer et de modifier les instructions réelles utilisées pour exécuter le processus de transformation.
3. La vue du processus de transformation vous permet de lancer le processus de transformation.

## Lancement de l'outil de transformation de texte

Pour lancer l'outil de transformation de texte, utilisez le script ou l'instruction approprié, fourni dans le répertoire WebSphere Commerce suivant :

-  *unité*:\WebSphere\CommerceServer\bin\TextTrans.cmd
-  *unité*:\Program Files\WebSphere\CommerceServer\bin\TextTrans.cmd
-  /usr/WebSphere/CommerceServer/bin/texttrans.sh
-   /opt/WebSphere/CommerceServer/bin/texttrans.sh

 Dans les environnements iSeries, les administrateurs doivent copier au préalable les fichiers nécessaires à l'exécution de l'outil de transformation de texte sur leurs machines Windows NT ou Windows 2000. Voici un exemple de la procédure à suivre :

1. Créez un nouveau répertoire sur la machine Windows (*unité*:\TextTrans par exemple).
2. Créez les sous-répertoires suivants sous *unité*:\TextTrans :
  - \bin
  - \lib\loader
  - \wcsadmin
3. Copiez les fichiers de la machine iSeries vers ces répertoires, conformément au schéma suivant :

| De                         |        | Vers                     |
|----------------------------|--------|--------------------------|
| /QIBM/ProdData/WebCommerce |        | <i>unité</i> :\TextTrans |
| /bin                       |        | /bin                     |
| /TextTrans.cmd             | -----> | \TextTrans.cmd           |
| /lib/loader                |        | \lib\loader              |
| /wcmxmlp.jar               | -----> | \wcmxmlp.jar             |
| /wcmxslt.jar               | -----> | \wcmxslt.jar             |
| /wcsadmin                  |        | \wcsadmin                |
| /acsxml.ico                | -----> | \acsxml.ico              |
| /swing.jar                 | -----> | \swing.jar               |
| /TextTransformerUI.zip     | -----> | \TextTransformerUI.zip   |
| /TextTransform.cnt         | -----> | \TextTransform.cnt       |
| /TextTransform.hlp         | -----> | \TextTransform.hlp       |
| /TextTransform.reg         | -----> | \TextTransform.reg       |

4. Modifiez le fichier TextTrans.cmd sur la machine Windows en insérant le texte suivant :

```
set WCS_HOME=unité:\TextTrans
```

avant la ligne suivante :

```
set lib=%WCS_HOME%\lib\loader
```
5. Lancez l'outil de transformation de texte en exécutant TextTrans.cmd sur la machine Windows.

---

## Utilisation de la vue d'édition de schéma de texte

Les procédures suivantes concernent la création et la modification d'un fichier schéma XML en vue de la conversion de données d'un format variable délimité par caractère en un format XML.

### Création d'un fichier schéma

Pour créer un fichier schéma, procédez comme suit :

1. Lancez l'outil de transformation de texte.
2. Sélectionnez **Fichier > Nouveau**, ou cliquez sur l'icône Nouveau dans la barre d'outils.
3. Sélectionnez un chemin et saisissez le nom du fichier schéma XML à créer.

**Remarque :** Le nom de fichier par défaut est "Default.xml."

4. Sélectionnez **Format CSV**.

**Remarque :** Ne sélectionnez pas **Format de WebSphere Commerce Suite**. Cette option n'est utilisée qu'avec les versions antérieures de WebSphere Commerce.

5. Cliquez sur **OK**.

Vous pouvez maintenant créer de nouveaux éléments et attributs en suivant les procédures décrites ci-après.

### Ouverture d'un fichier schéma

Pour ouvrir un fichier schéma, procédez comme suit :

1. Lancez l'outil de transformation de texte.
2. Sélectionnez **Fichier > Ouvrir**, ou cliquez sur l'icône Ouvrir dans la barre d'outils.
3. Sélectionnez le fichier schéma à ouvrir.
4. Cliquez sur **OK**.

Vous pouvez maintenant modifier les éléments et attributs en suivant les procédures décrites ci-après.

### Sauvegarde d'un fichier schéma

Pour sauvegarder un fichier schéma, procédez comme suit :

1. Lancez l'outil de transformation de texte.
2. Pour sauvegarder les modifications que vous avez apportées au schéma, sélectionnez **Fichier > Sauvegarder** ou cliquez sur l'icône Sauvegarder dans la barre d'outils.
3. Pour sauvegarder une copie du schéma sous un nouveau nom, procédez comme suit :
  - a. Sélectionnez **Fichier > Sauvegarder sous**.
  - b. Sélectionnez un chemin et saisissez le nom du fichier XML à créer.
  - c. Cliquez sur **OK**.

## Modification d'un fichier schéma

Pour modifier un fichier schéma, ouvrez-le comme indiqué ci-avant et procédez comme suit :

### Ajout d'un élément

Pour ajouter un élément, procédez comme suit :

1. Saisissez un nouveau nom d'élément dans la zone Liste d'éléments.
2. Cliquez sur l'icône Ajouter ligne.

### Suppression d'un élément

Pour supprimer un élément, procédez comme suit :

1. Sélectionnez un nom d'élément.
2. Cliquez sur l'icône Supprimer ligne.

### Remplacement d'un élément

Pour remplacer un élément, procédez comme suit :

1. Saisissez un nouveau nom d'élément dans la zone Liste d'éléments.
2. Sélectionnez le nom de l'élément à remplacer.
3. Cliquez sur l'icône Remplacer ligne.

### Déplacement d'un élément d'une ligne vers le haut

Pour déplacer un élément d'une ligne vers le haut, procédez comme suit :

1. Sélectionnez le nom de l'élément à déplacer vers le haut.
2. Cliquez sur l'icône Déplacer la ligne vers le haut.

### Déplacement d'un élément d'une ligne vers le bas

Pour déplacer un élément d'une ligne vers le bas, procédez comme suit :

1. Sélectionnez le nom de l'élément à déplacer vers le bas.
2. Cliquez sur l'icône Déplacer la ligne vers le bas.

### Ajout d'un attribut

Pour ajouter un attribut, procédez comme suit :

1. Saisissez un nouveau nom d'attribut dans la zone Liste d'attributs.
2. Cliquez sur l'icône Ajouter ligne.

### Suppression d'un attribut

Pour supprimer un attribut, procédez comme suit :

1. Sélectionnez un nom d'attribut.
2. Cliquez sur l'icône Supprimer ligne.

### Remplacement d'un attribut

Pour remplacer un attribut, procédez comme suit :

1. Saisissez un nouveau nom d'attribut dans la zone Liste d'attributs.
2. Sélectionnez le nom de l'attribut à remplacer.
3. Cliquez sur l'icône Remplacer ligne.

### Déplacement d'un attribut d'une ligne vers le haut

Pour déplacer un attribut d'une ligne vers le haut, procédez comme suit :

1. Sélectionnez le nom de l'attribut à déplacer vers le haut.
2. Cliquez sur l'icône Déplacer la ligne vers le haut.

## Déplacement d'un attribut d'une ligne vers le bas

Pour déplacer un attribut d'une ligne vers le bas, procédez comme suit :

1. Sélectionnez le nom de l'attribut à déplacer vers le bas.
2. Cliquez sur l'icône Déplacer la ligne vers le bas.

## Modification de la structure d'un fichier schéma

La vue Structure de fichier figurant dans le volet inférieur de la vue d'édition de schéma de texte décrit la présentation d'un fichier au format variable délimité par caractère. Les zones suivantes sont obligatoires dans la structure de fichier prévue :

### Séparateur de zone :

Indique le délimiteur qui sépare les valeurs d'attribut. La valeur par défaut est une virgule (",").

### Séparateur d'enregistrement :

Indique le délimiteur qui sépare les enregistrements de données. La valeur par défaut est "&#10;&#13;" (cette valeur équivaut à une référence d'entité de type `\r\n`).

### Délimiteur de chaîne :

Indique le délimiteur qui précise les limites de début et de fin de la chaîne de caractères. La valeur par défaut est un guillemet simple (" ' ").

### En-tête inclus :

Valeur booléenne définie comme "true" si le fichier de données texte comprend une ligne d'en-tête et "false" dans le cas contraire. Si le fichier comprend une ligne d'en-tête, elle doit se conformer aux règles XML concernant les noms d'indicateur, car dans ce cas, l'en-tête sera utilisé comme nom d'indicateur XML. La valeur par défaut est "false".

### Nombre de lignes d'en-tête :

Indique le nombre de lignes d'en-tête existant dans le fichier de données texte. La valeur par défaut est zéro ("0").

## Préparation d'un fichier schéma pour convertir des données d'un format XML en un format variable délimité par caractère

Si vous utilisez un fichier schéma XML créé à l'aide de la vue d'édition de schéma de texte pour convertir des données d'un format XML en un format variable délimité par caractère, vous devez utiliser un éditeur de texte pour remplacer la spécification du type de données "Format CSV" par "Format XML" dans le fichier schéma avant d'effectuer la conversion.

---

## Utilisation de la vue d'édition de commande de transformation

La vue d'édition de commande de transformation vous permet de créer un fichier de commandes, d'ouvrir un fichier de commandes existant ou de sauvegarder les modifications apportées à un fichier de commandes. Le fichier de commandes par défaut est nommé "Manifest.txt."

Vous pouvez créer ou supprimer une instruction, remplacer une instruction par des informations modifiées ou changer l'ordre d'une instruction.

**Remarque :** Le fichier de commandes est automatiquement sauvegardé lorsque la table des instructions est mise à jour.

### Création d'une instruction

Pour créer une instruction, procédez comme suit :

1. Indiquez le fichier source, qui peut être un fichier au format variable délimité par caractère (avec l'extension .csv) ou un fichier au format XML (avec l'extension .xml).
2. Indiquez le nom du fichier schéma XML à utiliser pour la transformation.
3. Indiquez le nom du fichier de sortie devant être créé ou modifié au cours du processus de transformation (c'est-à-dire, le fichier dans lequel les nouvelles données seront stockées). Il peut s'agir d'un fichier au format variable délimité par caractère (avec l'extension .csv) ou d'un fichier au format XML (avec l'extension .xml).
4. Indiquez un mode commande.  
Sélectionnez **Créer** si le fichier de sortie doit être créé ou **Ajouter** s'il doit être ajouté à la fin d'un fichier de données existant.
5. Cliquez sur l'icône Ajouter ligne.

### Suppression d'une instruction

Pour supprimer une instruction, procédez comme suit :

1. Sélectionnez une instruction.
2. Cliquez sur l'icône Supprimer ligne.

### Modification ou remplacement d'une instruction

Pour modifier ou remplacer une instruction, procédez comme suit :

1. Cliquez sur l'icône Editer commande.  
Les zones d'entrée adéquates sont renseignées avec des lignes de données.
2. Modifiez le texte des zones d'entrée appropriées.
3. Cliquez sur l'icône Remplacer ligne pour mettre à jour les lignes de données.

### Déplacement d'une instruction d'une ligne vers le haut

Pour déplacer une instruction d'une ligne vers le haut, procédez comme suit :

1. Sélectionnez une instruction.
2. Cliquez sur l'icône Déplacer la ligne vers le haut.

**Remarque :** Cette opération modifie la séquence du processus de transformation.

## Déplacement d'une instruction d'une ligne vers le bas

Pour déplacer une instruction d'une ligne vers le bas, procédez comme suit :

1. Sélectionnez une instruction.
2. Cliquez sur l'icône Déplacer la ligne vers le bas.

**Remarque :** Cette opération modifie la séquence du processus de transformation.

## Effacement d'une instruction

Pour effacer une instruction, procédez comme suit :

1. Sélectionnez une instruction.
2. Cliquez sur l'icône Effacer les zones d'entrée.

Cette opération efface le texte des zones Fichier source, Fichier schéma et Fichier de sortie.

---

## Utilisation de la vue du processus de transformation

Pour lancer le processus de transformation de texte, procédez comme suit :

1. Dans la zone Fichier, saisissez le nom du fichier de paramètres ou recherchez-le à l'aide du bouton Parcourir.
2. Cliquez sur **Traiter**.

La zone de sortie située sous le bouton Traiter affiche des informations indiquant l'état du processus de transformation. Vous pouvez sauvegarder les informations de sortie en cliquant sur le bouton Sauvegarder situé en bas de la zone de texte, ou vous pouvez effacer toutes les informations d'état en cliquant sur le bouton Effacer.



---

## Chapitre 3. Transformation de données XML

Le langage XSL remplit les rôles suivants :

1. Langage permettant de spécifier le formatage des documents XML
2. Langage qui décrit la procédure de transformation d'un fichier XML en un autre fichier structuré

Les capacités de transformation du langage XSL peuvent être utilisées pour transformer un fichier XML en un autre fichier XML se conformant à un schéma XML différent ou à un fichier DTD.

Pour convertir un fichier XML dans un autre format XML, vous devez spécifier les règles de transformation en utilisant un fichier de règles XSL de transformation.

L'exemple suivant transforme les données du fichier MemberSubsystemExtracted.xml en utilisant le fichier MemberSubsystem.xsl comme fichier de règles XSL de transformation, avec le japonais comme langue nationale :



```
java com.ibm.wca.XMLTransformer.XMLTransformer -infile MemberSubsystemExtracted.xml  
-transform MemberSubsystem.xsl -outfile TransMbrStr.xml -param 'language="-10"'
```



```
QWEBCOMM/TRNWCSXML INFILE(MemberSubsystemExtracted.xml)  
TRANSFORM(MemberSubsystem.xsl) INSTROOT(/QIBM/UserData/WebCommerce/instances/my_inst)  
OUTFILE(TransMbrStr.xml) PARAM('language=-10')
```

L'outil de transformation XML utilise le langage XSL pour définir les règles de transformation d'un fichier XML en un autre fichier XML. La fonction de mappage de l'éditeur XSL Editor offre une interface visuelle vous permettant d'établir une association entre un élément d'un fichier DTD source et un élément d'un fichier DTD cible. Avec deux DTD, vous pouvez développer des règles XSL déterminant comment un fichier XML qui se conforme au premier fichier DTD (fichier source) est transformé en un fichier se conformant au deuxième fichier DTD (fichier cible).

## Lancement de l'éditeur XSL Editor

Pour lancer XSL Editor, utilisez le script ou l'instruction approprié, fourni dans le répertoire WebSphere Commerce suivant :

-  *unité:*\WebSphere\CommerceServer\bin\XSLEditor.cmd
-  *unité:*\Program Files\WebSphere\CommerceServer\bin\XSLEditor.cmd
-  /usr/WebSphere/CommerceServer/bin/xsleditor.sh
-   /opt/WebSphere/CommerceServer/bin/xsleditor.sh

 Dans les environnements iSeries, les administrateurs doivent copier au préalable les fichiers nécessaires à l'exécution de l'éditeur XSL Editor sur leurs machines Windows NT ou Windows 2000. Voici un exemple de la procédure à suivre :

1. Créez un nouveau répertoire sur la machine Windows (*unité:*\XMLTrans par exemple).
2. Créez les sous-répertoires suivants sous *unité:*\XMLTrans :
  - \bin
  - \lib\loader
  - \wcsadmin
3. Copiez les fichiers de la machine iSeries vers ces répertoires, conformément au schéma suivant :

| De                         |        | Vers                    |
|----------------------------|--------|-------------------------|
| /QIBM/ProdData/WebCommerce |        | <i>unité:</i> \XMLTrans |
| /bin                       |        | \bin                    |
| /XSLEditor.cmd             | -----> | \XSLEditor.cmd          |
| /lib/loader                |        | \lib\loader             |
| /wcmxmlp.jar               | -----> | \wcmxmlp.jar            |
| /wcmxslt.jar               | -----> | \wcmxslt.jar            |
| /wcsadmin                  |        | \wcsadmin               |
| /config.dtd                | -----> | \acsxml.ico             |
| /swing.jar                 | -----> | \swing.jar              |
| /XML_transform.ico         | -----> | \XML_transform.ico      |
| /XMLTransformerUI.zip      | -----> | \XMLTransformerUI.zip   |
| /XMLTransformWP.xml        | -----> | \XMLTransformWP.xml     |
| /XMLTransformWP.dtd        | -----> | \XMLTransformWP.dtd     |
| /XMLRuleConfig.xml         | -----> | \XMLRuleConfig.xml      |
| /XMLTransform.cnt          | -----> | \XMLTransform.cnt       |
| /XMLTransform.hlp          | -----> | \XMLTransform.hlp       |
| /XMLTransform.reg          | -----> | \XMLTransform.reg       |

4. Modifiez le fichier XSLEditor.cmd sur la machine Windows en insérant le texte suivant :

```
set WCS_HOME=unité:\XMLTrans
```

avant la ligne suivante :

```
set lib=%WCS_HOME%\lib\loader
```
5. Lancez l'outil de transformation XML en exécutant XSLEditor.cmd sur la machine Windows.

---

## Gestion des zones de constitution de règle de mappage

Lorsque vous lancez XSL Editor, il affiche la fenêtre Zone de constitution de règle de mappage. Cette fenêtre vous permet de gérer les zones de constitution de règle de mappage.

### Création d'une zone de constitution de règle de mappage

Pour créer une zone de constitution de règle de mappage, procédez comme suit :

1. Lancez XSL Editor.
2. Dans le menu déroulant, sélectionnez **[Nouvelle]**.
3. Dans la zone Nom, saisissez le nom de la nouvelle zone de constitution de règle de mappage.
4. Dans la zone Description , saisissez une description abrégée de la nouvelle zone de constitution de règle de mappage.
5. Dans la zone Schéma source, saisissez le nom d'un fichier existant ou recherchez, à l'aide du navigateur, un fichier existant qui sera utilisé en tant que schéma source.
6. Dans la zone Schéma cible, saisissez le nom d'un fichier existant ou recherchez, à l'aide du navigateur, un fichier existant qui sera utilisé en tant que schéma cible.
7. Dans la zone Fichier de règle XSL, saisissez le nom du nouveau fichier de règle à créer.

Vous pouvez indiquer un chemin complet dans cette zone. Si vous ne précisez pas le chemin, le fichier est créé dans le répertoire de travail en cours.

8. Cliquez sur **Ouvrir** pour créer et ouvrir la nouvelle zone de constitution de règle de mappage.

### Modification d'une zone de constitution de règle de mappage

Pour modifier une zone de constitution de règle de mappage, procédez comme suit :

1. Lancez XSL Editor.
2. Sélectionnez la zone de constitution de règle de mappage à modifier dans le menu déroulant.
3. Cliquez sur **Ouvrir** pour ouvrir la zone de constitution de règle de mappage.
4. Mettez à jour les zones que vous souhaitez modifier.
5. Cliquez sur **Sauvegarder** pour sauvegarder les modifications.

### Suppression d'une zone de constitution de règle de mappage

Pour supprimer une zone de constitution de règle de mappage, procédez comme suit :

1. Lancez XSL Editor.
2. Sélectionnez la zone de constitution de règle de mappage à supprimer dans le menu déroulant.
3. Cliquez sur le bouton Retirer pour supprimer l'entrée.

**Remarque :** Le retrait d'une zone de constitution de règle de mappage ne supprime pas les fichiers physiques du disque.

---

## Utilisation de l'éditeur XSL

Lorsque vous ouvrez une zone de constitution de règle de mappage à l'aide de l'éditeur XSL Editor, cette zone de constitution de règle de mappage est affichée dans la fenêtre principale de XSL Editor.

Le volet gauche de la fenêtre principale de l'éditeur XSL affiche une vue hiérarchique du fichier DTD source, intitulée "Schéma source". Le volet droit affiche une vue hiérarchique du fichier DTD cible, intitulée "Schéma cible".

### Création d'une règle de mappage

#### Mappage d'un élément sur un élément

Sélectionnez un élément de la hiérarchie source, puis faites-le glisser et déposez-le sur un élément de la hiérarchie cible. Une règle XSL est alors générée et affichée dans la vue Règle de mappage en bas de la fenêtre.

Voici un exemple de règle XSL générée :

```
<xsl:template match="merchant">
  <xsl:element name="MERCHANT">
    </xsl:element>
  </xsl:template>
```

**Remarque :** Toute relation ancêtre requise mais n'existant pas est automatiquement générée.

#### Mappage d'un attribut sur un attribut

Sélectionnez un attribut de la hiérarchie source, puis faites-le glisser et déposez-le sur un attribut de la hiérarchie cible. Une règle XSL est alors générée et affichée dans la vue Règle de mappage en bas de la fenêtre.

Voici un exemple de règle XSL générée :

```
<xsl:attribute name="MEADDR1">
  <xsl:apply-templates select="@mecmail1"/>
</xsl:attribute>
```

**Remarque :** Toute relation ancêtre requise mais n'existant pas est automatiquement générée.

#### Création d'une expression de mappage personnalisée

Pour créer une expression de mappage personnalisée, commencez par sélectionner un élément ou un attribut dans la hiérarchie cible. Cliquez ensuite dessus avec le bouton droit de la souris et sélectionnez l'option de menu Créer expression personnalisée. La fenêtre Créer expression personnalisée s'affiche alors et présente une liste de modèles disponibles et une liste d'expressions de règle dans deux menus déroulants. Créez l'expression personnalisée en procédant comme suit :

1. Sélectionnez le modèle auquel l'expression personnalisée doit être ajoutée.
2. Sélectionnez une expression de règle à créer (par exemple, **Expression de constante**).
3. Pour chaque paramètre répertorié dans la table, saisissez une valeur dans la colonne Valeur, puis appuyez sur **Entrée** pour valider cette valeur.
4. Cliquez sur **OK** pour terminer l'étape de création, ou sur **Annuler** pour quitter la fenêtre sans créer de règle.

La règle XSL générée est basée sur l'expression personnalisée définie dans le fichier de configuration de règles (XSLRuleConfig.xml). Vous pouvez modifier le fichier de configuration de règles et ajouter de nouvelles règles si besoin est. Pour mettre une règle à disposition dans la liste Expressions de règle, affectez la valeur "true" à l'attribut Visibilité de cette règle.

## Suppression d'une règle de mappage

Pour supprimer une règle de mappage, procédez comme suit :

1. Sélectionnez une règle dans la table de règles de mappage.
2. Cliquez dessus avec le bouton droit de la souris et sélectionnez **Supprimer**.

La règle et tous ses descendants sont alors supprimés.

**Remarque :** Les règles de mappage mises à jour et les règles XSL générées sont automatiquement conservées.

## Exécution d'une transformation XML

Pour exécuter une transformation XML, procédez comme suit :

1. Sélectionnez **Outils > Transformer** pour afficher la fenêtre Exécution de la transformation.
2. Renseignez les zones requises :
  - a. Dans la zone Fichier d'entrée XML, saisissez le chemin et le nom du fichier de données XML source, ou recherchez ce dernier à l'aide de l'explorateur.
  - b. Dans la zone Fichier de règle XSL, saisissez le chemin et le nom du fichier de règle de mappage à utiliser pour la transformation, ou recherchez ce dernier à l'aide de l'explorateur.

Si une zone de constitution de règle de mappage est ouverte, cette zone est déjà renseignée avec le chemin d'accès du fichier de règle de mappage actuellement ouvert dans la zone de constitution de règle de mappage.
  - c. Dans la zone Fichier de sortie XML, saisissez le chemin et le nom du nouveau fichier de données XML à créer au cours du processus de transformation.
3. Cliquez sur **Démarrer** pour lancer le processus de transformation XML, ou sur **Fermer** pour quitter la fenêtre sans exécuter de transformation.

## Personnalisation de la table des règles de mappage

Pour personnaliser la table des règles de mappage, procédez comme suit :

1. Pour masquer une colonne de la table, cliquez sur une cellule de la table avec le bouton droit de la souris et sélectionnez **Masquer colonne**.
2. Pour afficher une cellule masquée de la table, procédez comme suit :
  - a. Cliquez sur une cellule de la table avec le bouton droit de la souris.
  - b. Sélectionnez **Afficher colonnes** pour afficher la liste des colonnes masquées.
  - c. Sélectionnez la colonne voulue dans la liste.

**Remarque :** Pour sélectionner plusieurs colonnes, appuyez sur la touche **Maj**, maintenez-la enfoncée et cliquez sur les noms des colonnes.

- d. Cliquez sur **OK** pour afficher les colonnes sélectionnées ou sur **Annuler** pour annuler l'opération.

3. Pour afficher toutes les colonnes masquées de la table, cliquez sur une cellule de la table avec le bouton droit de la souris et sélectionnez **Afficher toutes les colonnes**.

Toutes les colonnes sont affichées dans l'ordre par défaut.

## **Affichage d'une expression de valeur ou d'une règle XSL complète**

Si vous cliquez sur une cellule dans les colonnes Expression de valeur ou Règle XSL, une fenêtre s'affiche et présente l'intégralité des règles pour la ligne sélectionnée.

## Chapitre 4. Génération d'un fichier DTD et d'un schéma

L'utilitaire de génération de DTD (DTD Generator) crée un fichier DTD et un schéma pouvant être utilisés avec le module Loader. DTD Generator utilise un fichier d'entrée contenant des noms de tables de base de données et génère soit un fichier DTD, soit un fichier DTD et un schéma avec un fichier XML détaillé décrivant la base de données, selon la façon dont vous appelez l'instruction DTD Generate.

### Configuration de DTD Generator

Le fichier DTD de Loader établit directement une correspondance avec le schéma de base de données de WebSphere Commerce. Chaque table est un élément et chaque colonne est un attribut.

**Exemple : Mappage d'un fichier DTD de Loader sur un schéma de base de données**

| Instructions DDL pour la table CATENTRY | DTD                           |
|---|-------------------------------|
| CREATE TABLE                            | <!ELEMENT CATENTRY EMPTY>     |
| "CATENTRY" (                            | <!ATTLIST CATENTRY            |
| "CATENTRY_ID" BIGINT NOT NULL           | CATENTRY_ID CDATA #REQUIRED   |
| "MEMBER_ID" BIGINT NOT NULL             | MEMBER_ID CDATA #REQUIRED     |
| "CATENTTYPE_ID" CHAR(16) NOT NULL       | CATENTTYPE_ID CDATA #REQUIRED |
| "MARKFORDELÈTE" INTEGER NOT NULL        | MARKFORDELÈTE CDATA #REQUIRED |
| "PARTNUMBER" VARCHAR(64) NOT NULL       | PARTNUMBER CDATA #REQUIRED    |
| "MFPARTNUMBER" VARCHAR(64)              | MFPARTNUMBER CDATA #IMPLIED   |
| "MFNAME" VARCHAR(64)                    | MFNAME CDATA #IMPLIED         |
| "URL" VARCHAR(254)                      | URL CDATA #IMPLIED            |
| "FIELD1" INTEGER                        | FIELD1 CDATA #IMPLIED         |
| "FIELD2" INTEGER                        | FIELD2 CDATA #IMPLIED         |
| "FIELD3" DECIMAL(20,5)                  | FIELD3 CDATA #IMPLIED         |
| "FIELD4" VARCHAR(254)                   | FIELD4 CDATA #IMPLIED         |
| "FIELD5" VARCHAR(254)                   | FIELD5 CDATA #IMPLIED         |
| "LASTUPDATE" TIMESTAMP                  | LASTUPDATE CDATA #IMPLIED     |
| "OID" VARCHAR(64)                       | OID CDATA #IMPLIED            |
| "ONSPECIAL" INTEGER                     | ONSPECIAL CDATA #IMPLIED      |
| "ONAUCTION" INTEGER                     | ONAUCTION CDATA #IMPLIED      |
| "BUYABLE" INTEGER                       | BUYABLE CDATA #IMPLIED        |
| "BASEITEM_ID" INTEGER                   | BASEITEM_ID CDATA #IMPLIED    |
| "CLASSIFGRP_ID" INTEGER                 | CLASSIFGRP_ID CDATA #IMPLIED  |
| "ITEMSPC_ID" INTEGER                    | ITEMSPC_ID CDATA #IMPLIED     |
| "STATE" INTEGER                         | STATE CDATA "1"               |
| );                                      | >                             |

Vous pouvez modifier le mode de fonctionnement de DTD Generator en procédant comme suit :

1. Créez un nouveau fichier de propriétés de personnalisation de DTD Generator.



Le fichier DB2ConnectionCustomizer.properties se trouve dans l'archive DTDGenerator.zip. Extrayez ce fichier, renommez-le en gardant l'extension .properties et placez-le dans un répertoire figurant dans le chemin de

classe. **Important** : Ne supprimez pas et ne modifiez pas le fichier DB2ConnectionCustomizer.properties existant.

•  400

Le fichier ISeries\_GENWCSDTD\_Customizer.properties se trouve dans le répertoire /QIBM/ProdData/WebCommerce/properties. Copiez ce fichier dans le répertoire /instroot/xml, renommez le nouveau fichier en gardant l'extension .properties, puis apportez toutes les modifications nécessaires au nouveau fichier. **Important** : Ne supprimez pas et ne modifiez pas le fichier ISeries\_GENWCSDTD\_Customizer.properties d'origine.

2. Modifiez les valeurs des pilotes de base de données dans le nouveau fichier. Par exemple :

```
DBVendorName = DB2
DBDriverName = COM.ibm.db2.jdbc.app.DB2Driver
DBURL = jdbc:db2:
```

où :

- DBVendorName est utilisé pour sélectionner le type de base de données. Les options disponibles sont les suivantes :
    - DB2 Universal Database for iSeries (DB2/iSeries)
    - DB2 pour les autres systèmes d'exploitation (DB2)
    - Base de données Oracle (Oracle)
  - DBDriverName est utilisé pour sélectionner le pilote JDBC. Les options disponibles sont les suivantes :
    - DB2 Universal Database for iSeries (com.ibm.db2.jdbc.app.DB2Driver)
    - DB2 pour les autres systèmes d'exploitation (COM.ibm.db2.jdbc.app.DB2Driver)
    - Base de données Oracle (oracle.jdbc.driver.OracleDriver)
  - DBURL est utilisé pour sélectionner l'URL d'accès à la base de données. Les options disponibles sont les suivantes :
    - DB2 Universal Database for iSeries (jdbc:db2://)
    - DB2 pour les autres systèmes d'exploitation (jdbc:db2:)
    - Base de données Oracle (jdbc:oracle:oci8:@)
3. Indiquez le nom du nouveau fichier comme valeur du paramètre de personnalisation de l'instruction DTD Generate.

---

## Génération d'un fichier DTD

Le fichier d'entrée TableNames.txt contient les noms de tables de base de données suivants, un sur chaque ligne :

```
MEMBER
ADDRBOOK
ADDRESS
```

Voici un exemple de méthode d'appel de DTD Generator :



```
java com.ibm.wca.DTDGenerator.GenerateDTD -dbname MALL -dbuser db2inst1
-dbpwd db2ibm -outfile wc.dtd -infile TableNames.txt
```

400

```
QWBCOMM/GENWCSDTD DATABASE(DATABASE_NAME) SCHEMA(MALL)
INSTROOT(/QIBM/UserData/WebCommerce/instances/mser)
PASSWD(mypassword) OUTFILE(wc.dtd)
INFILE(TableNames.txt)
```

Le fichier de sortie wc.dtd contient les informations suivantes :

```
<!ELEMENT MALL (( MEMBER | ADDRBOOK | ADDRESS)*)>
<!ELEMENT MEMBER EMPTY>
<!ATTLIST MEMBER
  MEMBER_ID      CDATA #REQUIRED
  TYPE           CDATA #REQUIRED
  STATE         CDATA #IMPLIED
>
<!ELEMENT ADDRBOOK EMPTY>
<!ATTLIST ADDRBOOK
  ADDRBOOK_ID CDATA #REQUIRED
  MEMBER_ID   CDATA #REQUIRED
  TYPE        CDATA #IMPLIED
  DISPLAYNAME CDATA #REQUIRED
  DESCRIPTION CDATA #IMPLIED
>
<!ELEMENT ADDRESS EMPTY>
<!ATTLIST ADDRESS
  ADDRESS_ID CDATA #REQUIRED
  ADDRESS_TYPE CDATA #IMPLIED
  MEMBER_ID   CDATA #REQUIRED
  ADDRBOOK_ID CDATA #REQUIRED
  ORGUNITNAME CDATA #IMPLIED
  FIELD3      CDATA #IMPLIED
  BILLINGCODE CDATA #IMPLIED
  BILLINGCODETYPE CDATA #IMPLIED
  STATUS      CDATA #IMPLIED
  ORGNAME     CDATA #IMPLIED
  ISPRIMARY   CDATA #IMPLIED
  LASTNAME    CDATA #IMPLIED
  PERSONTITLE CDATA #IMPLIED
  FIRSTNAME   CDATA #IMPLIED
  MIDDLENAME  CDATA #IMPLIED
  BUSINESSTITLE CDATA #IMPLIED
  PHONE1      CDATA #IMPLIED
  FAX1        CDATA #IMPLIED
  PHONE2      CDATA #IMPLIED
  ADDRESS1    CDATA #IMPLIED
  FAX2        CDATA #IMPLIED
  NICKNAME    CDATA #REQUIRED
  ADDRESS2    CDATA #IMPLIED
  ADDRESS3    CDATA #IMPLIED
  CITY        CDATA #IMPLIED
  STATE       CDATA #IMPLIED
  COUNTRY     CDATA #IMPLIED
  ZIPCODE     CDATA #IMPLIED
  EMAIL1      CDATA #IMPLIED
  EMAIL2      CDATA #IMPLIED
  PHONE1TYPE  CDATA #IMPLIED
  PHONE2TYPE  CDATA #IMPLIED
  PUBLISHPHONE1 CDATA #IMPLIED
  PUBLISHPHONE2 CDATA #IMPLIED
  BESTCALLINGTIME CDATA #IMPLIED
  PACKAGESUPPRESSION CDATA #IMPLIED
  LASTCREATE   CDATA #IMPLIED
  OFFICEADDRESS CDATA #IMPLIED
  SELFADDRESS  CDATA "0"
  FIELD1      CDATA #IMPLIED
```

```

FIELD2          CDATA #IMPLIED
TAXGEOCODE     CDATA #IMPLIED
SHIPPINGGEOCODE CDATA          #IMPLIED

```

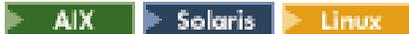
## Génération d'un schéma et d'un fichier XML détaillé

Dans cet exemple, DTD Generator est appelé comme suit :

- 

```

java com.ibm.wca.DTDGenerator.GenerateDTD -dbname SAMPLE
-dbuser johndoe -dbpwd password -xmlTableDesc c:\sample\sample.xml
-outfile tables.dtd -tablenames "employee,staff"

```
- 

```

java com.ibm.wca.DTDGenerator.GenerateDTD -dbname SAMPLE
-dbuser johndoe -dbpwd password -xmlTableDesc usr/sample/sample.xml
-outfile tables.dtd -tablenames "employee,staff"

```
- 

```

QWEBCOMM/GENWCSDTD DATABASE(MYDB) SCHEMA(SAMPLE)
INSTROOT(/QIBM/UserData/WebCommerce/instances/mser) PASSWD(mypassword)
OUTFILE(tables.dtd) TABNAMES('employee,staff') XMLTABDESC(/sample/sample.xml)

```

Le fichier schéma est créé dans le répertoire sample et porte le nom WCAWebForm.xsd. Le fichier de sortie sample.xml contient les informations suivantes :

```

<?xml version="1.0" encoding="UTF-8"?>
<formList xmlns="WCAWebForm.xsd" dbname="SAMPLE" dtdname="tables.dtd">
  <form name="EMPLOYEE">
    <uniqueIndex name="U2" columns="FIRSTNME, LASTNAME"/>
    <uniqueIndex name="U3" columns="MIDINIT, LASTNAME"/>
    <field name="EMPNO" type="string" maxLength="6" minOccurs='1'
      uniqueKey="true" showColumnInList="true" />
    <field name="FIRSTNME" type="string" maxLength="32" minOccurs='1'
      showColumnInList="true" />
    <field name="MIDINIT" type="string" maxLength="1" minOccurs='1'
      showColumnInList="true" />
    <field name="LASTNAME" type="string" maxLength="15" minOccurs='1'
      showColumnInList="true" />
    <field name="WORKDEPT" type="string" maxLength="3" showColumnInList="true" />
    <field name="PHONENO" type="string" maxLength="4"/>
    <field name="HIREDATE" type="date" maxLength="10"/>
    <field name="JOB" type="string" maxLength="8"/>
    <field name="EDLEVEL" type="integer" maxLength="5" minOccurs='1' />
    <field name="SEX" type="string" maxLength="1"/>
    <field name="BIRTHDATE" type="date" maxLength="10"/>
    <field name="SALARY" type="decimal" maxLength="9"/>
    <field name="BONUS" type="decimal" maxLength="9"/>
    <field name="COMM" type="decimal" maxLength="9"/>
  </form>
  <form name="STAFF">
    <field name="ID" type="integer" maxLength="5" minOccurs='1'
      uniqueKey="true" showColumnInList="true" />
    <field name="NAME" type="string" maxLength="9" showColumnInList="true" />
    <field name="DEPT" type="integer" maxLength="5" showColumnInList="true" />
    <field name="JOB" type="string" maxLength="5" showColumnInList="true" />
    <field name="YEARS" type="integer" maxLength="5" showColumnInList="true" />
    <field name="SALARY" type="decimal" maxLength="7"/>
    <field name="COMM" type="decimal" maxLength="7"/>
  </form>
</formList>

```

---

## Chapitre 5. Résolution d'identificateurs

Les données XML devant être chargées dans une base de données cible doivent contenir des identificateurs pour les éléments XML pour lesquels ils sont nécessaires. Pour générer ou localiser des identificateurs pour les entités de catalogue dans le document XML, appelez l'instruction ID Resolve.

L'utilitaire ID Resolver ne résout que les identificateurs d'une table primaire. Une table primaire est une table répertoriée dans la table KEYS ou SUBKEYS. S'il faut résoudre les identificateurs d'une table qui ne figure pas dans KEYS ou SUBKEYS, ajoutez cette table dans la table SUBKEYS avant de lancer l'utilitaire ID Resolver.

Voici des exemples de situations dans lesquelles vous aurez peut-être besoin de l'utilitaire ID Resolver :

- Chargement de nouvelles données au format XML lorsque des identificateurs sont requis pour ces données
- Mise à jour de données lorsque des identificateurs existent déjà pour un objet de la base de données

ID Resolver peut fournir des identificateurs à jour, ou bien les identificateurs peuvent être résolus à l'aide des techniques suivantes :

- Résolution d'alias interne
- Spécification du fichier de propriétés
- Résolution d'index à entrées uniques

---

### Configuration de l'utilitaire ID Resolver

Vous pouvez définir la façon dont ID Resolver gère les horodatages, l'archivage et les pilotes de base de données, en procédant comme suit :

1. Créez un nouveau fichier de propriétés de personnalisation d'ID Resolver.



Le fichier `DB2ConnectionCustomizer.properties` se trouve dans l'archive `IdResGen.zip`. Extrayez ce fichier, renommez-le en gardant l'extension `.properties` et placez-le dans un répertoire figurant dans le chemin de classe. **Important** : Ne supprimez pas et ne modifiez pas le fichier `DB2ConnectionCustomizer.properties` existant.



Le fichier `ISeries_RESWCSID_Customizer.properties` se trouve dans le répertoire `/QIBM/ProdData/WebCommerce/properties`. Copiez ce fichier dans le répertoire `/instroot/xml`, renommez le nouveau fichier en gardant l'extension `.properties`, puis apportez toutes les modifications nécessaires au nouveau fichier. **Important** : Ne supprimez pas et ne modifiez pas le fichier `ISeries_RESWCSID_Customizer.properties` d'origine.

2. Modifiez les valeurs des propriétés spécifiées dans le nouveau fichier.
3. Indiquez le nom du nouveau fichier comme valeur du paramètre de personnalisation de l'instruction ID Resolve.

## Définition de la façon dont ID Resolver gère les horodatages

Les masques d'horodatage d'entrée par défaut suivants sont fournis dans le fichier de propriétés de personnalisation d'ID Resolver :

```
InputTimeStampFormat.1 = dd/yyyy hh:mm:ss:SSSSSS
InputTimeStampFormat.2 = DD/MM/yyyy hh:mm:ss:SSSSSS
InputTimeStampFormat.3 = dd/yyyy/hh:mm:ss:SSSSSS
InputTimeStampFormat.4 = DD/MM/yyyy/HH:mm:ss:SSSSSS
InputTimeStampFormat.5 = dd/MM/yyyy/hh:mm:ss:SSSSSS
InputTimeStampFormat.6 = dd/MM/yyyy HH:mm:ss:SSSSSS
InputTimeStampFormat.7 = DD/yyyy HH:mm:ss:SSSSSS
```

Vous pouvez modifier ces masques d'horodatage ou ajouter autant de masques que vous voulez dans votre fichier de propriétés de personnalisation d'ID Resolver. Si vous ajoutez un horodatage d'entrée, vous devez utiliser le nombre suivant dans la séquence en cours. (Par exemple, le prochain masque d'horodatage d'entrée serait InputTimeStampFormat.8 si vous ajoutiez un masque à la liste ci-dessus).

Vous pouvez également personnaliser le format d'horodatage de sortie, le masque des microsecondes et le format spécifique de la base de données en modifiant les valeurs des propriétés suivantes dans votre fichier de propriétés de personnalisation d'ID Resolver :

```
TargetTimeStampFormat = dd/MM/yyyy HH:mm:ss:SSSSSS
MicroSecondMask = SSSSSS
DatabaseSpecificFormat = YYYY-MM-DD HH24:MI:SS
```

## Définition de la façon dont ID Resolver gère les archivages

Voici la section du fichier de propriétés de personnalisation d'ID Resolver qui spécifie les valeurs par défaut des propriétés associées à des "hashmaps" persistants :

```
////////////////////////////////////
/// 0 = Normal hashmap with no backend storage
/// 1 = JDBM
////////////////////////////////////

PersistentStorageType = 0

////////////////////////////////////
/// If PersistentStorageType != 0, set MemoryStorageSize to the maximum size
/// of the hashmap in memory data and after that the hashmap will stream
/// the data to a persistent storage as specified
/// If -1, then it uses the normal hashmap with no backend storage
////////////////////////////////////

MemoryStorageSize = 1
```

Vous pouvez spécifier la façon dont ID Resolver gère une mémoire persistante en définissant une valeur pour PersistentStorageType dans votre fichier de propriétés de personnalisation d'ID Resolver.

- Si vous définissez PersistentStorageType = 0, ID Resolver fonctionne en mode "normal" (les "hashmaps" de symboles existent en mémoire).
- Si vous définissez PersistentStorageType = 1, JDBM est utilisé pour conserver les symboles et les clés.

Vous pouvez spécifier le nombre d'enregistrements stockés en mémoire en définissant une valeur pour `MemoryStorageSize` dans votre fichier de propriétés de personnalisation d'ID Resolver.

- La valeur "1" pour `MemoryStorageSize` indique qu'un seul enregistrement est conservé en mémoire.
- La valeur "-1" pour `MemoryStorageSize` a une signification spéciale, indiquant que tous les enregistrements sont conservés en mémoire.

Dans ce cas, ID Resolver reprend son comportement "normal".

## Définition de la façon dont ID Resolver gère les pilotes de base de données

Les lignes suivantes du fichier de propriétés de personnalisation d'ID Resolver spécifient les valeurs par défaut des pilotes de base de données :

```
DBVendorName = DB2
DBDriverName = COM.ibm.db2.jdbc.app.DB2Driver
DBURL = jdbc:db2:
```

où :

- `DBVendorName` est utilisé pour sélectionner le type de base de données.  
Les options disponibles sont les suivantes :
  - DB2 Universal Database for iSeries (DB2/iSeries)
  - DB2 pour les autres systèmes d'exploitation (DB2)
  - Base de données Oracle (oracle)
- `DBDriverName` est utilisé pour sélectionner le pilote JDBC.  
Les options disponibles sont les suivantes :
  - DB2 Universal Database for iSeries (com.ibm.db2.jdbc.app.DB2Driver)
  - DB2 pour les autres systèmes d'exploitation (COM.ibm.db2.jdbc.app.DB2Driver)
  - Base de données Oracle (oracle.jdbc.driver.OracleDriver)
- `DBURL` est utilisé pour sélectionner l'URL d'accès à la base de données.  
Les options disponibles sont les suivantes :
  - DB2 Universal Database for iSeries (jdbc:db2://)
  - DB2 pour les autres systèmes d'exploitation (jdbc:db2:)
  - Base de données Oracle (jdbc:oracle:oci8:@)

---

## Détermination du mode de traitement des données

L'instruction ID Resolve vous permet de choisir la méthode de chargement, de mise à jour ou mixte pour traiter le fichier d'entrée.

- Utilisez la méthode de chargement pour traiter le fichier d'entrée si *aucun* enregistrement du fichier *n'existe* dans la base de données.
- Utilisez la méthode de mise à jour pour traiter le fichier d'entrée si *tous* les enregistrements du fichier *existent* dans la base de données.
- Utilisez la méthode mixte pour traiter le fichier d'entrée si *seulement certains* enregistrements du fichier *existent* dans la base de données.

## Choix de la méthode de chargement

La méthode de chargement de l'utilitaire ID Resolver est utilisée pour générer de nouveaux identificateurs pour les enregistrements chargés dans la base de données. Par ce biais, de nouveaux identificateurs sont créés pour les enregistrements. L'exemple suivant permet de générer des identificateurs pour de nouvelles données :

-    

```
idresgen -dbname db -dbuser user -dbpwd pwd -infile input.xml  
-outfile output.xml -method load -customizer customizer -schemaname mall
```

La méthode de chargement est la méthode par défaut.

-     

```
./idresgen.sh -dbname db -dbuser user -dbpwd pwd -infile input.xml  
-outfile output.xml -method load -customizer customizer -schemaname mall
```

La méthode de chargement est la méthode par défaut.

-   
QWEBCOMM/RESWC SID DATABASE(DATABASE\_NAME) SCHEMA(MALL)  
INSTROOT(/QIBM/UserData/WebCommerce/instances/mser)  
PASSWD(mypassword) INFILE(input.xml) OUTFILE(output.xml)  
METHOD(\***LOAD**)

## Choix de la méthode de mise à jour

Si vous spécifiez la méthode de mise à jour de l'utilitaire ID Resolver, les enregistrements contenus dans le fichier d'entrée doivent déjà exister dans la base de données. ID Resolver localise les identificateurs dans la base de données. Si un enregistrement n'existe pas dans la base de données, ID Resolver ne pourra pas résoudre l'identificateur de cet enregistrement et il indiquera qu'une erreur s'est produite. L'exemple suivant permet de localiser les identificateurs des données qui existent déjà dans la base de données :

-    

```
idresgen -dbname db -dbuser user -dbpwd pwd -infile input.xml  
-outfile output.xml -method update -customizer customizer -schemaname mall
```

-     

```
./idresgen.sh -dbname db -dbuser user -dbpwd pwd -infile input.xml  
-outfile output.xml -method update -customizer customizer -schemaname mall
```

-   
QWEBCOMM/RESWC SID DATABASE(DATABASE\_NAME) SCHEMA(MALL)  
INSTROOT(/QIBM/UserData/WebCommerce/instances/mser) PASSWD(mypassword)  
INFILE(input.xml) OUTFILE(output.xml) METHOD(\***UPD**)

## Choix de la méthode mixte

Si le fichier de données d'entrée contient des enregistrements qui existent déjà dans la base de données ainsi que quelques enregistrements nouveaux, ID Resolver doit être exécuté en utilisant la méthode mixte. Avec cette méthode, ID Resolver crée de nouveaux identificateurs uniquement pour les enregistrements qui n'existent pas dans la base de données. Pour les autres enregistrements, l'identificateur existant est obtenu depuis la base de données. L'exemple suivant permet de générer des identificateurs pour les nouvelles données et de localiser les identificateurs des données qui existent déjà dans la base de données :

-   

```
idresgen -dbname db -dbuser user -dbpwd pwd -infile input.xml  
-outfile output.xml -method mixed -customizer customizer -schemaname mall
```
-   

```
./idresgen.sh -dbname db -dbuser user -dbpwd pwd -infile input.xml  
-outfile output.xml -method mixed -customizer customizer -schemaname mall
```
-   

```
QWEBCOMM/RESWC SID DATABASE(DATABASE_NAME) SCHEMA(MALL)  
INSTROOT(/QIBM/UserData/WebCommerce/instances/mser) PASSWD(my password)  
INFILE(input.xml) OUTFILE(output.xml) METHOD(*MIX)
```

---

## Utilisation des techniques de résolution d'ID

ID Resolver peut utiliser un fichier de propriétés Java pour déterminer quelles colonnes d'une table primaire doivent être utilisées en consultation pour les tables nécessitant l'identificateur d'une table primaire. Une table est dite primaire si elle est répertoriée dans la table KEYS ou SUBKEYS.

Pour permettre l'utilisation de la résolution d'alias interne avec ID Resolver, un alias est placé dans l'attribut de clé primaire (identificateur) du fichier XML. Il est alors possible d'utiliser l'alias dans tout le fichier XML pour faire référence à cet élément. Ce processus évite au programme d'avoir à déterminer les index à entrées uniques nécessaires pour créer le fichier XML.

L'utilitaire ID Resolver peut également analyser le schéma de la base de données pour déterminer si un des index à entrées uniques peut répondre à ses besoins. Il ne recherche un index à entrées uniques que lorsqu'il n'y a pas d'entrée dans le fichier de propriétés pour la table analysée ou lorsqu'il n'y a pas de fichier de propriétés. Si ces conditions sont remplies, une vérification d'index à entrées uniques est effectuée. L'index à entrées uniques est considéré comme valide s'il existe et n'inclut pas de clé primaire pour la table.

## Spécification d'un fichier de propriétés à l'aide de l'utilitaire ID Resolver

ID Resolver vous permet d'utiliser un autre fichier de propriétés Java pour décrire les colonnes d'une entrée principale qui doivent être utilisées en consultation pour les tables nécessitant l'identificateur d'une ligne principale.

Le fichier de propriétés par défaut est `IdResolveKeys.properties`, mais vous pouvez, si vous le souhaitez, le modifier ou indiquer votre propre fichier lorsque vous appelez l'instruction ID Resolve.

- NT
2000
AIX
Solaris
Linux

Le fichier `IdResolveKeys.properties` se trouve dans le répertoire suivant :

- NT *unité:* \WebSphere\CommerceServer\properties
- 2000 *unité:* \Program Files\WebSphere\CommerceServer\properties
- AIX /usr/WebSphere/CommerceServer/properties
- Solaris Linux /opt/WebSphere/CommerceServer/properties

Si vous ne voulez pas placer ce fichier dans le répertoire en cours lorsque vous exécutez ID Resolver, vous pouvez le placer dans un répertoire défini dans la variable d'environnement CLASSPATH. Vous pouvez également indiquer le chemin d'accès complet de ce fichier.

- 400

Pour modifier le fichier `IdResolveKeys.properties`, copiez-le à partir du répertoire `/QIBM/ProdData/WebCommerce/properties`, sauvegardez-le dans le répertoire `/instroot/xml`, puis apportez toutes les modifications nécessaires au nouveau fichier.

**Remarque :** Le répertoire ci-dessus se trouve dans le chemin de classe utilisé par l'instruction RESWCSID.

## Utilisation d'un fichier de propriétés pour générer des identificateurs

Dans l'exemple suivant, vous devez résoudre les identificateurs `ADDRBOOK_ID` et `ADDRESS_ID` correspondant respectivement aux enregistrements `ADDRBOOK` et `ADDRESS`. Les identificateurs des enregistrements `MEMBER` sont déjà connus. Chaque enregistrement nécessite un identificateur valide pour la base de données WebSphere Commerce. En outre, l'identificateur `ADDRBOOK_ID` contenu dans l'enregistrement `ADDRESS` nécessite l'identificateur provenant de la table primaire pour satisfaire à sa contrainte de clé associée.

```

<MEMBER
  MEMBER_ID="100"
  TYPE="U"
  STATE="1"
/>
<MEMBER
  MEMBER_ID="101"
  TYPE="U"
  STATE="1"
/>
<ADDRBOOK
  MEMBER_ID="100"
  DISPLAYNAME="Amis"
  DESCRIPTION="Tous mes amis"
  TYPE="P"
/>
<ADDRESS
  ADDRBOOK_ID="@Friends"
  MEMBER_ID="101"
  NICKNAME="Bob"
  ADDRESS1="15 Brave Developers St."
  CITY="Toronto"

```

**Valeur réelle de la colonne DISPLAYNAME**

**Se réfère à l'enregistrement ADDRBOOK en utilisant la valeur DISPLAYNAME pour la recherche**

```

    ZIPCODE="A0A0A0"
    COUNTRY="Canada"
    STATUS="P"
  />

```

Il vous faut un fichier de propriétés pour identifier les colonnes de la ligne principale qui seront utilisées par les lignes de relation lors de la résolution de l'identificateur pour la colonne de clé associée. La procédure suivante garantit que l'analyse syntaxique du fichier précédent s'exécute correctement.

Dans le fichier `IdResolveKeys.properties`, spécifiez les informations suivantes :

```

NAMEDELIMITER=@
SELECTDELIMITER=:

ADDRBOOK=@DISPLAYNAME:DISPLAYNAME
ADDRESS=@NICKNAME:NICKNAME

```

`NAMEDELIMITER` et `SELECTDELIMITER` définissent les délimiteurs utilisés dans tout le fichier de propriétés et doivent être utilisés de manière cohérente.

`ADDRBOOK=@DISPLAYNAME:DISPLAYNAME` indique que, lorsqu'un enregistrement de carnet d'adresses est reçu, l'identificateur de la ligne de ce carnet d'adresses est créé. La zone `DISPLAYNAME` est alors extraite de l'enregistrement d'entrée et utilisée pour former une association avec le nouvel identificateur. La chaîne `DISPLAYNAME` est utilisée pour établir une correspondance avec le `DISPLAYNAME` de la ligne du carnet d'adresses afin de résoudre l'identificateur nécessaire à la clé associée.

En utilisant l'exemple d'entrée précédent, dans lequel la valeur de `DISPLAYNAME` est `Amis`, considérons que l'identificateur créé pour cet enregistrement est `12951`. Le `DISPLAYNAME` est utilisé comme détecteur clé de `12951`. Le traitement passe à l'enregistrement suivant, `ADDRESS`, où `ADDRBOOK_ID` se présente sous la forme `"@..."`, ce qui indique que ce qui suit le délimiteur doit être utilisé pour rechercher l'identificateur du carnet d'adresses. La chaîne correspond au `DISPLAYNAME` et la valeur `12951` est renvoyée et placée dans l'attribut `ADDRBOOK_ID`.

```

<MEMBER
  MEMBER_ID="100"
  TYPE="U"
  STATE="1"
/>
<MEMBER
  MEMBER_ID="101"
  TYPE="U"
  STATE="1"
/>
<ADDRBOOK
  ADDRBOOK_ID="12951"           Clé primaire générée
  MEMBER_ID="100"
  DISPLAYNAME="Amis"           Valeur de ADDRBOOK DISPLAYNAME inchangée
  DESCRIPTION="Tous mes amis"
  TYPE="P"
/>
<ADDRESS
  ADDRESS_ID="13051"           Clé primaire générée
  ADDRBOOK_ID="12951"        ADDRESS fait référence à l'enregistrement
  MEMBER_ID="101"
  NICKNAME="Bob"
  ADDRESS1="15 Brave Developers St."
  CITY="Toronto"

```

```
ZIPCODE="A0A0A0"  
COUNTRY="Canada"  
STATUS="P"  
</>
```

## Utilisation d'un fichier de propriétés avec clés composées

Une clé constituée de plus de deux colonnes est une clé composée. Vous pouvez définir une recherche par clés composées dans le fichier de propriétés en spécifiant NAMEDELIMITER et SELECTDELIMITER, suivis des noms de zones. Par exemple, pour que les critères de recherche des enregistrements ADDRBOOK soient le composé du nom affiché et de l'ID membre, indiquez les informations suivantes dans le fichier de propriétés :

```
ADDRBOOK=@DISPLAYNAME@MEMBER_ID:DISPLAYNAME MEMBER_ID
```

Ensuite, le fragment de fichier d'entrée XML suivant :

```
<ADDRBOOK  
  MEMBER_ID="100"  
  DISPLAYNAME="Amis"                ADDRBOOK "Amis" de MEMBER 100  
  DESCRIPTION="Tous mes amis"  
  TYPE="P"  
</>  
<ADDRBOOK  
  MEMBER_ID="101"  
  DISPLAYNAME="Amis"                ADDRBOOK "Amis" de MEMBER 101  
  DESCRIPTION="Tous mes amis"  
  TYPE="P"  
</>  
<ADDRESS  
  ADDRBOOK_ID="@Friends@100"      Recherche la clé primaire pour ADDRBOOK  
                                   "Amis" de MEMBER 100  
  
  MEMBER_ID="101"  
  NICKNAME="Bob"  
  ADDRESS1="15 Brave Developers St."  
  CITY="Toronto"  
  ZIPCODE="A0A0A0"  
  COUNTRY="Canada"  
  STATUS="P"  
</>
```

produirait le fragment suivant après résolution :

```
<MEMBER  
  MEMBER_ID="100"  
  TYPE="U"  
  STATE="1"  
</>  
<MEMBER  
  MEMBER_ID="101"  
  TYPE="U"  
  STATE="1"  
</>  
<ADDRBOOK  
  ADDRBOOK_ID="12951"              ADDRBOOK intéressant  
  MEMBER_ID="100"  
  DISPLAYNAME="Amis"  
  DESCRIPTION="Tous mes amis"  
  TYPE="P"  
</>  
<ADDRBOOK  
  ADDRBOOK_ID="12952"  
  MEMBER_ID="101"  
  DISPLAYNAME="Amis"
```

```

    DESCRIPTION="Tous mes amis"
    TYPE="P"
  />
  <ADDRESS
    ADDRESS_ID="13051"
    ADDRBOOK_ID="12951"
    MEMBER_ID="101"
    NICKNAME="Bob"
    ADDRESS1="15 Brave Developers St."
    CITY="Toronto"
    ZIPCODE="A0A0A0"
    COUNTRY="Canada"
    STATUS="P"
  />

```

**ADDRESS fait référence à l'enregistrement**

## Utilisation d'un fichier de propriétés avec clés primaires en cascade

La table primaire STOREENT définit une clé primaire STOREENT\_ID. STORE, une table externe référençant STOREENT, définit une clé primaire STORE\_ID qui est une clé associée à la table primaire STOREENT. Cela signifie que la valeur de STORE\_ID doit être l'une des valeurs de STOREENT\_ID. STORE\_ID, la clé primaire de la table externe STORE, a par conséquent une double fonction : clé primaire et clé associée.

Supposons qu'une autre table, CONTRACT, soit une table externe définie dans STORE et que la clé associée pour CONTRACT, STORE\_ID, fasse référence à la clé primaire STORE\_ID de STORE. La table STORE est par conséquent une table primaire par rapport à la table CONTRACT.

Puisque l'identificateur STORE\_ID de la table STORE est référencé à partir de STOREENT\_ID plutôt que créé, l'utilitaire ID Resolver ne crée pas d'alias interne ni d'association de valeur d'ID pour la table STORE. Lorsque la table CONTRACT tente de résoudre l'identificateur STORE\_ID à partir de la table STORE, elle n'obtient aucune valeur.

Du fait de cette condition spéciale, vous devez spécifier de manière explicite la création de l'alias interne en créant une entrée dans le fichier de propriétés. Dans le fichier IdResolveKeys.properties, spécifiez les informations suivantes :

```
"STORE=@STORE_ID:STORE_ID"
```

Cela force ID Resolver à effectuer les opérations suivantes :

- création de l'alias interne et de l'association de valeur d'ID lors de la résolution de l'identificateur STORE\_ID comme référence externe,
- utilisation de l'association lors de la résolution de l'identificateur STORE\_ID pour la table CONTRACT.

L'utilisation de l'entrée STORE=@STORE\_ID:STORE\_ID dans le fichier de propriétés et dans le fragment de fichier d'entrée XML suivant :

```

<STOREENT
  IDENTIFIER="Out Fashions"
  MEMBER_ID="-2000"
  STOREENT_ID="@storeent_id_1"
  TYPE="G"
/>
<STORE
  STORE_ID="@storeent_id_1"
  STOREGRP_ID="1"
  STORELEVEL="store_level"

```

```

/>
<CONTRACT
  CONTRACT_ID="@contract_id_1"
  STATE="0"
  STORE_ID="@storeent_id_1"
/>

```

produirait le fragment suivant après résolution :

```

<STOREENT
  IDENTIFIER="Out Fashions"
  MEMBER_ID="-2000"
  STOREENT_ID="10501"
  TYPE="G"
/>
<STORE
  STORE_ID="10501"
  STOREGRP_ID="1"
  STORELEVEL="store_level"
/>
<CONTRACT
  CONTRACT_ID="@contract_id_1"
  STATE="0"
  STORE_ID="10501"
/>

```

## Utilisation de la résolution d'alias interne

Pour permettre l'utilisation de la résolution d'alias interne avec ID Resolver, un alias est placé dans l'attribut de clé primaire (identificateur) du fichier XML. Il est alors possible d'utiliser l'alias dans tout le fichier XML pour faire référence à cet élément. Ce processus évite au programme d'avoir à déterminer les index à entrées uniques nécessaires pour créer le fichier XML.

Les alias internes doivent être utilisés de manière cohérente dans tout le fichier. Si un ID de carnet d'adresses, ADDRBOOK\_ID, est associé à l'alias @addrbook\_1, toutes les références de clés associées à cet ID, dans le fichier, doivent utiliser @addrbook\_1. Notez que les alias sont transitoires. Ils ne sont pas sauvegardés et ne peuvent pas être utilisés dans un autre fichier XML sans être réintroduits.

### Exemple partiel d'utilisation de la résolution d'ID d'alias interne

Avant la résolution :

```

<MEMBER
  MEMBER_ID="100"
  TYPE="U"
  STATE="1"
/>
<ADDRBOOK
  ADDRBOOK_ID="@addrbook_1"           Alias d'ADDRBOOK
  MEMBER_ID="100"
  DISPLAYNAME="Amis"
  DESCRIPTION="Tous mes amis"
  TYPE="P"
/>
<ADDRESS
  ADDRESS_ID="@address_1"             Alias d'ADDRESS
  ADDRBOOK_ID="@addrbook_1"         Fait référence à l'alias d'ADDRBOOK
  MEMBER_ID="101"
  NICKNAME="Bob"
  ADDRESS1="1 Brave Developer St."
  CITY="Toronto"

```

```
    ZIPCODE="A3B0F4"  
    COUNTRY="Canada"  
    STATUS="P"  
/>
```

### Après la résolution :

```
<MEMBER  
  MEMBER_ID="100"  
  TYPE="U"  
  STATE="1"  
/>  
<ADDRBOOK  
  ADDRBOOK_ID="11801"           Clé primaire générée  
  MEMBER_ID="100"  
  DISPLAYNAME="Amis"  
  DESCRIPTION="Tous mes amis"  
  TYPE="P"  
/>  
<ADDRESS  
  ADDRESS_ID="11901"           Clé primaire générée  
  ADDRBOOK_ID="11801"        Fait référence à l'entrée d'ADDRBOOK  
  MEMBER_ID="100"  
  NICKNAME="Bob"  
  ADDRESS1="1 Brave Developer St."  
  CITY="Toronto"  
  ZIPCODE="A3B0F4"  
  COUNTRY="Canada"  
  STATUS="P"  
/>
```

## Utilisation de la résolution d'index à entrées uniques

La résolution d'index à entrées uniques, comportement par défaut d'ID Resolver, est utilisée lorsqu'il n'y a pas d'entrée dans le fichier de propriétés pour la table analysée ou lorsqu'il n'y a pas de fichier de propriétés. Ce type de résolution utilise tout index à entrées uniques spécifié dans une table comme moyen de localisation de l'identificateur. Par exemple, MEMBER\_ID plus IDENTIFIÉ est un index à entrées uniques de la table CATALOG, qui peut par conséquent être utilisé comme point de résolution de la clé primaire CATALOG\_ID de la table CATALOGDSC.

Pour mettre à jour le contenu de la base de données, vous devez connaître la clé unique de la table primaire de la base de données. Vous pouvez demander à votre base de données de la trouver. Par exemple, une instruction DB2 permettant d'extraire une clé unique peut se présenter comme suit :

```
db2 describe indexes for table schema.tablename show detail
```

### Exemple partiel de résolution d'index à entrées uniques

#### Avant la résolution :

```
<MEMBER  
  MEMBER_ID="100"  
  TYPE="O"  
  STATE="1"  
/>  
  
<CATALOG  
  DESCRIPTION="Catalogue d'hiver"  
  IDENTIFIÉ="WC2001"  
  MEMBER_ID="100"
```

```

    TPCLEVEL="2"
  />

  <CATALOGDSC
    CATALOG_ID="@WC2001@100"
    FULLIMAGE="c:\store\img\wc.gif"
    LANGUAGE_ID="-1"
    LONGDESCRIPTION="Catalogue d'hiver 2001"
    SHORTDESCRIPTION="Catalogue d'hiver 2001"
    NAME="InFashion - Catalogue d'hiver 2001"
    THUMBNAIL="c:\store\img\wc_th.gif"
  />

```

**Fait référence au catalogue "WC2001" du membre "100" (Remarque : l'ordre est important.)**

### Après la résolution :

```

  <MEMBER
    MEMBER_ID="100"
    TYPE="0"
    STATE="1"
  />
  <CATALOG
    CATALOG_ID="10351"
    DESCRIPTION="Catalogue d'hiver"
    IDENTIFIER="WC2001"
    MEMBER_ID="100"
    TPCLEVEL="2"
  />
  <CATALOGDSC
    CATALOG_ID="10351"
    FULLIMAGE="c:\store\img\wc.gif"
    LANGUAGE_ID="-1"
    LONGDESCRIPTION="Catalogue d'hiver 2001"
    SHORTDESCRIPTION="Catalogue d'hiver 2001"
    NAME="InFashion - Catalogue d'hiver 2001"
    THUMBNAIL="c:\store\img\wc_th.gif"
  />

```

**Clé primaire générée automatiquement**

**Fait référence au catalogue approprié**

---

## Chargement de données dans la table MEMBER

L'utilitaire ID Resolver gère la résolution pour les tables dont les identificateurs sont générés par le système. Cela inclut toutes les tables et colonnes enregistrées dans les tables KEYS ou SUBKEYS. Le processus de résolution se décompose en deux étapes :

1. Détermination de l'existence d'une table primaire (c'est-à-dire répertoriée dans KEYS ou SUBKEYS) dans la base de données.  
 Cette résolution est basée sur le contenu des données XML de cet élément et utilise soit une résolution d'index à entrées uniques, soit une spécification du fichier de propriétés.
2. Détermination de l'existence d'une clé associée à une table primaire.  
 Cette opération est réalisée via une spécification de résolution dans l'attribut de clé associée de la table concernée.

La table MEMBER est utilisée comme "super classe" pour les tables ORGENTITY, MBRGRP et USER. Cela crée un schéma "is-a", qui s'avère utile pour conserver l'intégrité référentielle lorsque des tables ont des contraintes de clés associées pour les sous-types de la table MEMBER. Toutefois, comme tous les sous-types MEMBER partagent un type de base commun, l'identificateur doit être unique au niveau des sous-types. Cela signifie qu'un identificateur ORGENTITY\_ID doit être unique dans l'ensemble des MBRGRP\_ID et USER\_ID. Pour ce faire, la table KEYS fait référence uniquement aux tables ORGENTITY, MBRGRP et USER, et définit

des intervalles réciproquement exclusifs pour leurs identificateurs. Chacun des sous-types possède une clé primaire et chacune de ces clés primaires est également une clé associée à la clé primaire de la table MEMBER.

Les contraintes existant entre la table MEMBER et ses sous-types créent une situation où un MEMBER et un sous-type ne peuvent pas avoir d'ID synchronisé. Pour pouvoir charger les tables ORGENTITY, MBRGRP et USER dans le système, l'utilitaire ID Resolver identifie le schéma "is-a" et le traite de manière appropriée. La syntaxe XML suivante pour l'utilitaire ID Resolver :

```
<ORGENTITY
  ORGENTITY_ID="@orgAlias"
  ORGENTITYNAME="Test Org"
  ORGENTITYTYPE="0">
  <ISA>
    <MEMBER
      TYPE="0"
      STATE="1"
    />
  </ISA>
</ORGENTITY>
```

génère la syntaxe suivante :

```
<MEMBER
  MEMBER_ID="12345"
  TYPE="0"
  STATE="1"
/>
<ORGENTITY
  ORGENTITY_ID="12345"           Synchronisation avec l'élément membre
  ORGENTITYNAME="Test Org"
  ORGENTITYTYPE="0"
/>
```

De cette façon, ID Resolver traite le sous-élément <isa> et crée un identificateur synchronisé.

---

## Création d'une relation externe à l'aide de la table REFKEYS

La table REFKEYS est créée pour représenter une relation externe entre des tables, qui n'existe pas dans la base de données. En règle générale, le schéma de la base de données décrit la relation externe en créant une déclaration de clé associée qui établit un lien entre une colonne d'une table et une autre table. Si aucune relation externe n'est définie dans le schéma de base de données et que les identificateurs doivent être résolus en tant que clé associée, procédez comme suit :

1. Créez une table REFKEYS comme indiqué dans l'exemple de DDL suivant :

```
CREATE TABLE "REFKEYS" (
  "FKTABLE_NAME" CHAR(18) NOT NULL ,
  "FKCOLUMN_NAME" CHAR(18) NOT NULL ,
  "TABLENAME" CHAR(18) NOT NULL
);
```

où :

FKTABLE\_NAME est le nom de la table externe (ou "enfant"),  
FKCOLUMN\_NAME est le nom de la colonne externe,  
TABLENAME est le nom de la table primaire (ou "parent").

2. Créez dans la table REFKEYS une entrée qui décrit la relation externe requise.

## Identification et résolution des erreurs

Si des erreurs se produisent lors de la résolution des identificateurs, reportez-vous au tableau suivant :

| Erreur                   | Méthode utilisée | Cause possible  | Solution possible  |  |
|--------------------------|------------------|---|--|--|
| Clé primaire non résolue | Toutes           | ID Resolver <b>ne résoud pas</b> la clé primaire (identificateur) d'une table qui <b>n'est pas</b> spécifiée dans la table KEYS ou SUBKEYS.   | Ajoutez dans la table SUBKEYS le nom de la table pour laquelle la clé primaire doit être résolue avant de lancer l'utilitaire ID Resolver.<br><br>En outre, assurez-vous que la relation de clé associée existe bien dans le schéma de base de données.  |  |
|                          | Mise à jour      | La clé primaire est résolue en effectuant une recherche dans la base de données. La recherche dans la base de données est générée en utilisant les informations d'entrée du fichier de propriétés ou l'index à entrées uniques de la table donnée. Les entrées du fichiers de propriétés ont la priorité. | Vérifiez que les informations d'index à entrées uniques figurant dans le fichier d'entrée sont correctes.<br><br>Vous pouvez également créer ou modifier l'entrée appropriée du fichier de propriétés pour générer la recherche appropriée dans la base de données afin de résoudre la clé primaire. |  |
| Clé associée non résolue | Toutes           | ID Resolver <b>ne résoud pas</b> la clé associée d'une table dans laquelle les références externes sont associées à des tables qui <b>ne sont pas</b> spécifiées dans la table KEYS ou SUBKEYS.   | Ajoutez dans la table SUBKEYS le nom de la table référencée avant de lancer l'utilitaire ID Resolver.<br><br>En outre, assurez-vous que la relation de clé associée existe bien dans le schéma de base de données.   |  |
|                          |                  | Une clé associée est résolue à l'aide de l'alias interne ou en effectuant une recherche dans la base de données. La recherche dans la base de données est effectuée uniquement si l'alias interne ne parvient pas à résoudre la clé associée.   | L'alias interne est généré en utilisant la clé primaire et l'entrée du fichier de propriétés.  | Vérifiez que les valeurs d'alias interne utilisées comme source et comme cible dans le fichier d'entrée sont correctes.  |
|                          |                  |   | La recherche dans la base de données est générée en utilisant les informations d'entrée du fichier de propriétés ou l'index à entrées uniques de la table donnée. Les entrées du fichiers de propriétés ont la priorité.   | Vérifiez que les informations d'index à entrées uniques figurant dans le fichier d'entrée sont correctes.<br><br>Vous pouvez également créer ou modifier l'entrée appropriée du fichier de propriétés pour générer la recherche appropriée dans la base de données afin de résoudre la clé primaire. |

---

## Chapitre 6. Chargement de données

Avant de charger des données, vous devez effectuer les tâches suivantes :

1. Génération d'un fichier DTD et d'un schéma à utiliser avec le module Loader (la première fois que des données sont chargées)

**Remarque :** Si vous chargez des données pour un archivage de magasin et que vous avez créé le fichier XML à l'aide des fichiers DTD fournis avec cet archivage, cette étape n'est pas nécessaire.

2. Résolution des identificateurs (si nécessaire)

Les données doivent être au format XML avec un fichier DTD associé. Pour charger des données, appelez l'instruction Load.

---

### Configuration du module Loader

Le module Loader vous permet de définir le mode de fonctionnement de l'utilitaire de chargement à l'aide des fonctions suivantes :

- Non prise en compte des éléments du fichier d'entrée
- Insertion de la valeur NULL dans une colonne
- Chargement de données d'horodatage et de date
- Chargement des horodatages en cours
- Gestion de files d'attente d'événements
- Exécution avec différents logiciels de base de données et systèmes d'exploitation
- Substitution d'un composant
- Utilisation de la synchronisation de l'espace de recherche de l'Assistant produit

Vous pouvez personnaliser ces fonctions du module Loader en procédant comme suit :

1. Créez un nouveau fichier de propriétés de personnalisation de Loader.



Le fichier `MassLoadCustomizer.properties` se trouve dans l'archive `MassLoader.zip`. Extrayez ce fichier, renommez-le en gardant l'extension `.properties` et placez-le dans un répertoire figurant dans le chemin de classe. **Important :** Ne supprimez pas et ne modifiez pas le fichier `MassLoadCustomizer.properties` existant.



Le fichier `ISeries_LODWCSDTA_Customizer.properties` se trouve dans le répertoire `/QIBM/ProdData/WebCommerce/properties`. Copiez ce fichier dans le répertoire `/instroot/xml`, renommez le nouveau fichier en gardant l'extension `.properties`, puis apportez toutes les modifications nécessaires au nouveau fichier. **Important :** Ne supprimez pas et ne modifiez pas le fichier `ISeries_LODWCSDTA_Customizer.properties` d'origine.

2. Modifiez les valeurs des propriétés spécifiées dans le nouveau de propriétés de personnalisation de Loader.
3. Indiquez le nom du nouveau fichier comme valeur du paramètre de personnalisation de l'instruction Load.

## Non prise en compte des éléments du fichier d'entrée

Si votre fichier d'entrée contient des éléments qui ne se mappent pas dans la base de données cible, vous pouvez définir Loader pour qu'il ignore ces éléments dans le fichier de propriétés de personnalisation de Loader. Utilisez `IgnoreElements` pour spécifier les éléments à ignorer et séparez ces éléments par un point-virgule (;). Par exemple, pour ignorer les éléments `import`, `literals` et `ProductRepository`, spécifiez les informations suivantes dans le fichier de propriétés de personnalisation de Loader :

```
IgnoreElements = import;literals;ProductRepository
```

## Insertion de la valeur NULL dans une colonne

Vous pouvez paramétrer Loader pour qu'il insère la valeur NULL dans une colonne en affectant la valeur "true" à la propriété `EnableNULLCheck` dans le fichier de propriétés de personnalisation de Loader. Par exemple :

```
EnableNULLCheck = true
```

Pour des raisons de performances, cette fonction est désactivée par défaut.

Utilisez la propriété `NULLStringLiteral` pour déterminer la représentation de chaîne d'une valeur Null dans vos données. Par exemple, pour paramétrer Loader de telle façon que la chaîne "-" soit utilisée pour représenter une valeur Null, définissez la propriété et la valeur suivantes dans le fichier de propriétés de personnalisation de Loader :

```
NULLStringLiteral = -
```

Par défaut, la valeur de cette propriété est "NULL" (sans les guillemets).

## Chargement de données d'horodatage et de date

Loader peut charger des données dans des colonnes avec les types de données horodatage et date. Les formats de données utilisés pour les données d'horodatage et de date du document sont déterminés par des masques qui peuvent être personnalisés. L'utilisateur peut modifier un masque existant ou ajouter des masques à la liste de masques existante.

Les données correspondant à un horodatage ou une date sont comparées aux masques disponibles. Le premier masque correspondant aux données est utilisé pour convertir les données au format d'horodatage cible avant de les charger dans la base de données.

Il y a deux masques d'horodatage de sortie personnalisables, `TimeStampFormat.JDBC` et `TimeStampFormat.Load`.

1. `TimeStampFormat.JDBC` est utilisé lorsque Loader utilise des connexions JDBC pour effectuer une opération.  
Les méthodes d'importation SQL et de suppression de Loader utilisent les connexions JDBC pour la mise à jour de la base de données.
2. `TimeStampFormat.Load` est utilisé lorsque Loader utilise les utilitaires natifs.  
Les méthodes d'importation et de chargement de Loader recourent à des utilitaires natifs.

Vous pouvez personnaliser les formats d'horodatage en modifiant ou en ajoutant des masques dans le fichier de propriétés de personnalisation de Loader.

Les masques d'horodatage d'entrée suivants sont fournis :

```
InputTimeStampFormat.1 = dd/yyyy hh:mm:ss:SSSSSS
InputTimeStampFormat.2 = DD/MM/yyyy hh:mm:ss:SSSSSS
InputTimeStampFormat.3 = dd/yyyy/hh:mm:ss:SSSSSS
InputTimeStampFormat.4 = DD/MM/yyyy/HH:mm:ss:SSSSSS
InputTimeStampFormat.5 = dd/MM/yyyy/hh:mm:ss:SSSSSS
InputTimeStampFormat.6 = dd/MM/yyyy HH:mm:ss:SSSSSS
InputTimeStampFormat.7 = DD/yyyy HH:mm:ss:SSSSSS
```

Les masques par défaut des formats de date d'entrée sont les suivants :

```
InputDateFormat.1 = MM-dd-yyyy
InputDateFormat.2 = yyyy-dd-MM
InputDateFormat.3 = dd/MM/yyyy
InputDateFormat.4 = MM/dd/yyyy
InputDateFormat.5 = yyyy/dd/MM
InputDateFormat.6 = DD/yyyy
```

Vous pouvez modifier ces masques d'horodatage et de date ou ajouter autant de masques que vous voulez. Spécifiez ces masques dans le fichier de propriétés de personnalisation de Loader dans l'ordre numérique selon lequel vous voulez qu'ils soient comparés avec l'horodatage d'entrée. Si vous ajoutez un horodatage d'entrée, vous devez utiliser le nombre suivant dans la séquence en cours. (Par exemple, le prochain masque d'horodatage d'entrée serait `InputTimeStampFormat.8` si vous ajoutiez un masque à la liste ci-dessus).

Pour le formatage des données d'entrée d'horodatage et de date dans la sortie, les masques sont les suivants :

```
TimeStampFormat.JDBC = DD/MM/yyyy hh:mm:ss:SSSSSS
TimeStampFormat.Load = dd/MM/yyyy/hh:mm:ss:SSSSSS

DateFormat.JDBC = dd/MM/yyyy
DateFormat.Load = dd/MM/yyyy
```

En règle générale, les formats de date et d'horodatage de sortie ne sont *pas* personnalisés.

## Chargement des horodatages en cours

Loader peut insérer des valeurs dans des colonnes avec le type de données horodatage en fonction de la lecture d'une horloge système. Par exemple, dans WebSphere Commerce, les paramètres `STARTDATE` et `ENDDATE` d'une offre peuvent avoir des valeurs basées sur l'heure à laquelle l'offre a été insérée dans la table. Pour prendre en charge cette fonctionnalité, Loader utilise la table `MLTIME` pour conserver les instances d'horodatage. Le schéma de cette table est le suivant :

```
table MLTIME
(
  INSTANCEID BIGINT not null,
  MLTIMESTAMP TIMESTAMP
)
```

Vous pouvez personnaliser le nom de la table et ses colonnes en modifiant les propriétés suivantes dans le fichier de propriétés de personnalisation de Loader :

```
TimestampTableName = MLTIME
TimestampIdColumn = INSTANCEID
TimestampValueColumn = MLTIMESTAMP
```

Les données d'entrée spécifiant les valeurs d'horodatage en cours sont basées sur des masques de chaîne d'horodatage. Les masques suivants sont utilisés pour spécifier la durée de l'horodatage :

%D pour jours  
%M pour mois  
%Y pour années  
%H pour heures  
%m pour minutes  
%s pour secondes

Vous pouvez personnaliser les formats d'horodatage en cours en modifiant ou en ajoutant des masques dans le fichier de propriétés de personnalisation de Loader. Les masques d'entrée suivants sont fournis :

```
InputCurrentTimestampFormat.1 = CURRENT TIMESTAMP
InputCurrentTimestampFormat.2 = CURRENT TIMESTAMP %D DAYS
InputCurrentTimestampFormat.3 = CURRENT TIMESTAMP %D DAYS %M MONTHS
InputCurrentTimestampFormat.4 = CURRENT TIMESTAMP %D DAYS %M MONTHS %Y YEARS
InputCurrentTimestampFormat.5 = CURRENT TIMESTAMP %Y YEARS %M MONTHS %D DAYS
InputCurrentTimestampFormat.6 = SYSDATE
InputCurrentTimestampFormat.7 = ADDDAYS(SYSDATE,%D)
InputCurrentTimestampFormat.8 = ADDDAYS(ADDMONTHS(SYSDATE,%M),%D)
InputCurrentTimestampFormat.9 = ADDDAYS(ADDMONTHS(ADDEYEARS(SYSDATE,%Y),%M),%D)
```

Les données d'entrée indiquant l'horodatage en cours sont comparées aux masques spécifiés. Si les données correspondent à un masque d'entrée spécifié, celui-ci est utilisé pour analyser les données d'entrée et Loader convertit les données dans le format de sortie approprié avant de les insérer dans la base de données. De nouveaux masques peuvent être ajoutés dans la liste ci-dessus pourvu que les numéros d'indice soient classés de façon séquentielle.

Il y a deux formats de sortie cible pour spécifier les horodatages en cours :

1. CurrentTimestampFormat.Load est employé lorsque Loader fonctionne en mode chargement ou importation.
2. CurrentTimestampFormat.JDBC est employé lorsque Loader utilise JDBC pour insérer, mettre à jour ou supprimer des valeurs dans la base de données.

Les masques cible par défaut de Loader sont les suivants :

```
CurrentTimestampFormat.Load = CURRENT TIMESTAMP %Y YEARS %M MONTHS %D DAYS
                              %h HOURS %m MINUTES %s SECONDS
CurrentTimestampFormat.JDBC = CURRENT TIMESTAMP %Y YEARS %M MONTHS %D DAYS
                              %h HOURS %m MINUTES %s SECONDS
```

Les propriétés associées à ces masques peuvent également être personnalisées dans le fichier de propriétés de personnalisation de Loader. Lorsque vous personnalisez les propriétés CurrentTimestampFormat.Load et CurrentTimestampFormat.JDBC, vous devez vous assurer que la syntaxe de l'instruction obtenue est valide pour le système de gestion de la base de données concernée.

La propriété CurrentTimestampLiteral est utilisée par Loader pour effectuer une détermination précoce afin de savoir si la valeur de la colonne d'horodatage est dans un format d'horodatage en cours, évitant ainsi des calculs coûteux pour déterminer que la valeur n'est pas une représentation de chaîne d'horodatage.

```
CurrentTimestampLiteral = CURRENT TIMESTAMP
```



La valeur par défaut de cette propriété pour DB2 est CURRENT  
TIMESTAMP.



La valeur par défaut pour la base de données Oracle est SYSDATE.

### Exemple de chargement d'horodatages en cours

Les informations ci-après sont fournies à Loader pour qu'il mette à jour l'offre ayant l'identificateur OFFER\_ID 10123. "CURRENT TIMESTAMP" est la valeur de la date de début et "CURRENT TIMESTAMP + 14 DAYS" est celle de la date de fin.

```
<OFFER
  OFFER_ID="10123"
  STARTDATE="CURRENT TIMESTAMP">
  ENDDATE="CURRENT TIMESTAMP + 14 DAYS"
/>
```

Loader reconnaît que les colonnes STARTDATE et ENDDATE ont le type de données horodatage dans la base de données. Conformément à la propriété CurrentTimeStampLiteral, les valeurs des colonnes sont déterminées pour avoir les valeurs spécifiées dans le format d'horodatage en cours. La valeur de STARTDATE correspond au masque InputCurrentTimeStampFormat.1 et elle est convertie selon le masque spécifié par la propriété CurrentTimeStampFormat.JDBC. La valeur de ENDDATE correspond au format de la propriété InputCurrentTimeStampFormat.2 et elle est également convertie selon le masque spécifié par la propriété CurrentTimeStampFormat.JDBC.

### Exemples d'ajout de durées à des horodatages en cours

Loader vous donne la possibilité d'ajouter des durées à des horodatages en cours. Par exemple, vous pouvez vouloir charger une offre sans entrer de date spécifique. Pour ce faire, vous devez créer une date de fin éloignée d'une certaine durée de la date de début. L'exemple suivant fonctionne bien sous DB2 :

```
<Offer
  Startdate="Current Timestamp"
  Enddate="Current Timestamp +14 Days +4 Months +1 Year +0 Hours
  +0 Minutes +0 Seconds"
/>
```

Pour traiter des durées d'horodatage en cours indépendantes de la plate-forme, vous devez cependant personnaliser les formats d'horodatage en cours en modifiant les masques dans le fichier de propriétés de personnalisation de Loader. Voici un exemple de spécifications de propriétés d'horodatage en cours personnalisées :

```
CurrentTimeStampLiteral=Current Timestamp

InputCurrentTimeStampFormat.0=Current Timestamp
InputCurrentTimeStampFormat.1=Current Timestamp %D Days
InputCurrentTimeStampFormat.2=Current Timestamp %M Months
InputCurrentTimeStampFormat.3=Current Timestamp %Y Years
InputCurrentTimeStampFormat.4=Current Timestamp %D Days %M Months
InputCurrentTimeStampFormat.5=Current Timestamp %D Days %M Months %Y Years
InputCurrentTimeStampFormat.5=Current Timestamp %H Hours %m Minutes %s Seconds

CurrentTimeStampFormat.JDBC=Current Timestamp %D Days %M Months %Y Years
%H Hours %m Minutes %s Seconds
```

En utilisant l'exemple d'offre et ces spécifications de propriétés, la date de fin de l'offre correspond au masque InputCurrentTimestampFormat.5. Les informations suivantes concernant l'offre sont renvoyées, en utilisant la propriété CurrentTimestampFormat.JDBC :

```
<Offer
  Startdate="Current Timestamp"
  Enddate="Current Timestamp +14 Days +4 Months +1 Year +0 Hours +0 Minutes +0 Seconds"
/>
```

L'exemple ci-dessus montre comment Loader peut entrer plusieurs formats d'horodatage en cours et formater les horodatages de façon appropriée selon le format de sortie voulu. L'exemple suivant montre comment vous pouvez gérer des formats indépendants de la plate-forme et les mapper sur des formats de sortie spécifiques de la plate-forme.

```
<Offer
  Startdate="Now"
  Enddate="Now +14D +4M +1Y"
/>
```

```
CurrentTimestampLiteral=Now
```

```
InputCurrentTimestampFormat.0=Now
InputCurrentTimestampFormat.1=Now %DD
InputCurrentTimestampFormat.2=Now %MM
InputCurrentTimestampFormat.3=Now %YY
InputCurrentTimestampFormat.4=Now %DD %MM
InputCurrentTimestampFormat.5=Now %DD %MM %YY
InputCurrentTimestampFormat.5=Sysdate %HH %mm %ss
```

```
CurrentTimestampFormat.JDBC=AddYears (AddMonths (AddDays (AddHours (AddMinutes (AddSeconds
  (Sysdate,%s),%m),%H),%D),%M),%Y)
```

**Remarque :** L'instruction ci-dessus est juste un exemple. Elle est simplement utilisée pour illustrer la fonction de personnalisation d'un système de gestion de base de données hypothétique. Elle n'est pas valide pour DB2 ou une base de données Oracle.

En utilisant l'exemple d'offre et ces spécifications de propriétés, la date de fin de l'offre correspondra au masque InputCurrentTimestampFormat.5. Les informations suivantes concernant l'offre sont renvoyées, en utilisant la propriété CurrentTimestampFormat.JDBC :

```
<Offer
  Startdate="Current Timestamp"
  Enddate="AddYears (AddMonths (AddDays (AddMinutes (AddSeconds (Sysdate,0),0),0),14),4),1)"
/>
```

## Gestion de files d'attente d'événements

Vous pouvez personnaliser les files d'attente des événements de Loader en modifiant les paramètres du fichier de propriétés de personnalisation de Loader. Par exemple :

```
QueueLowCount = 35
QueueHighCount = 90
```

La source des événements remplissant la file d'attente est bloquée lorsque le nombre d'éléments en file d'attente atteint la limite supérieure, ce qui empêche la

mise en file d'attente d'autres événements. La file d'attente accepte de nouveau des événements lorsque le nombre d'éléments en file d'attente tombe au-dessous de la limite inférieure.

## Exécution avec différents logiciels de base de données et systèmes d'exploitation

Vous pouvez personnaliser Loader pour qu'il s'exécute avec différents logiciels de base de données et systèmes d'exploitation en modifiant les paramètres des éléments suivants dans le fichier de propriétés de personnalisation de Loader afin de spécifier des logiciels de base de données et des systèmes d'exploitation différents :

- Instruction de connexion à une base de données
- Instruction de table de chargement de base de données
- Instruction d'importation de base de données
- Instruction système appelant le chargement

Pour personnaliser un de ces éléments, supprimez les doubles barres obliques (//) de mise en commentaire qui précèdent l'instruction dans le fichier de propriétés de personnalisation de Loader et modifiez les valeurs par défaut.

**Instruction de connexion à une base de données :** vous pouvez modifier les paramètres de l'instruction de connexion à une base de données si vous souhaitez modifier les valeurs par défaut (qui supposent que vous utilisez DB2).

```
DBConnectCommand = connect to {0} user {1} using {2};
```

où :

0 = nom de la base de données  
1 = utilisateur de la base de données  
2 = mot de passe utilisateur

**Instruction de table de chargement de base de données :** vous pouvez modifier les paramètres de l'instruction de table de chargement de base de données si vous souhaitez modifier les valeurs par défaut.

```
DBLoadTableCommand = load from {0} of del modified by coldel{1}  
charcodel{2} insert into {3} ({4});
```

où :

0 = nom du fichier  
1 = délimiteur de colonne  
2 = délimiteur de caractères  
3 = nom de la table  
4 = noms de colonnes, séparés par des virgules (,)

**Instruction d'importation de base de données :** vous pouvez modifier les paramètres de l'instruction d'importation de base de données si vous souhaitez modifier les valeurs par défaut.

```
DBImportCommand = import from {0} of del modified by coldel{1} charcodel{2}  
insert_update into {3} ({4});
```

où :

- 0 = nom du fichier
- 1 = délimiteur de colonne
- 2 = délimiteur de caractères
- 3 = nom de la table
- 4 = noms de colonnes, séparés par des virgules (,)

**Instruction système appelant le chargement :** vous pouvez modifier les paramètres de l’instruction système appelant le chargement si vous souhaitez modifier les valeurs par défaut. Cette instruction exécute les scripts natifs de chargement et d’importation générés par Loader.

• 

```
DBLoadCommand = db2c1pex DB2 -z {0} -astvf {1}
```

où :

- 0 = nom du fichier journal
- 1 = nom du fichier de commandes

Pour DB2 sous AIX, par exemple, la valeur de la propriété DBLoadCommand est la suivante :

```
db2 -tvf {1} -z {0}
```

•    

```
DBLoadCommand = sqlldr log={0} control={1} USERID={2}
```

où :

- 0 = nom du fichier journal
- 1 = nom du fichier de commandes
- 2 = nom de l’utilisateur de la base de données

Utilisez les paramètres suivants pour les différentes combinaisons de base de données et de système d’exploitation :

•   

Définissez la variable d’environnement système CLASSPATH de telle sorte qu’elle inclue le fichier db2/dbconnect.zip pour DB2 sous Windows NT ou Windows 2000 avec la méthode d’importation SQL, de chargement, d’importation ou de suppression.

•    

Pour DB2 dans les environnements AIX, Solaris ou Linux, procédez comme suit :

- Avec la méthode d’importation SQL, de chargement, d’importation ou de suppression, définissez la variable d’environnement système CLASSPATH de telle sorte qu’elle inclue le fichier db2/dbconnect.zip.

- Avec la méthode de chargement ou d'importation, modifiez les propriétés suivantes dans le fichier de propriétés de personnalisation de Loader :

```

/**
 * Connection command. (Default is for DB2)
 * parameter 0 = dbName
 * parameter 1 = dbUser
 * parameter 2 = userPasswd
 */

DBConnectCommand = connect to {0} user {1} using {2};

/**
 * Load Data into Table command. (Default is for DB2)
 * parameter 0 = filename
 * parameter 1 = column delimiter
 * parameter 2 = character delimiter
 * parameter 3 = name of the table
 * parameter 4 = name of the columns, separated by comma(,)s
 */

DBLoadTableCommand = load from {0} of del modified by coldel{1}
insert into {3} ({4});

/**
 * Insert Data into Table command. (Default is for DB2)
 * parameter 0 = filename
 * parameter 1 = column delimiter
 * parameter 2 = character delimiter
 * parameter 3 = name of the table
 * parameter 4 = name of the columns, separated by comma(,)s
 */

DBUpdateTableCommand = import from {0} of del modified by coldel{1}
insert_update into {3} ({4});

/**
 * System command to invoke load (Default is for DB2)
 * parameter 0 = logFileName
 * parameter 1 = commandFileName
 */

DBLoadCommand = db2 -z {0} -tf {1}

```

-  Modifiez les propriétés suivantes dans le fichier de propriétés de personnalisation de Loader pour DB2 sous iSeries avec la méthode d'importation SQL, de chargement, d'importation ou de suppression :

```

/**
 * The connect string.
 */

ConnectionStringID = jdbc:db2://

/**
 * The JDBC driver information.
 */

JDBCdriverName = com.ibm.db2.jdbc.app.DB2Driver
DbVendorName=DB2/iSeries

/**
 * Custom writer for load/import methods.
 */

WriterName=com.ibm.wca.MassLoader.Writer.ISeriesWriter

```



Définissez la variable d'environnement système CLASSPATH de telle sorte qu'elle inclue le fichier oracle/dbconnect.zip pour la base de données Oracle dans les environnements Windows NT, Windows 2000, AIX, Solaris ou Linux avec la méthode d'importation SQL, de chargement, d'importation ou de suppression.

## Substitution d'un composant

Vous pouvez remplacer un composant de Loader en affectant aux éléments suivants du fichier de propriétés de personnalisation de Loader les valeurs de classe que vous voulez substituer aux implémentations par défaut :

### ParserName

Nom de l'analyseur syntaxique à utiliser

### ValidatorName

Nom du programme de validation à utiliser

### FormatterName

Nom du programme de formatage à utiliser

### JDBCFormatterName

Nom du programme de formatage lorsque la méthode d'importation SQL est utilisée

### WriterName

Nom du programme d'écriture à utiliser

### JDBCWriterName

Nom du programme d'écriture lorsque la méthode d'importation SQL est utilisée

Pour remplacer le programme d'écriture par défaut de Loader (DefaultWriter) par le programme d'écriture com.abc.writer.SpecialWriter, par exemple, spécifiez les informations suivantes dans le fichier de propriétés de personnalisation de Loader :

```
WriterName = com.abc.writer.SpecialWriter
```

Loader va alors utiliser le programme "com.abc.writer.SpecialWriter" pour effectuer la fonction d'écriture.

## Utilisation de la synchronisation de l'espace de recherche de l'Assistant produit

Pour utiliser la synchronisation de l'espace de recherche de l'Assistant produit, procédez comme suit :

1. Créez un fichier d'informations de configuration XML pour la synchronisation, nommé "PASyncInfo.xml".
2. Dans le fichier PASyncInfo.xml, spécifiez PASync.xsd comme schéma XML à utiliser. Par exemple :

```
<PASync
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation='PASync.xsd'
```

Le fichier PASync.xsd est fourni. Le texte suivant présente le contenu du fichier PASync.xsd :

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd='http://www.w3.org/2001/XMLSchema'>

  <xsd:element name="PASync">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="SearchScheme" />
        <xsd:element ref="Command" minOccurs="1" maxOccurs="unbounded" />
      </xsd:sequence>
      <xsd:attribute name="member" type="xsd:string" use="required" />
      <xsd:attribute name="store" type="xsd:string" use="required" />
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SearchScheme">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="RelatedTable" />
        <xsd:element ref="Search" minOccurs="1" maxOccurs="unbounded" />
      </xsd:sequence>
      <xsd:attribute name="tableName" type="xsd:string" use="required" />
      <xsd:attribute name="primary" type="xsd:string" use="required" />
      <xsd:attribute name="columnName" type="xsd:string" use="required" />
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="RelatedTable">
    <xsd:complexType>
      <xsd:attribute name="tableName" type="xsd:string" use="required" />
      <xsd:attribute name="from" type="xsd:string" use="required" />
      <xsd:attribute name="to" type="xsd:string" use="required" />
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="Search">
    <xsd:complexType>
      <xsd:attribute name="value" type="xsd:string" use="required" />
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="Command">
    <xsd:complexType>
      <xsd:attribute name="tableName" type="xsd:string" use="required" />
      <xsd:attribute name="idColumnName" type="xsd:string" use="required" />
      <xsd:attribute name="addCommand" type="xsd:string" />
      <xsd:attribute name="updateCommand" type="xsd:string" />
      <xsd:attribute name="deleteCommand" type="xsd:string" />
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

3. Dans le fichier PASyncInfo.xml, spécifiez l'ID de membre et l'ID d'entité magasin pour lesquels la synchronisation doit être réalisée. Par exemple :

```
member = "-2000"
store = "10351"
```

4. Sous l'élément du schéma de recherche, dans le fichier PASyncInfo.xml, spécifiez les identificateurs CATGROUP qui délimitent l'espace de recherche. Par exemple :

```
<SearchScheme
  tableName = "catgroup"
  primary = "CATGROUP_ID"
  columnName = "identifiant" >
```

```

<RelatedTable
  tableName = "catgpenrel"
  from = "CATGROUP_ID"
  to = "CATENTRY_ID" />

<Search value="Pantalons" />
<Search value="Chemises" />

</SearchScheme>

```

Les identificateurs "Pantalons" et "Chemises" sont spécifiés dans cet exemple. Vous pouvez spécifier autant d'identificateurs CATGROUP que vous voulez.

5. Dans le fichier PASyncInfo.xml, spécifiez des attributs pour déterminer quelles instructions doivent être programmées. Par exemple :

```

<Command tableName = "CATENTRY" idColumnName = "CATENTRY_ID"
  updateCommand = "UpdateSearchSpaces"
  deleteCommand = "RemoveProductsFromAllSearchSpaces"
/>

<Command tableName = "CATENTDESC" idColumnName = "CATENTRY_ID"
  addCommand = "UpdateSearchSpaces"
  updateCommand = "UpdateSearchSpaces"
  deleteCommand = "UpdateSearchSpaces"
/>

<Command tableName = "LISTPRICE" idColumnName = "CATENTRY_ID"
  addCommand = "UpdateSearchSpaces"
  updateCommand = "UpdateSearchSpaces"
  deleteCommand = "UpdateSearchSpaces"
/>

<Command tableName = "ATTRVALUE" idColumnName = "CATENTRY_ID"
  addCommand = "UpdateSearchSpaces"
  updateCommand = "UpdateSearchSpaces"
  deleteCommand = "UpdateSearchSpaces"
/>

<Command tableName = "CATENTATTR" idColumnName = "CATENTRY_ID"
  addCommand = "UpdateSearchSpaces"
  updateCommand = "UpdateSearchSpaces"
  deleteCommand = "UpdateSearchSpaces"
/>

<Command tableName = "CATGPENREL" idColumnName = "CATENTRY_ID"
  addCommand = "AddProductsToSearchSpace"
  deleteCommand = "RemoveProductsFromSearchSpace"
/>

</PASync>

```

6. Dans le nouveau fichier de propriétés de personnalisation de Loader, spécifiez le fichier d'informations de configuration XML. Par exemple :  
PASyncDocumentURL = PASyncInfo.xml
7. Dans le nouveau fichier de propriétés de personnalisation de Loader, activez la synchronisation. Par exemple :  
PASyncEnabled = true
8. Utilisez la méthode d'importation SQL ou la méthode de suppression avec l'instruction Load. Voici un exemple d'entrée XML appropriée pour Loader :

```

<store-asset>

  <catentry
    CATENTRY_ID="10351"
    MEMBER_ID="-2000"
  />

```

```

PARTNUMBER="000051"
CATENTTYPE_ID="ProductBean"
MFPARTNUMBER="m000051"
MARKFORDELETE="0"
BUYABLE="1"
/>

<catentry
  CATENTRY_ID="10352"
  MEMBER_ID="-2000"
  PARTNUMBER="000052"
  CATENTTYPE_ID="ProductBean"
  MFPARTNUMBER="m000052"
  MARKFORDELETE="0"
  BUYABLE="1"
/>

<catentry
  CATENTRY_ID="10353"
  MEMBER_ID="-2000"
  PARTNUMBER="000053"
  CATENTTYPE_ID="ProductBean"
  MFPARTNUMBER="m000053"
  MARKFORDELETE="0"
  BUYABLE="1"
/>

<catentry
  CATENTRY_ID="10358"
  MEMBER_ID="-2000"
  PARTNUMBER="000058"
  CATENTTYPE_ID="ProductBean"
  MFPARTNUMBER="m000058"
  MARKFORDELETE="0"
  BUYABLE="1"
/>

<catentry
  CATENTRY_ID="10365"
  MEMBER_ID="-2000"
  PARTNUMBER="000065"
  CATENTTYPE_ID="ProductBean"
  MFPARTNUMBER="m000065"
  MARKFORDELETE="0"
  BUYABLE="1"
/>

<catentry
  CATENTRY_ID="10372"
  MEMBER_ID="-2000"
  PARTNUMBER="000072"
  CATENTTYPE_ID="ProductBean"
  MFPARTNUMBER="m000072"
  MARKFORDELETE="0"
  BUYABLE="1"
/>

<catgenre1
  CATGROUP_ID="10354"
  CATENTRY_ID="10372"
  CATALOG_ID="10351"
  SEQUENCE="3"
/>

<catgenre1
  CATGROUP_ID="10354"
  CATENTRY_ID="10365"

```

```

        CATALOG_ID="10351"
        SEQUENCE="4"
    />

    <catgpenrel
        CATGROUP_ID="10354"
        CATENTRY_ID="10358"
        CATALOG_ID="10351"
        SEQUENCE="5"
    />

    <catgpenrel
        CATGROUP_ID="10355"
        CATENTRY_ID="10372"
        CATALOG_ID="10351"
        SEQUENCE="3"
    />

</store-asset>

```

**Remarque :** La désactivation de la synchronisation de l'espace de recherche de l'Assistant produit permet d'obtenir de meilleures performances du programme Loader ; par conséquent, n'utilisez cette fonction qu'en cas de nécessité.

## Personnalisation de la synchronisation de l'espace de recherche de l'Assistant produit

Le module Loader vous permet de personnaliser la synchronisation de l'espace de recherche de l'Assistant produit en modifiant le fichier de propriétés de personnalisation de Loader pour effectuer les opérations suivantes :

- **Activer ou désactiver la synchronisation**

Vous pouvez activer ou désactiver la synchronisation en affectant la valeur true ou false à la propriété suivante dans le fichier de propriétés de personnalisation de Loader :

```
PASyncEnabled = true
```

- **Spécifier le fichier d'informations de configuration pour la synchronisation**

Vous pouvez spécifier quel fichier d'informations de configuration XML la synchronisation doit utiliser en définissant la valeur de la propriété suivante dans le fichier de propriétés de personnalisation de Loader :

```
PASyncDocumentURL = PASyncInfo.xml
```

- **Spécifier la longueur de la requête de programmation**

Vous pouvez spécifier la longueur de la requête de programmation en définissant la valeur de la propriété suivante dans le fichier de propriétés de personnalisation de Loader :

```
PAScheduleQueryLength = 30
```

La valeur de cette propriété doit être comprise entre 20 et 900.

- **Spécifier l'heure de début programmée**

Vous pouvez spécifier l'heure de début programmée en spécifiant un horodatage absolu, un horodatage en cours ou un horodatage en cours avec durée comme valeur de la propriété PAScheduledStartTime dans le fichier de propriétés de personnalisation de Loader.

**Remarque :** Le format de l'horodatage doit être approprié pour votre base de données.

Voici un exemple, sous DB2, qui exécutera le travail programmé 5 minutes après le chargement :

```
PAScheduledStartTime = CURRENT TIMESTAMP + 5 MINUTES
```



Voici un exemple, sous Oracle, qui exécutera le travail immédiatement :

```
PAScheduledStartTime = SYSDATE
```

---

## Détermination de la façon de traiter les données en utilisant le programme de chargement Loader

Le programme Loader offre les options suivantes pour le traitement des données en utilisant l'instruction Load :

- Chargement
- Importation
- Utilisation de la fonction d'importation SQL

Avant de charger des données, vous devez déterminer quelle méthode de traitement donnerait les meilleurs résultats.

### Choix de la méthode de chargement

Recourez à la méthode de chargement dans les situations suivantes :

- Si vous savez que les données sont propres et si la base de données ne contient aucune donnée.
- Si vous savez que les données sont propres et que la base de données ne contient pas les données qui sont en cours de chargement.
- Si vous savez que les données sont propres, si les tables cible ne contiennent aucune clé primaire et si vous savez que la base de données ne contient pas les données qui sont en cours de chargement.
- Si le temps de chargement est votre principal souci.
- Si la base de données est une base DB2 locale.



Avec la méthode de chargement, les données sont chargées dans la base de données. Si les données existent déjà, l'instruction échoue en raison d'une erreur de clé en double, et un message d'erreur de duplication s'affiche.

### Choix de la méthode d'importation



Avec la méthode d'importation pour DB2, les données sont également chargées dans la base de données. Si elles existent déjà, elles ne sont pas supprimées, mais mises à jour avec les nouvelles valeurs. Recourez à cette méthode dans les situations suivantes :

- Si le système de gestion de base de données est DB2.
- Si vous ne savez pas si les données sont propres.
- Si vous devez mettre à jour de grandes quantités de données homogènes au niveau d'une colonne.
- Si le temps de chargement n'est pas votre principal souci.
- Si la table dans laquelle les données sont importées comporte des clés primaires.

Avec la méthode d'importation, les données sont également chargées dans la base de données. Si elles existent déjà, elles ne sont pas supprimées, mais mises à jour avec les nouvelles valeurs. Recourez à cette méthode dans les situations suivantes :

- Si vous ne savez pas si les données sont propres.
- Si les données existent déjà dans la base de données.
- Si le temps de chargement n'est pas votre principal souci.
- Si la table dans laquelle les données sont importées comporte des clés primaires.

## Choix de la méthode d'importation SQL

Avec la méthode d'importation SQL, des instructions JDBC ou SQL sont utilisées pour mettre à jour ou insérer des données dans la base de données. Les données sont insérées si elles n'existent pas déjà et les données existantes sont mises à jour. Recourez à cette méthode dans les situations suivantes :

- Si vous mettez à jour des données existantes et avez besoin de mises à jour au niveau des colonnes.

Avec cette méthode, le relevé des erreurs relatives aux violations de contraintes et des erreurs de type de données est meilleur.

- Si vous savez que certaines des données ne sont pas propres.
- Si l'intégrité de la base de données est votre principal souci.
- Si la base de données n'est pas locale.
- Si vous utilisez la synchronisation de l'espace de recherche de l'Assistant produit.

## Autres considérations

### • Restrictions sur l'utilisation de la méthode de chargement

La méthode de chargement ne peut pas insérer ni mettre à jour des données dans les zones de données binaires.

**DB2**

Avec la méthode de chargement, seuls les nouveaux enregistrements sont insérés dans la base de données ; les enregistrements existants ne sont pas mis à jour.

**DB2**

La méthode de chargement ne peut être utilisée que pour les bases de données DB2 locales (et non éloignées).

### • Restrictions sur l'utilisation de la méthode d'importation

Le système de gestion de base de données doit être DB2 pour que la méthode d'importation puisse être utilisée.

La méthode d'importation ne peut pas insérer ni mettre à jour des données dans les zones de données binaires.

Avec la méthode d'importation, le programme Loader n'insère ou ne met à jour des données que dans les tables comportant des clés primaires définies ; cette méthode ne peut pas insérer ou mettre à jour des données dans des tables n'ayant pas de clé primaire. Si l'enregistrement d'entrée contient uniquement des valeurs pour des colonnes primaires, il est rejeté.

### • Comparaison des méthodes de chargement et d'importation SQL

La méthode d'importation SQL vérifie la cohérence des données, y compris les références associées, et elle vous permet de mettre à jour les données existantes, ce que ne fait pas la méthode de chargement.

- **Comparaison des méthodes d'importation et d'importation SQL**

Les méthodes d'importation et d'importation SQL remplissent des fonctions similaires. La méthode d'importation est généralement plus rapide mais nécessite de l'espace disque pour des fichiers temporaires.

La méthode d'importation ne peut insérer ou mettre à jour des données que dans les tables comportant des clés primaires définies, par opposition à la méthode d'importation SQL qui n'impose pas cette condition.

- **Comparaison de méthodes basée sur le produit de base de données utilisé**

Les méthodes d'importation et de chargement utilisent des utilitaires natifs optimisés pour DB2, tandis que la méthode d'importation SQL utilise des appels JDBC, communs à de nombreux produits de base de données.

- **Autres considérations**

La méthode de suppression est utilisée pour supprimer de la base de données les données se trouvant dans le document XML d'entrée. L'élément doit contenir les valeurs de la clé primaire ou de l'index à entrées uniques de la table. Si les données en cours de suppression ont des interdépendances avec des données d'une autre table pour laquelle la suppression en cascade ("cascade on delete") est activée, les données dépendantes sont également supprimées.

Si vous utilisez la synchronisation de l'espace de recherche de l'Assistant produit, vous devez utiliser la méthode d'importation SQL pour charger des données.

---

## Chargement de documents volumineux

Lorsque vous utilisez les utilitaires du module Loader pour charger des documents volumineux dans une base de données, vous devez prendre en compte les éléments suivants :

- **Taille de segment de mémoire JVM (Java Virtual Machine)**

Par défaut, le volume de mémoire maximal alloué au segment JVM est de 64 Mo. Si ce volume n'est pas augmenté, la machine virtuelle JVM risque de ne pas disposer de suffisamment de mémoire lors du processus de chargement. Le volume de mémoire maximal alloué au segment JVM peut être adapté en utilisant l'option JVM -mx dans l'instruction Java.

- **Consignation de trace**

Le programme de consignation de trace peut saturer le segment JVM lors du chargement d'un document XML volumineux. Les informations de trace sont le plus souvent utilisées pour le débogage d'une exécution si celle-ci échoue. Si le traçage du processus de chargement n'est pas nécessaire, la fonction de trace doit être désactivée. La désactivation de la fonction de trace entraîne une amélioration significative des performances. Cette fonction est désactivée en modifiant le document XML de configuration du programme de consignation.

Le fichier de configuration du programme de consignation par défaut est WCALoggerConfig.xml. Pour désactiver la consignation de trace, modifiez la configuration du programme de consignation de trace pour Loader, en remplaçant :

```
<logger type="trace">
  <handler type="file">
    <filePath>MassLoadTrace.log</filePath>
    <filter type="Any">
      <messageType name="PUBLIC" />
    </filter>
  </handler>
</logger>
```

par :

```
<logger type="trace">
  <handler type="file">
    <filePath>MassLoadTrace.log</filePath>
    <filter type="Any">
      </filter>
    </handler>
  </logger>
```

Pour plus d'informations sur la modification du fichier `WCALoggerConfig.xml`, reportez-vous à la section «Personnalisation de la journalisation pour le module Loader» à la page 64.

- **Nombre de validations**

Le nombre de validations par défaut pour le programme de chargement Loader, lorsqu'il fonctionne en mode d'importation SQL, est défini à 1. Par défaut, les transactions sont donc validées pour chaque mise à jour ou insertion dans la base de données. Pour améliorer les performances du programme de chargement Loader dans le cas de documents volumineux, le nombre de validations doit être augmenté. Une valeur de "100" est suggérée, mais cette valeur peut être supérieure en fonction de la quantité de mémoire physique sur le serveur, de la taille du journal de transactions DBMS, etc.

Le nombre de validations défini pour le programme Loader est modifié en utilisant l'option `-commitcount nombre` pour l'instruction Load (où *nombre* est le nombre d'instructions exécutées avant la validation de la transaction).

## Conseils d'identification et de résolution des incidents

Si l'exécution du programme est anormalement lente lors du chargement des données, il se peut que le programme de consignation de Loader utilise un gestionnaire de fichiers mal configuré. Cela peut être dû à une des situations suivantes :

- L'utilisateur qui appelle Loader ne dispose pas des droits suffisants pour écrire dans le répertoire ou pour mettre à jour le fichier indiqué dans le document de configuration du programme de consignation.
- Le répertoire indiqué comme emplacement du fichier dans le document de configuration du programme de consignation n'existe pas.
- L'unité indiquée comme emplacement du fichier dans le document de configuration du programme de consignation n'existe pas.

Si vous corrigez un de ces incidents, vous devrez changer l'emplacement de fichier indiqué en modifiant le document de configuration du programme de consignation ( `WCALoggerConfig.xml` par défaut). Pour plus d'informations sur les gestionnaires de fichiers et le fichier `WCALoggerConfig.xml`, reportez-vous à la section «Personnalisation de la journalisation pour le module Loader» à la page 64.

---

## Chapitre 7. Extraction de données

Pour extraire des données d'une base de données à l'aide de l'utilitaire Extractor, vous devez spécifier les données que vous voulez extraire de cette base en utilisant un fichier de filtres d'extraction. Le filtre d'extraction que vous utilisez dépend du type de données que vous voulez extraire.

L'exemple suivant extrait des données du sous-système membres d'une base de données en utilisant le fichier MemberSubsystemFilter.xml comme filtre d'extraction.

- 

```
java com.ibm.wca.MassExtract.Extract -filter MemberSubsystemFilter.xml
-outfile MemberSubsystemExtracted.xml -dbname mall -dbuser myname
-dbpwd mypassword -customizer MemberSubsystemCustomizer
```
- 

```
QWBCOMM/EXTWCSDTA FILTER(MemberSubsystemFilter.xml)
OUTFILE(MemberSubsystemExtracted.xml) DATABASE(database_name)
SCHEMA(mall) INSTRROOT(/QIBM/UserData/WebCommerce/instances/mser)
PASSWD(mypassword)
```

---

### Création d'un filtre d'extraction

L'exemple suivant est un filtre d'extraction qui extrait des informations de catégorie et de produit des tables CATGROUP, CATGRPDASC, CATGRPDEL, CATENTRY, CATENTSHIP, OFFER, CATENTREL, CATGPENREL, CATENTDESC et ATTRVALUE :

```
<sqlx>

<!-- ***** -->
<!-- extract Category information -->
<!-- ***** -->

<functionDef id="Category" description="Extract Categories" schemaentity="catgroup">
  <paramDef name=":lastRecord" type="string" value="10301" description="Last record
  before loading new data" />
  <body>
    select * from catgroup where catgroup_id > :lastRecord
  </body>
</functionDef>

<execute id="Category" description="Extract Categories" schemaentity="catgroup">
  <param name=":lastRecord" type="string" value="10300" description="Last record
  before loading new data" />
</execute>

<functionDef id="Category Description" description="Extract Category Descriptions
for a Locale" schemaentity="catgrpdesc">
  <paramDef name=":lastRecord" type="string" value="10300" description="Last record
  before loading new data" />
  <body>
    select * from catgrpdesc where catgroup_id > :lastRecord
  </body>
</functionDef>

<execute id="Category Description" description="Extract Category Descriptions
for a Locale" schemaentity="catgrpdesc">
  <param name=":lastRecord" type="string" value="10300" description="Last record
  before loading new data" />
```

```

</execute>

<functionDef id="Category Relationship" description="Extract Category-Relations
for a Locale" schemaentity="catgrprel">
  <paramDef name=":lastRecord" type="string" value="10300" description="Last record
before loading new data" />
  <body>
    select * from catgrprel where catgroup_id_child > :lastRecord
  </body>
</functionDef>

<execute id="Category Relationship" description="Extract Category-Relations for
a Locale" schemaentity="catgrprel">
  <param name=":lastRecord" type="string" value="10300" description="Last record
before loading new data" />
</execute>

<!-- ***** -->
<!-- extract Product information -->
<!-- ***** -->

<functionDef id="Product" description="Extract Product" schemaentity="catentry">
  <paramDef name=":lastrecord" type="string" value="10300" description="Last record
before loading new data" />
  <body>
    select * from catentry where catentry_id > :lastrecord
  </body>
</functionDef>

<execute id="Product" description="Extract Product" schemaentity="catentry">
  <param name=":lastrecord" type="string" value="10300" description="Last record
before loading new data" />
</execute>

<functionDef id="Product Relationship" description="Extract Product Ship
information" schemaentity="catentrel">
  <paramDef name=":lastrecord" type="string" value="10300" description="Last record
before loading new data" />
  <body>
    select * from catentrel where catentry_id_child > :lastrecord
  </body>
</functionDef>

<execute id="Product Relationship" description="Extract Product Ship information"
schemaentity="catentrel">
  <param name=":lastrecord" type="string" value="10300" description="Last record
before loading new data" />
</execute>

<functionDef id="Product Description" description="Extract Product Description"
schemaentity="catentdesc">
  <paramDef name=":lastrecord" type="string" value="10300" description="Last record
before loading new data" />
  <body>
    select * from catentdesc where catentry_id > :lastrecord
  </body>
</functionDef>

<execute id="Product Description" description="Extract Product Description"
schemaentity="catentdesc">
  <param name=":lastrecord" type="string" value="10300" description="Last record
before loading new data" />
</execute>

<functionDef id="Product Ship" description="Extract Product Ship information"
schemaentity="catentship">
  <paramDef name=":lastrecord" type="string" value="10300" description="Last record
before loading new data" />
  <body>
    select * from catentship where catentry_id > :lastrecord
  </body>
</functionDef>

```

```

<execute id="Product Ship" description="Extract Product Ship information"
  schemaentity="catentship">
  <param name=":lastrecord" type="string" value="10300" description="Last record
  before loading new data" />
</execute>

<functionDef id="Category Product Relationship" description="Extract Category
Product Relations" schemaentity="catgpenrel">
  <paramDef name=":lastrecord" type="string" value="10300" description="Last record
  before loading new data" />
  <body>
    select * from catgpenrel where catgroup_id > :lastrecord
  </body>
</functionDef>

<execute id="Category Product Relationship" description="Extract Category Product
Relations" schemaentity="catgpenrel">
  <param name=":lastrecord" type="string" value="10300" description="Last record
  before loading new data" />
</execute>

<!-- ***** -->
<!-- Extract Product Attribute Information -->
<!-- ***** -->

<functionDef id="Product Attribute Values" description="Extract Product Attribute
values for a Locale" schemaentity="attrvalue">
  <paramDef name=":lastrecord" type="string" value="10300" description="Last record
  before loading new data" />
  <body>
    select * from attrvalue where catentry_id > :lastrecord
  </body>
</functionDef>

<execute id="Product Attribute Values" description="Extract Product Attribute values
for a Locale" schemaentity="attrvalue">
  <param name=":lastrecord" type="string" value="10300" description="Last record
  before loading new data" />
</execute>

<!-- ***** -->
<!-- Extract Product Price Information -->
<!-- ***** -->

<functionDef id="Offer" description="Extract Offer" schemaentity="offer">
  <paramDef name=":lastrecord" type="string" value="10300" description="Last record
  before loading new data" />
  <body>
    select * from offer where catentry_id > :lastrecord
  </body>
</functionDef>

<execute id="Offer" description="Extract Offer" schemaentity="offer">
  <param name=":lastrecord" type="string" value="10300" description="Last record
  before loading new data" />
</execute>

</sqlx>

```

## Configuration de l'utilitaire Extractor

Vous pouvez modifier les pilotes de base de données utilisés par Extractor en procédant comme suit :

1. Créez un nouveau fichier de propriétés de personnalisation d'Extractor.



Le fichier `DB2ConnectionCustomizer.properties` se trouve dans l'archive `MassExtract.zip`. Extrayez ce fichier, renommez-le en gardant l'extension `.properties` et placez-le dans un répertoire figurant dans le chemin de classe. **Important** : Ne supprimez pas et ne modifiez pas le fichier `DB2ConnectionCustomizer.properties` existant.



Le fichier `ISeries_EXTWCSDTA_Customizer.properties` se trouve dans le répertoire `/QIBM/ProdData/WebCommerce/properties`. Copiez ce fichier dans le répertoire `/instroot/xml`, renommez le nouveau fichier en gardant l'extension `.properties`, puis apportez toutes les modifications nécessaires au nouveau fichier. **Important** : Ne supprimez pas et ne modifiez pas le fichier `ISeries_EXTWCSDTA_Customizer.properties` d'origine.

2. Modifiez les valeurs des pilotes de base de données dans le nouveau fichier.
3. Indiquez le nom du nouveau fichier comme valeur du paramètre de personnalisation de l'instruction `Extract`.

Voici un extrait du fichier de propriétés de personnalisation d'Extractor :

```
DBVendorName = DB2
DBDriverName = COM.ibm.db2.jdbc.app.DB2Driver
DBURL = jdbc:db2:
```

où :

- `DBVendorName` est utilisé pour sélectionner le type de base de données. Les options disponibles sont les suivantes :
  - DB2 Universal Database for iSeries (DB2/iSeries)
  - DB2 pour les autres systèmes d'exploitation (DB2)
  - Base de données Oracle (Oracle)
- `DBDriverName` est utilisé pour sélectionner le pilote JDBC. Les options disponibles sont les suivantes :
  - DB2 Universal Database for iSeries (`com.ibm.db2.jdbc.app.DB2Driver`)
  - DB2 pour les autres systèmes d'exploitation (`COM.ibm.db2.jdbc.app.DB2Driver`)
  - Base de données Oracle (`oracle.jdbc.driver.OracleDriver`)
- `DBURL` est utilisé pour sélectionner l'URL d'accès à la base de données. Les options disponibles sont les suivantes :
  - DB2 Universal Database for iSeries (`jdbc:db2://`)
  - DB2 pour les autres systèmes d'exploitation (`jdbc:db2:`)
  - Base de données Oracle (`jdbc:oracle:oci8:@`)

---

## Chapitre 8. Utilisation du programme de consignation du module Loader

Chaque utilitaire du module Loader crée des messages pour signaler un succès, un échec ou une erreur, ainsi que pour fournir des informations de trace du programme.

Les utilitaires du module Loader font référence au fichier `WCALoggerConfig.xml`.

- 

Ce fichier se trouve dans un répertoire spécifié dans la variable d'environnement système `CLASSPATH` (chemin de classe). Il peut également être spécifié par la propriété système Java `com.ibm.wca.logging.configFile`.

- 

Ce fichier se trouve dans le répertoire `/instroot/xml`.

Le fichier `WCALoggerConfig.xml` détermine les informations de journalisation fournies par chaque utilitaire, ainsi que l'emplacement d'affichage ou de stockage de ces informations. Vous pouvez personnaliser ce fichier et spécifier les types de journaux créés, ainsi que les types de messages consignés.

---

### Configuration de la journalisation dans votre environnement pour les systèmes Windows NT, Windows 2000, AIX, Linux et Solaris

Pour configurer la journalisation dans votre environnement, vous devez soit définir la variable d'environnement système du chemin de classe (`CLASSPATH`) de telle sorte qu'elle inclue le fichier `WCALoggerConfig.xml`, soit spécifier la propriété système `com.ibm.wca.logging.configFile`.

#### Exemple de définition de la variable `CLASSPATH`

Si le fichier `WCALoggerConfig.xml` se trouve dans le répertoire `d:\WebSphere\CommerceServer\xml\loader` d'un poste Windows NT, par exemple, vous pouvez utiliser l'instruction suivante pour définir la variable `CLASSPATH` :

```
SET CLASSPATH=%CLASSPATH%;D:\WebSphere\CommerceServer\xml\loader
```

#### Exemple de spécification de la propriété système `com.ibm.wca.logging.configFile`

Pour spécifier la propriété système `com.ibm.wca.logging.configFile`, utilisez l'option `-D` lors de l'appel de l'interpréteur Java. Voici un exemple :

```
java -Dcom.ibm.wca.logging.configFile=D:\ice_tea\src\classlib\logger\xml\WC.xml  
com.ibm.wca.DTDGenerator.GeneratedDTD
```

---

## Personnalisation de la journalisation pour le module Loader

Pour personnaliser la journalisation pour le module Loader, utilisez le fichier `WCALoggerConfig.xml`.

-     

Ce fichier se trouve dans un répertoire spécifié dans la variable d'environnement système `CLASSPATH` (chemin de classe). Il peut également être spécifié par la propriété système Java `com.ibm.wca.logging.configFile`.

- 

Ce fichier se trouve dans le répertoire `/instroot/xml`.

Le fichier `WCALoggerConfig.xml` contient un ou plusieurs indicateurs de composant, tels que `<component name="DTDGenerator">`, par exemple. Dans chacun de ces indicateurs, vous pouvez ajouter des programmes de consignation et des gestionnaires. Vous ne devez pas modifier les indicateurs d'utilitaire et de programme de consignation fournis avec le système, mais vous pouvez ajouter des indicateurs de gestionnaire aux programmes de consignation. Pour plus d'informations sur ce qui peut être inclus dans ce fichier, reportez-vous au fichier `WCALogger.dtd`.

Les journaux du module Loader se trouvent dans le fichier `messages.txt`, dans les répertoires suivants :

-  `unité:\WebSphere\CommerceServer\instances\nom_instance\logs`
-  `unité:\Program Files\WebSphere\CommerceServer\instances\nom_instance\logs`
-  `/usr/WebSphere/CommerceServer/instances/nom_instance/logs`
- 
-   `/opt/WebSphere/CommerceServer/instances/nom_instance/logs`  
`/QIBM/UserData/WebCommerce/instances/nom_instance/logs`

## Gestionnaires

Pour ajouter un gestionnaire à un programme de consignation, spécifiez le type de gestionnaire dans le fichier `WCALoggerConfig.xml`. Vous pouvez ajouter plus d'un gestionnaire à un programme de consignation. Veuillez noter que chaque gestionnaire possède ses propres attributs et indicateurs secondaires, qui ne s'appliquent pas nécessairement à d'autres gestionnaires. Les différents types de gestionnaires sont les suivants :

| Type de gestionnaire  | Description et attributs  |
|---|---|
| console   | Envoie des messages à la sortie standard, généralement la ligne de commande.  |
| file  | Stocke les messages dans un fichier texte.<br><br>Vous devez ajouter "<filePath>chemin du journal</filePath>" à ce gestionnaire comme indicateur secondaire.  |
| multifile   | Crée un journal circulaire de fichiers.<br><br>Vous devez spécifier "<filePath>chemin du journal</filePath>". Les fichiers journaux 1 à <i>n</i> sont créés. Vous pouvez ajouter les attributs suivants :<br><br><b>MaxFiles</b><br>Entier indiquant combien de fichiers journaux utiliser avant d'effacer le premier fichier journal<br><br><b>MaxKBFileSize</b><br>Entier indiquant le nombre maximal de kilo-octets à stocker dans chaque fichier journal  |
| <br><br><br><br><br>database | Enregistre les messages dans une table DB2, dans un journal circulaire. Vous pouvez ajouter les attributs suivants :<br><br><b>brand</b> Nom du fabricant de la base de données. DB2 est la seule base de données actuellement prise en charge.<br><br><b>maxRows</b> Nombre maximal d'enregistrements à stocker dans une table avant l'effacement de l'entrée la plus ancienne<br><br>Vous pouvez inclure "<jdbc/>" comme indicateur secondaire et ajouter les attributs suivants :<br><br><b>url</b> URL utilisée dans JDBC pour accéder à une base de données (par exemple, "Jdbc:db2:" <i>wcm</i> ""), où " <i>wcm</i> " est le nom de la base de données). La base de données doit exister avant l'exécution de l'utilitaire.<br><br><b>table</b> Nom de la table de base de données où les messages seront consignés. Elle doit être créée à l'aide de l'instruction DB2 suivante :<br><br><pre>"CREATE TABLE"nom_de_table (KEY char(13) FOR BIT DATA NOT NULL, COMPONENTNAME VARCHAR(30), ENTRY VARCHAR(2000), PRIMARY KEY(key))"</pre><br><b>userid</b> Nom de l'utilisateur de la base de données. L'utilisateur doit disposer de droits d'accès suffisants pour mettre à jour la table. L'instruction DB2 suivante permet de les accorder :<br><br><pre>"GRANT SELECT, UPDATE, INSERT, DELETE ON TABLE"nom_de_table TO USER id_utilisateur"</pre><br><b>password</b> Mot de passe de connexion à la base de données pour le nom d'utilisateur spécifié. |

Dans l'exemple suivant, un gestionnaire de base de données (type "database") est ajouté à un programme de consignation :

```
<handler type="database" brand="DB2" maxRows="50">
  <jdbc url="jdbc:db2:wcm"
    table="wcm.log"
    userid="wasuser"
    password="123456"/>
  <filter type="Any">
    <messageType name="FATAL"/>
    <messageType name="ERROR"/>
    <messageType name="WARNING"/>
  </filter>
</handler>
```

## Filtres

Des filtres peuvent être ajoutés aux gestionnaires ou retirés de ceux-ci pour inclure ou exclure des types de messages. Si un programme de consignation n'a pas de filtre, aucun message n'est consigné. Chaque indicateur de filtre possède un indicateur secondaire messageType qui répertorie le type de message, qui est généralement l'un des suivants :

- INFO
- ERROR
- FATAL
- WARNING

D'autres types de messages sont répertoriés dans le fichier WCALogger.dtd, mais la plupart ne sont généralement pas utilisés avec le module Loader.

Les types de filtres sont les suivants :

| Type de filtre | Description et attributs  |
|----------------|---|
| Any            | Inclut dans le fichier journal tout message marqué avec l'un des types messageType spécifiés.<br><br>Par exemple, si la liste d'indicateurs messageType inclut ERROR et que l'application génère un message de type ERROR, ce message est consigné. |
| All            | Pour être inclus dans un journal, un message doit avoir tous les attributs de type messageType spécifiés.   |
| Exclude        | Consigne tous les messages non spécifiés dans la liste des indicateurs messageType.   |

Dans l'exemple suivant d'ajout d'un filtre à un gestionnaire, la journalisation des messages de type FATAL et ERROR est autorisée et les autres messages sont ignorés :

```
<handler type="file">
  <filter type="Any">
    <messageType name="FATAL"/>
    <messageType name="ERROR"/>
  </filter>
</handler>
```

## Formats

Vous pouvez spécifier l'un des deux types de programmes de formatage suivants pour les messages :

| Type de programme de formatage | Description et attributs  |
|--------------------------------|---|
| safe (valeur par défaut)       | Evite qu'une exception soit définie si un message est introuvable dans un fichier de propriétés.<br><br>Ce programme de formatage crée un message indiquant que la ressource est manquante. |
| xml                            | Convertit le message au format XML.<br><br>Si un message est introuvable, ce programme de formatage écrit également un message au lieu de définir une exception.                            |

## Exemple : WCALoggerConfig.xml et WCALogger.dtd

### WCALoggerConfig.xml

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE WCALoggerConfig SYSTEM "WCALogger.dtd">
<WCALoggerConfig>
<component name="MassLoader">
  <logger type="message">
    <handler type="file">
      <filePath>c:\temp\out.txt</filePath>
      <filter type="Any">
        <messageType name="FATAL"/>
        <messageType name="ERROR"/>
        <messageType name="WARNING"/>
        <messageType name="INFO"/>
      </filter>
    </handler>
  </logger>
  <logger type="trace">
    <handler type="file">
      <filePath>out2.txt</filePath>
      <filter type="Any">
        <messageType name="PUBLIC"/>
      </filter>
    </handler>
  </logger>
  <logger type="typedMessage">
    <handler type="file">
      <filePath>tout.txt</filePath>
      <filter type="Any">
        <messageType name="FATAL"/>
        <messageType name="ERROR"/>
        <messageType name="WARNING"/>
        <messageType name="INFO"/>
      </filter>
    </handler>
  </logger>
  <logger type="progress">
    <handler type="console" format="safe">
      <filter type="Any">
        <messageType name="FATAL"/>
        <messageType name="ERROR"/>
        <messageType name="WARNING"/>
        <messageType name="INFO"/>
      </filter>
    </handler>
  </logger>
</component>
</WCALoggerConfig>
```

```

    </filter>
  </handler>
</logger>
</component>
<component name="DTDGenerator">
  <logger type="message">
    <handler type="console">
      <filter type="Any">
        <messageType name="FATAL"/>
        <messageType name="ERROR"/>
        <messageType name="WARNING"/>
        <messageType name="INFO"/>
      </filter>
    </handler>
  </logger>
  <logger type="trace">
    <handler type="console">
      <filter type="Any">
        <messageType name="FATAL"/>
        <messageType name="ERROR"/>
        <messageType name="WARNING"/>
        <messageType name="INFO"/>
      </filter>
    </handler>
  </logger>
</component>
</WCALoggerConfig>

```

## WCALogger.dtd

```

<!-- Ce fichier DTD décrit la structure possible d'un document WCALoggerConfig XML.
Le document WCALoggerConfig XML est le fichier de configuration d'entrée pour
la classe WCALoggerFactory.
-->

<!ELEMENT WCALoggerConfig (component)+>

<!ELEMENT component (logger)+>
<!ATTLIST component name CDATA #REQUIRED>
<!ELEMENT logger (handler+,messageFile?)>
<!ATTLIST logger type (message | trace | typedMessage | progress) "typedMessage">

<!-- messageFile est un fichier de propriétés par défaut facultatif pouvant être
utilisé pour rendre les messages spécifiques de l'environnement local
-->
<!ELEMENT messageFile (#PCDATA)>
<!ELEMENT handler (filePath?, filter, jdbc?)>
<!ATTLIST handler
type ( file|multiFile|console|error|textArea|database|ejbQueue|queue ) "console">

<!-- maxFiles & maxKBFileSize s'appliquent uniquement au type de gestionnaire
multiFile
-->
<!-- filePath & encoding s'appliquent uniquement lorsque le gestionnaire est de type
file ou multiFile
-->
<!ATTLIST handler maxFiles CDATA #IMPLIED>
<!ATTLIST handler maxKBFileSize CDATA #IMPLIED>
<!ATTLIST handler encoding CDATA #IMPLIED>
<!ATTLIST handler format (safe | xml) "safe">
<!-- maxRecords & brand s'appliquent uniquement au type de gestionnaire database
-->
<!ATTLIST handler maxRecords CDATA #IMPLIED>
<!ATTLIST handler brand (DB2) #IMPLIED>
<!-- l'indicateur jdbc doit être présent dans un indicateur de type de gestionnaire
database
-->

```

```

<!ELEMENT jdbc EMPTY>
<!ATTLIST jdbc url CDATA #IMPLIED>
<!ATTLIST jdbc table CDATA #IMPLIED>
<!ATTLIST jdbc userid CDATA #IMPLIED>
<!ATTLIST jdbc password CDATA #IMPLIED>

<!ELEMENT filter (messageType+)>
<!ATTLIST filter type (Any | All | Exclude ) "Any">

<!-- le nom d'attribut messageType est l'une des constantes JLog IRecordType
-->
<!ELEMENT messageType EMPTY>
<!ATTLIST messageType name ( NONE | ALL | INFO |
INFORMATION | WARN | WARNING | ERR | ERROR |
FATAL | DEFAULT_MESSAGE | API | CALLBACK |
ENTRY_EXIT | ENTRY | EXIT | ERROR_EXC |
MISC_DATA | OBJ_CREATE | OBJ_DELETE |
PRIVATE | PUBLIC | STATIC | SVC | PERF |
LEVEL1 | LEVEL2 | LEVEL3 ) "ALL">
<!ELEMENT filePath (#PCDATA)>

```



---

## Chapitre 9. Utilisation de la fonction de signalisation des erreurs du module Loader

Les utilitaires Loader et ID Resolver comprennent une fonction de signalisation des erreurs, qui génère un document d'exception en cas d'erreur.

Par défaut, ce document d'exception est créé dans le répertoire suivant :

-      répertoire où réside le document d'entrée
-  `/instroot/logs`

Pour spécifier le répertoire dans lequel le document d'exception doit être créé, utilisez la propriété Java `com.ibm.wcm.ErrorReporterDir`. Voici le début d'un exemple d'instruction pour l'utilitaire Loader dans un environnement Windows NT :

```
java -Dcom.ibm.wcm.ErrorReporterDir=d:\massloaderrors  
com.ibm.wca.MassLoader.MassLoad -dbname . . .
```

**Remarque :** L'utilisateur doit disposer des droits d'accès en écriture sur le répertoire indiqué.

Voici un exemple de fichier DTD (`store-all-error.dtd`) pour la fonction de signalisation des erreurs :

```
<!ENTITY % TABLE "calrule | catentry">  
<!ELEMENT store-asset (error, (%TABLE;)*)>  
<!ELEMENT message (#PCDATA) >  
<!ELEMENT error ( message ) >  
<!ATTLIST error  
  locus          CDATA      #REQUIRED  
  id             CDATA      #REQUIRED  
>  
<!ELEMENT calrule (error)>  
<!ATTLIST calrule  
  identifieur    CDATA      #REQUIRED  
  calrule_id     CDATA      #REQUIRED  
  calcode_id     CDATA      #REQUIRED  
  startdate      CDATA      #IMPLIED  
  taxcgry_id     CDATA      #IMPLIED  
  enddate        CDATA      #IMPLIED  
  sequence       CDATA      #REQUIRED  
  combination    CDATA      #REQUIRED  
  calmethod_id   CDATA      #REQUIRED  
  calmethod_id_qfy CDATA      #REQUIRED  
  flags          CDATA      #REQUIRED  
  field1         CDATA      #IMPLIED  
  field2         CDATA      #IMPLIED  
>  
<!ELEMENT catentry (error)>  
<!ATTLIST catentry  
  catentry_id    CDATA      #REQUIRED  
  member_id     CDATA      #REQUIRED  
  catenttype_id  CDATA      #REQUIRED  
  partnumber     CDATA      #IMPLIED  
  mfpnumber      CDATA      #IMPLIED  
  mfname         CDATA      #IMPLIED  
  markfordelete CDATA      #REQUIRED
```

```

url          CDATA    #IMPLIED
field1       CDATA    #IMPLIED
field2       CDATA    #IMPLIED
lastupdate   CDATA    #IMPLIED
field3       CDATA    #IMPLIED
onspecial    CDATA    #IMPLIED
onauction    CDATA    #IMPLIED
field4       CDATA    #IMPLIED
field5       CDATA    #IMPLIED
buyable      CDATA    #IMPLIED
>

```

Voici un exemple de document de signalisation des erreurs généré par Loader :

```

<?xml version="1.0"?>
<!DOCTYPE store-asset SYSTEM "store-all-error.dtd">
<store-asset>
  <error
    locus="Parser"
    id="SAXParseFatalError" >
    <message>
      Error The string "--" is not permitted within comments. : 155 : 18
    </message>
  </error>
  <calrule
    calcode_id="30"
    enddate="2100-01 10:20:30.000000"
    calmethod_id="-47"
    identifier="7"
    taxcgry_id="9"
    calmethod_id_qfy="-46"
    startdate="1900-01-01-00.00.00.000000"
    flags="1"
    combination="2"
    calrule_id="44"
    sequence="9.0E+1">
    <error
      locus="Writer"
      id="SQLException" >
      <message>
        A SQL Exception was received [IBM][CLI Driver][DB2/NT] SQL0530N
        The insert or update value of the FOREIGN KEY
        "JANTONY.CALRULE.F_CALRULE4" is not equal to any value of the
        parent key of the parent table. SQLSTATE=23503
      </message>
    </error>
  </calrule>
  <catentry
    catentry_id="10118"
    member_id="-2001"
    partnumber="1254"
    mfpartnumber="sku-163"
    mfname="InFashion"
    markfordelete="0"
    buyable="1"
    field1="abc" >
    <error
      locus="Formatter"
      id="FormattingError" >
      <message>
        Erreur lors du formatage de la valeur pour CATENTRY.FIELD1 :
        abc avec l'erreur
        [class java.lang.NumberFormatException(abc)].
      </message>
    </error>
  </catentry>
</store-asset>

```

---

## Chapitre 10. Configuration des instructions et scripts du module Loader

Pour lancer le module Loader et exécuter ses instructions, utilisez les scripts ou les instructions fournis dans le répertoire WebSphere Commerce :

-  *unité*:\WebSphere\CommerceServer\bin
-  *unité*:\Program Files\WebSphere\CommerceServer\bin
-  /usr/WebSphere/CommerceServer/bin
-   /opt/WebSphere/CommerceServer/bin
-  Bibliothèque native QWEBCOMM

Les scripts et les instructions fournis sont les suivants :



### GENWCSDTD

Instruction DTD Generate

### RESWCSID

Instruction ID Resolve

### EXTWCSDTA

Instruction Extract

### LODWCSDTA

Instruction Load

### TRNWCSTXT

Instruction Text Transform

### TRNWCXML

Instruction XML Transform



### **dtdgen.cmd**

Instruction DTD Generate

### **idresgen.cmd**

Instruction ID Resolve

### **massextract.cmd**

Instruction Extract

### **massload.cmd**

Instruction Load

### **txttransform.cmd**

Instruction Text Transform

### **xmltransform.cmd**

Instruction XML Transform



**dtdgen.sh**  
Script shell DTD Generate

**idresgen.sh**  
Script shell ID Resolve

**massextract.sh**  
Script shell Extract

**massload.sh**  
Script shell Load

**txttransform.sh**  
Script shell Text Transform

**xmltransform.sh**  
Script shell XML Transform

---

## Partie 3. Utilisation de Web editor

La présente section décrit les procédures d'administration et d'utilisation de Catalog Manager Web editor.

Web editor vous permet de créer ou supprimer des données de catalogue, ou encore d'apporter des modifications à ces dernières, via un navigateur Web. Les formulaires d'entrée de données utilisés pour l'affichage et la mise à jour des informations sont centralisés dans Web editor. Dans le cas le plus simple, les formulaires correspondent aux tables de la base de données WebSphere Commerce. L'administrateur peut choisir d'utiliser les formulaires par défaut fournis ou de personnaliser les formulaires disponibles.

**Remarque :** Web editor utilise Internet Explorer 5 et versions suivantes.



---

## Chapitre 11. Configuration de Web editor

L'administrateur peut configurer Web editor pour fournir des données à WebSphere Commerce, ainsi que des extensions et des personnalisations au schéma WebSphere Commerce. Web editor n'est pas conçu comme un ensemble de formulaires d'entrée de données sur le Web spécifiques d'une instance particulière de WebSphere Commerce. Il est conçu comme un outil flexible et personnalisable afin de pouvoir prendre en charge les besoins et profils individuels de chaque entreprise.

Une vue de base de données est une requête enregistrée sur une ou plusieurs tables de la base de données. Lors de l'installation de Catalog Manager, un modèle de fichier permettant la création d'une vue logique d'un produit WebSphere Commerce est placé dans chaque sous-répertoire spécifique du système (db2, oracle etos400) du répertoire suivant :

-  *unité:*\WebSphere\CommerceServer\schema
-  *unité:*\Program Files\WebSphere\CommerceServer\schema
-  /usr/WebSphere/CommerceServer/schema
-   /opt/WebSphere/CommerceServer/schema
-  /QIBM/ProdData/WebCommerce/schema

Ce modèle de fichier, *wcs.view.sql*, contient une vue *Produit* qui a été créée pour regrouper les informations concernant les produits de plusieurs tables. Il contient la définition de données SQL de cette vue *Produit*. Les administrateurs peuvent étudier ce fichier pour planifier le développement de leurs propres vues de base de données.

Le fichier de description de formulaire XML par défaut (*forms51\_be.xml*) contient des formulaires conçus pour faciliter l'ajout, la modification et la suppression de données dans une base de données WebSphere Commerce. Une copie de ce fichier se trouve dans le répertoire suivant :

-  *unité:*\WebSphere\CommerceServer\xml\wcwebeditor\xml
-  *unité:*\Program Files\WebSphere\CommerceServer\xml\wcwebeditor\xml
-  /usr/WebSphere/CommerceServer/xml/wcwebeditor/xml
-   /opt/WebSphere/CommerceServer/xml/wcwebeditor/xml
-  /instroot/xml/wcwebeditor/xml

Ce fichier de configuration peut être utilisé en l'état ou il peut être modifié et amélioré par l'administrateur qui configure Web editor. Pour personnaliser le fichier de description de formulaire XML, reportez-vous aux instructions figurant plus loin dans cette section.

**Remarque :** L'utilitaire DTD Generator peut créer automatiquement les formulaires devant être utilisés par Web editor.

---

## Configuration de Web editor

Cette section fournit des informations sur la procédure de configuration de Web editor. Bien que le processus d'installation traite initialement cette configuration, un administrateur peut utiliser ces informations pour reconfigurer Web editor afin d'utiliser une autre base de données, par exemple.

### Modification du fichier `webeditor.properties`

Web editor utilise certains paramètres d'application qui sont définis dans le fichier `webeditor.properties`. Ce fichier se trouve dans le répertoire suivant :

-  `unité:\WebSphere\AppServer\installedApps\WC_Enterprise_App_demo.ear\wcwebeditor.war\WEB-INF\classes\webeditor.properties`
-  `unité:\Program Files\WebSphere\AppServer\installedApps\WC_Enterprise_App_demo.ear\wcwebeditor.war\WEB-INF\classes\webeditor.properties`
-  `/usr/WebSphere/AppServer/installedApps/WC_Enterprise_App_demo.ear/wcwebeditor.war/WEB-INF/classes/webeditor.properties`
-   `/opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_demo.ear/wcwebeditor.war/WEB-INF/classes/webeditor.properties`
-  `/QIBM/UserData/WEBASADV4/nom_instance_was/installedApps/WC_Enterprise_App_nom_instance_wcs.ear/wcwebeditor.war/WEN-INF/classes`

En modifiant le fichier `webeditor.properties`, un administrateur peut par exemple changer le fichier utilisé pour décrire les formulaires qui sont affichés dans Web editor. Voici un exemple de contenu du fichier `webeditor.properties` :

```
# Fichier de propriétés pour WebEditor

(Le paramètre suivant indique où se trouvent la liste de processus personnalisée et les
enveloppes de configuration de l'utilitaire Catalog Manager.)
# Emplacement URI des processus et des enveloppes de configuration du sous-système WCM
ProcessConfigFile=file:///D:/WebSphere/CommerceServer/xml/wcwebeditor/xml/weProcessList.xml

(Le paramètre suivant indique où les formulaires sont définis.)
# Emplacement du fichier de formulaires
FormsURL=file:///D:/WebSphere/CommerceServer/xml/wcwebeditor/xml/forms51_be.xml

(Le paramètre suivant indique où se trouve la feuille de style de conversion XML en HTML)
# Emplacement de la feuille de style de conversion XML en HTML
StyleSheetURI=file:///D:/WebSphere/CommerceServer/xml/wcwebeditor/xsl/webeditor.xsl

(Le paramètre suivant indique l'emplacement des fichiers temporaires.)
# Emplacement du répertoire temporaire
#temp.dir=

# source de données WebSphere. Cette valeur est utilisée pour créer des listes
# déroulantes de formulaires et des valeurs de zone par défaut.
# Cependant, lors de la publication et des recherches, Web Editor utilise les
# sous-systèmes WCM ID Resolver, Mass Loader et Mass Extractor. L'accès aux
# bases de données pour ces sous-systèmes WCM doit être configuré séparément.
# Nom de source de la base de données WAS
dbsource=jdbc/WebSphere Commerce DB2 DataSource demo

# Nom de la base de données WAS. Si elle est spécifiée, cette valeur
# est préférée au nom de base de données dans le fichier XML
# de formulaires. Cette valeur est utilisée comme paramètre lors de
```

```
# l'appel de sous-systèmes WCM, tels que Mass Loader.
dbname=all
```

(Le paramètre suivant indique le jeu de caractères à utiliser. Par défaut, il s'agit du jeu de caractères mono-octet. Pour les autres pays, sélectionnez une des valeurs mises en commentaire.)

```
# Jeu de caractères de codage
Encoding=ISO-8859-1
```

```
# CN
#Encoding=gb2312
# TW
#Encoding=Big5
# KR
#Encoding=EUC-KR
# JP
#Encoding=Shift_JIS
```

(Le paramètre suivant permet d'autoriser la prévisualisation des images. Vous devez indiquer ici le nom d'hôte du serveur plus toute information de répertoire avant les informations sur l'image stockées dans WebSphere Commerce. Les informations du catalogue WebSphere Commerce sont ajoutées à cette valeur pour construire l'adresse URL d'une image.)

```
# Spécifie l'emplacement href des images dans la base
# Cette valeur doit être définie de façon que ces infos et celles stockées dans WCS
# (par exemple, /image/char.gif) se combinent pour créer l'adresse URL d'une image
```

```
#imageRootURL=http://%HOSTDOMAIN
imageRootURL=http://localhost/webeditor
```

(Le paramètre suivant permet de définir un format de date pour l'application. Si ces propriétés ne sont pas disponibles (c'est-à-dire, si elles sont mises en commentaire), c'est le format

spécifique de l'environnement local Java qui est utilisé. Ci-dessous figure un tableau de référence pour la définition de ces valeurs.)

```
# Utilisez ces propriétés pour spécifier un format de date si vous ne souhaitez pas
# utiliser le format spécifique de l'environnement local Java.
```

```
#dateFormat=dd/MM/yyyy
#dateTimeFormat=dd/MM/yyyy HH:mm:ss
```

| # | Symbole | Signification        | Présentation     | Exemple                |
|---|---------|----------------------|------------------|------------------------|
| # | -----   |                      | -----            | -----                  |
| # | G       | indicateur d'ère     | (Texte)          | Après J-C              |
| # | y       | année                | (Nombre)         | 1996                   |
| # | M       | mois de l'année      | (Texte & nombre) | Juillet & 07           |
| # | d       | jour du mois         | (Nombre)         | 10                     |
| # | h       | heure (format am/pm) | (1~12) (Nombre)  | 12                     |
| # | H       | heure (format 24h)   | (0~23) (Nombre)  | 0                      |
| # | m       | minutes              | (Nombre)         | 30                     |
| # | s       | secondes             | (Nombre)         | 55                     |
| # | S       | millisecondes        | (Nombre)         | 978                    |
| # | E       | jour de la semaine   | (Texte)          | Mardi                  |
| # | D       | jour de l'année      | (Nombre)         | 189                    |
| # | F       | jour du mois         | (Nombre)         | 2 (Mercredi 2 juillet) |
| # | w       | semaine de l'année   | (Nombre)         | 27                     |
| # | W       | semaine du mois      | (Nombre)         | 2                      |
| # | a       | Marqueur am/pm       | (Texte)          | PM                     |
| # | k       | heure (format 24h)   | (1~24) (Nombre)  | 24                     |
| # | K       | heure (format am/pm) | (0~11) (Nombre)  | 0                      |
| # | z       | fuseau horaire       | (Texte)          | Heure du Pacifique     |
| # | '       | échappement          | (Délimiteur)     | '                      |
| # | ''      | guillemet simple     | (Littéral)       | '                      |

## Changement d'emplacement des fichiers temporaires

L'emplacement des fichiers temporaires peut être changé en supprimant la marque de mise en commentaire et en ajoutant une valeur à la propriété temp.dir dans le fichier webeditor.properties.

Il est également possible d'utiliser la propriété Java java.io.tmpdir pour déterminer où les fichiers temporaires sont créés.

---

## Création d'un fichier de description de formulaire XML à l'aide de DTD Generator

Le fichier forms51\_be.xml est un exemple de fichier de description de formulaire XML. Il fournit un ensemble de formulaires devant être utilisés par Web editor. Une copie de ce fichier se trouve dans le répertoire suivant :

-  unité:\WebSphere\CommerceServer\xml\wcwebeditor\xml
-  unité:\Program Files\WebSphere\CommerceServer\xml\wcwebeditor\xml
-  /usr/WebSphere/CommerceServer/xml/wcwebeditor/xml
-   /opt/WebSphere/CommerceServer/xml/wcwebeditor/xml
-  /instroot/xml/wcwebeditor/xml

Les étapes suivantes décrivent comment l'administrateur système peut utiliser l'utilitaire DTD Generator pour ajouter de nouveaux formulaires XML.

**Remarque :** Avant d'exécuter la procédure décrite ci-après, renommez le fichier forms51\_be.xml existant dans le répertoire Web editor. Si vous préférez, vous pouvez également créer au cours de la procédure suivante un fichier de sortie portant un nouveau nom, puis reconfigurer Web editor pour qu'il utilise le fichier venant d'être créé. Pour ce faire, reportez-vous aux instructions de la section précédente.

Pour créer des formulaires XML, exécutez l'instruction DTD Generate.

Les étapes suivantes décrivent la procédure de création des formulaires XML :

1. Créez un fichier temporaire nommé "tables.txt", contenant les noms des tables que vous voulez utiliser dans les formulaires.  
Entrez chaque nom de table sur une seule ligne, comme dans l'exemple suivant :
- ```
catentry
catentdesc
catentship
inventory
```
2. Sauvegardez le fichier tables.txt dans le répertoire où se trouve l'instruction DTD Generate. (Reportez-vous au Chapitre 10, «Configuration des instructions et scripts du module Loader» à la page 73 pour connaître le répertoire d'installation de cette instruction.)
  3. A l'invite du système d'exploitation, passez dans le répertoire où se trouve l'instruction DTD Generate.
  4. Exécutez l'instruction DTD Generate en entrant ce qui suit :

▶ NT ▶ 2000

```
dtdgen -infile tables.txt -outfile tables51.dtd  
-dbname nom_bd -dbuser id_utilisateur -dbpwd mot_de_passe  
-xmlTableDesc tableFORMS.xml -schemaname schéma -propfile nom_fichier
```

▶ AIX ▶ Solaris ▶ Linux

```
./dtdgen.sh -infile tables.txt -outfile tables51.dtd  
-dbname nom_bd -dbuser id_utilisateur -dbpwd mot_de_passe  
-xmlTableDesc tableFORMS.xml -schemaname schéma -propfile nom_fichier
```

▶ 400

```
QWBECCOMM/GENWCSDTD DATABASE(base_de_données) SCHEMA(schéma)  
INSTROOT(instroot) PASSWD(mot_de_passe) OUTFILE(tables51.dtd)  
INFILE(tables.txt) XMLTABDESC(tableFORMS.xml)
```

L'indicateur de description de table (-xmlTableDesc ou XMLTABDESC) oblige DTD Generator à créer une nouvelle description des formulaires des tables en plus d'un fichier DTD.

▶ NT ▶ 2000 ▶ AIX ▶ Solaris ▶ Linux

L'option -propFile indique le nom d'un fichier de propriétés externe dans lequel peuvent être stockés le texte d'aide, les valeurs par défaut et les informations de description de zone.

5. Reconfigurez Web editor pour utiliser les fichiers venant d'être créés, en suivant les instructions de la section précédente.
6. Relancez Web editor à partir de la console d'administration WebSphere Advanced Administrative Console. Pour ce faire, procédez comme suit :
  - a. Développez le menu **Domaine administratif WebSphere**.
  - b. Développez le noeud **Application entreprise**.
  - c. Cliquez avec le bouton droit de la souris sur **Application entreprise WebSphere Commerce - démo** et sélectionnez **Arrêter**.
  - d. Attendez jusqu'à ce qu'un message indique que l'application a été arrêtée.
  - e. Cliquez avec le bouton droit de la souris sur **Application entreprise WebSphere Commerce - démo** et sélectionnez **Démarrer**.
  - f. Attendez jusqu'à ce qu'un message indique que l'application a été démarrée.
7. Pour voir le nouveau formulaire dans un navigateur Web, ouvrez l'adresse URL suivante :

`https://nom_hôte:8000/wcm/webeditor`

où *nom\_hôte* est le nom d'hôte HTTP qualifié complet du serveur WebSphere Application Server.

Web editor s'affiche dans la fenêtre du navigateur et présente la liste de tous les noms de table.

## Personnalisation de la description de formulaire XML

Cette section explique comment l'administrateur peut améliorer les formulaires affichés par Web editor.

La description de formulaire XML peut être personnalisée et améliorée en définissant des attributs et des valeurs dans le fichier de description de formulaire XML proprement dit ou dans un fichier de propriétés distinct.

**Remarque :** Le nom de ce fichier de propriétés doit être indiqué comme valeur de l'attribut `resourcePackage` dans l'indicateur `formList`. Si le nom du fichier apparaît dans un sous-répertoire d'un répertoire du chemin de classe, il doit utiliser la spécification des paquets (point).

Le tableau suivant répertorie les attributs de zone de formulaire de Web editor pouvant être modifiés par un administrateur.

| Attributs de zone       | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Currency                | Permet d'afficher les valeurs avec un séparateur numérique spécifique de l'environnement local (tel qu'une virgule comme séparateur des milliers aux États-Unis).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| DbColumn                | Permet de mapper le nom de zone sur la clé de fichier de propriétés.<br><br>Si vous utilisez des fichiers de propriétés spécifiques de l'environnement local, la valeur entrée ici doit correspondre à celle entrée dans le fichier de propriétés. DTD Generator ajoute un schéma à la fin de cette entrée.                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| DefaultValue            | Indique une valeur qui est affichée dans le formulaire de saisie de données lorsqu'un utilisateur remplit un nouveau formulaire.<br><br>La valeur par défaut de la base de données peut être affectée à cet attribut par DTD Generator. Il peut s'agir d'une chaîne statique, mais cette valeur peut également inclure une requête scalaire SQL s'appliquant à une table comportant une seule ligne. Pour extraire des données externes, une fonction définie par l'utilisateur peut être utilisée dans la requête. Par exemple :<br><br><code>DefaultValue="SELECT CURRENT TIMETAMP FROM EXEC"</code><br><br>où EXEC est un élément défini et alimenté comme suit :<br><br><code>CREATE TABLE EXEC (A CHAR(1));<br/>INSERT INTO A VALUES('A');</code> |
| dynamicSqlSelectionList | Permet de reconstituer un menu déroulant pour chaque formulaire.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |



| Attributs de zone                                               | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-----------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>readOnly</p> <p>readOnlyForCreate</p> <p>readOnlyForEdit</p> | <p>Ces attributs contrôlent les moments où une zone est disponible pour modification et ceux où elle ne peut être consultée par l'utilisateur qu'en lecture seule.</p> <p>readOnly="true" signifie qu'une zone est toujours en mode lecture seule. readOnlyForCreate="true" signifie qu'une zone est en mode lecture seule lors de la création d'un nouveau formulaire.</p> <p>readOnlyForEdit="true" signifie qu'une zone est en mode lecture seule lors de la modification d'un formulaire existant.</p>                                                                                                        |
| <p>ShowColumnInList</p><br><p>showInCreateMode</p>              | <p>Indique que la zone correspond à l'une des colonnes constituant la vue de plusieurs enregistrements de données, lorsque la valeur "true" est définie.</p> <p>DTD Generator affecte la valeur "true" aux six premières colonnes.</p> <p>Permet de masquer une zone dans le formulaire créé.</p> <p>Affectez la valeur "false" à l'attribut showInCreateMode pour masquer la zone dans le formulaire créé affiché. La valeur par défaut de cet attribut est "true".</p>                                                                                                                                          |
| <p>ShowInSearchMode</p>                                         | <p>Permet de masquer certaines colonnes dans la page des critères de recherche.</p> <p>Si la valeur "false" est affectée à cet attribut, la zone concernée n'est pas affichée dans le formulaire de recherche.</p>                                                                                                                                                                                                                                                                                                                                                                                                |
| <p>SqlSelectionList</p>                                         | <p>Crée un menu déroulant dans lequel un utilisateur peut faire une sélection lorsqu'il crée un nouveau formulaire.</p> <p>La requête doit renvoyer un ensemble de résultats dans une ou deux colonnes. La première colonne correspond à la liste des intitulés dans laquelle l'utilisateur peut faire son choix et la seconde colonne correspond à la liste des valeurs réelles à stocker dans la table de la base de données. Si une seule colonne est utilisée, la valeur affichée est celle qui est stockée dans la table. Cette fonction est très utile pour établir les relations entre clés associées.</p> |
| <p>Type</p>                                                     | <p>Indique le type de colonne de base de données et le type de contrôle de validation devant être effectué sur les données.</p> <p>Par exemple, si le type est "integer" (entier), l'application s'assure que seules des valeurs entières valides sont entrées dans cette zone.</p>                                                                                                                                                                                                                                                                                                                               |

| Attributs de zone | Description                                                                                                                                                                                                                                                                                                                                                          |
|-------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UniqueKey         | Indique que la zone est la clé primaire d'une table.<br><br>Si la valeur "true" est affectée à cet attribut, l'application établit une contrainte d'unicité sur les données saisies pour cette colonne. Ce contrôle n'est effectué que sur les enregistrements chargés dans l'application. Il ne s'applique pas aux enregistrements existants de la base de données. |
| ValidateInput     | Permet de désactiver le contrôle de validation.<br><br>Cet attribut permet à l'utilisateur de saisir du texte dans une zone numérique qui sera traitée par les utilitaires ID Resolver ou XML Transformer.                                                                                                                                                           |

## Modification des noms de formulaire

L'indicateur form du fichier de description de formulaire XML a les attributs name et displayName.

- L'attribut name doit avoir pour valeur le nom de la table ou de la vue de la base de données qu'il représente.
- L'attribut displayName fournit un nom simple à utiliser dans l'application Web editor. L'attribut displayName associé à un formulaire peut également être défini dans le fichier de propriétés spécifié dans l'attribut resourcePackage de l'indicateur formList. (L'attribut resourcePackage est utilisé pour la prise en charge multilingue.)

**Remarque :** Si une entrée figure dans le fichier de propriétés, elle a la priorité sur celle figurant dans le fichier de description de formulaire XML.

## Modification d'une description de zone

Voici un exemple de description de zone figurant dans un fichier de description de formulaire XML :

```
<form name = "CATALOG.CAENTRY"
.
.
<field name="MEMBER_ID"
  showInCreateMode="false"
  fieldDescription="MEMBER_ID"
  type="integer"
  maxLength="19"
  defaultValue=""
.
.
<field name="CATENTTYPE_ID"
  fieldDescription="CATENTTYPE_ID"
  type="string"
  maxLength="16"
  defaultValue=""
.
.
```

Vous pouvez, si vous le souhaitez, la remplacer par la description de zone suivante :

```
<form name = "CATALOG.CATENTRY"  
  displayName="Produit"  
  .  
  .  
  <field name="MEMBER_ID"  
    showInCreateMode="false"  
    fieldDescription="Identificateur de membre"  
    type="integer"  
    maxlength="19"  
    defaultValue=""  
    .  
    .  
  <field name="CATENTTYPE_ID"  
    fieldDescription="Type de produit"  
    type="string"  
    maxlength="16"  
    defaultValue=""  
    .  
    .
```

## Ajout d'un menu déroulant

Vous pouvez ajouter une liste de sélection, sous la forme d'un menu déroulant, à un formulaire Web editor afin de simplifier le remplissage de ce dernier et de limiter la sélection de l'utilisateur à un ensemble de choix valides.

Pour régénérer les listes de sélection, l'administrateur peut relancer Web editor. L'administrateur peut également affecter la valeur "true" à l'attribut de zone de description de formulaire `dynamicSqlSelectionList`, afin de s'assurer qu'un menu déroulant est rechargé à chaque fois.

La requête associée à la liste de sélection peut renvoyer un ensemble de résultats dans une ou deux colonnes. Si le résultat renvoie deux colonnes, la seconde colonne contient les valeurs réelles stockées, tandis que la première colonne contient les intitulés désignés par l'utilisateur.

Une liste de sélection est créée en entrant une requête SQL dans l'attribut `sqlSelectionList` de l'indicateur `field` ou en créant une énumération dans le fichier de description de formulaire XML. Les deux méthodes sont présentées dans l'exemple suivant :

```
<field name="MEMBER_ID"  
  showInCreateMode="false"  
  fieldDescription="Identificateur de membre"  
  .  
  .  
  readOnly="false"  
  sqlSelectionList="select orgentityname,orgentity.orgentity_id  
    from member,orgentity where member.type='0'  
    and member.member_id=orgentity.orgentity_id"  
  .  
  .  
  fieldHelp=""  
  .  
  .  
</field>  
  <field name="MARKFORDELETE"  
    fieldDescription="Marque de suppression"  
    type="NMOKEN"  
    sqlSelectionList=""
```

```

>
<datatype source="integer">
  <enumeration label="No" value="0"/>
  <enumeration label="Yes" value="1"/>
</datatype>
</field>
>

```

Certaines zones des tables de base de données WebSphere Commerce utilisent les chiffres 1 et 0 pour désigner les valeurs booléennes true et false. Pour créer des zones plus intuitives pour l'utilisateur, il est possible d'établir des énumérations de ces zones. Catalog Manager fournit un script SQL pour créer une table d'énumérations nommée "NUMDESC." Ce script est nommé "createEnum.sql." Un administrateur peut modifier l'attribut sqlSelectionList pour créer une liste de sélection en utilisant la table ENUMDESC, comme indiqué dans l'exemple ci-après :

```

fieldDescription="Offre spéciale"
.
.
readOnly="false"
sqlSelectionList="select description,value from
  enumdesc where columnname='ALL' and type='YESNO'"
fieldHelp=""
.
.

```

## Ajout d'aide sur les zones

L'aide sur la zone mise en évidence s'affiche en bas de la fenêtre du navigateur Web. Il est possible d'utiliser l'attribut fieldHelp de l'indicateur field du fichier de description de formulaire XML ou la clé fieldHelp du fichier de propriétés pour définir cette valeur.

Par exemple, le fichier de propriétés peut contenir les spécifications d'aide sur les zones suivantes :

```

CATEGORY.MARKFORDELETE.defaultValue=No
CATEGORY.MARKFORDELETE.fieldDescription=Supprimer l'entrée

```

Si aucune valeur n'est spécifiée, un message par défaut du type "Entrez ici une valeur pour *nom\_de\_zone type\_de\_zone*" est créé.

**Remarque :** S'il existe un fichier de propriétés contenant des informations relatives au formulaire, il a priorité sur les entrées du fichier de description de formulaire XML.

## Personnalisation des résultats de la recherche et de la liste de travail

L'ensemble de colonnes qui s'affiche sur la page des résultats de la recherche peut être personnalisé. Le fichier de description de formulaire XML contient un attribut showColumnInList pour l'indicateur field. Pour inclure une zone dans les résultats de la recherche, affectez la valeur "true" à cet attribut. Dans le cas contraire, cette zone ne fera pas partie des colonnes affichées dans la vue des résultats de la recherche. La zone est affichée en même temps que le formulaire.

## Modification du fichier weProcessList

Le fichier weProcessList permet à un administrateur de personnaliser les utilitaires de Catalog Manager qui sont exécutés lorsqu'une liste de travail Web editor est traitée.

- ▶ DB2 ▶ NT ▶ 2000 ▶ AIX ▶ Solaris ▶ Linux Le fichier weProcessList.xml se trouve dans le répertoire suivant :
  - ▶ NT unité:\WebSphere\CommerceServer\xml\wcwebeditor\xml
  - ▶ 2000 unité:\Program Files\WebSphere\CommerceServer\xml\wcwebeditor\xml
  - ▶ AIX /usr/WebSphere/CommerceServer/xml/wcwebeditor/xml
  - ▶ Solaris ▶ Linux /opt/WebSphere/CommerceServer/xml/wcwebeditor/xml
- ▶ Oracle ▶ NT ▶ 2000 ▶ AIX ▶ Solaris ▶ Linux Le fichier weProcessListOracle.xml se trouve dans le répertoire suivant :
  - ▶ NT unité:\WebSphere\CommerceServer\xml\wcwebeditor\xml
  - ▶ 2000 unité:\Program Files\WebSphere\CommerceServer\xml\wcwebeditor\xml
  - ▶ AIX /usr/WebSphere/CommerceServer/xml/wcwebeditor/xml
  - ▶ Solaris ▶ Linux /opt/WebSphere/CommerceServer/xml/wcwebeditor/xml
- ▶ 400 Le fichier weProcessListAS400.xml se trouve dans le répertoire suivant :
  - /instroot/xml/wcwebeditor/xml

Ce fichier contient des modèles d'enveloppe pour les divers utilitaires. Il peut également contenir des références aux applications personnalisées que l'administrateur souhaite exécuter.

Un certain nombre de variables système peuvent être utilisées dans ce fichier. La variable système %-dbname%, par exemple, provoque l'insertion du nom de la base de données dans l'enveloppe qui est générée pour un appel donné d'un utilitaire tel que Loader. Le fichier de description de formulaire XML contient des références à ces processus, qui indiquent quels sont ceux qui doivent être appelés pour un ajout, une modification ou une suppression.

Voici un exemple de fichier weProcessList.xml :

```
<?xml version="1.0" encoding="UTF-8"?>
<processSet>
  <!-- Ne modifiez pas le nom extract -->
  <process name="extract"
    subsystem="com.ibm.wca.MassExtract.extract.ExtractSubSystem">
    <envelope-input xmlns='saf_params.xsd'>
      <param name="-dbname" type="scalar" value="%-dbname%"/>
      <param name="-dbuser" type="scalar" value="%-dbuser%"/>
      <param name="-dbpwd" type="scalar" value="%-dbpwd%"/>
      <param name="-outfile" type="file" reside="local" value="%tempFilePath%"/>
      <param name="-filter" type="file" reside="local" value="%tempFileURI%"/>
    </envelope-input>
  </process>
  <process name="transformer"
    subsystem="com.ibm.wca.XMLTransformer.XMLTransformerSubSystem">
    <envelope-input xmlns='saf_params.xsd'>
      <param name="-infile" type="file" reside="local" value="%tempFileURI%"/>
    </envelope-input>
  </process>
</processSet>
```

```

        <param name="-transform" type="file" reside="local"
            value="%webEditorDir%/xsl/ViewsToWCS51.XSL"/>
        <param name="-outfile" type="file" reside="local" value="%tempFilePath1%"/>
        <param name="-param" value="root=%-dbname%"/>
        <param name="-param" value="dtdname=%-dtdname%"/>
    </envelope-input>
</process>
<process name="transformerForDelete"
    subsystem="com.ibm.wca.XMLTransformer.XMLTransformerSubSystem">
    <envelope-input xmlns='saf_params.xsd'>
        <param name="-infile" type="file" reside="local" value="%tempFileURI%"/>
        <param name="-transform" type="file" reside="local"
            value="%webEditorDir%/xsl/ViewsToWCS51.XSL"/>
        <param name="-outfile" type="file" reside="local" value="%tempFilePath1%"/>
        <param name="-param" value="root=%-dbname%"/>
        <param name="-param" value="dtdname=%-dtdname%"/>
        <param name="-param" value="forDelete=true"/>
    </envelope-input>
</process>
<process name="resolver"
    subsystem="com.ibm.wca.IdResGen.IdResGenSubSystem">
    <envelope-input xmlns='saf_params.xsd'>
        <param name="-dbname" type="scalar" value="%-dbname%"/>
        <param name="-dbuser" type="scalar" value="%-dbuser%"/>
        <param name="-dbpwd" type="scalar" value="%-dbpwd%"/>
        <param name="-infile" type="file" reside="local"
            value="%previousOutFileAsURI%"/>
        <param name="-outfile" type="file" reside="local"
            value="%tempFilePath2%"/>
        <param name="-propfile" type="file" reside="local"
            value="propertyFiles.IdKeys"/>
        <param name="-method" type="scalar" value="mixed"/>
    </envelope-input>
</process>
<!-- Resolver est défini comme le premier processus -->
<process name="resolverFirstProcess"
    subsystem="com.ibm.wca.IdResGen.IdResGenSubSystem">
    <envelope-input xmlns='saf_params.xsd'>
        <param name="-dbname" type="scalar" value="%-dbname%"/>
        <param name="-dbuser" type="scalar" value="%-dbuser%"/>
        <param name="-dbpwd" type="scalar" value="%-dbpwd%"/>
        <param name="-infile" type="file" reside="local" value="%tempFileURI%"/>
        <param name="-outfile" type="file" reside="local" value="%tempFilePath2%"/>
        <param name="-propfile" type="file" reside="local" value="propertyFiles.IdKeys"/>
        <param name="-method" type="scalar" value="mixed"/>
    </envelope-input>
</process>
<process name="loader"
    subsystem="com.ibm.wca.MassLoader.MassLoadSubSystem">
    <envelope-input xmlns='saf_params.xsd'>
        <param name="-dbname" type="scalar" value="%-dbname%"/>
        <param name="-dbuser" type="scalar" value="%-dbuser%"/>
        <param name="-dbpwd" type="scalar" value="%-dbpwd%"/>
        <param name="-commitcount" type="scalar" value="1000"/>
        <param name="-infile" type="file" reside="local" value="%previousOutFileAsURI%"/>
        <param name="-method" type="scalar" value="sqlimport"/>
        <param name="-nopriamry" type="scalar" value="insert"/>
    </envelope-input>
</process>
<process name="loaderFirstProcess"
    subsystem="com.ibm.wca.MassLoader.MassLoadSubSystem">
    <envelope-input xmlns='saf_params.xsd'>
        <param name="-dbname" type="scalar" value="%-dbname%"/>
        <param name="-dbuser" type="scalar" value="%-dbuser%"/>
        <param name="-dbpwd" type="scalar" value="%-dbpwd%"/>
        <param name="-commitcount" type="scalar" value="1000"/>
        <param name="-infile" type="file" reside="local" value="%tempFileURI%"/>
        <param name="-method" type="scalar" value="sqlimport"/>
        <param name="-nopriamry" type="scalar" value="insert"/>
    </envelope-input>
</process>
<process name="loaderForDelete"
    subsystem="com.ibm.wca.MassLoader.MassLoadSubSystem">
    <envelope-input xmlns='saf_params.xsd'>

```

```

        <param name="-dbname" type="scalar" value="%-dbname%"/>
        <param name="-dbuser" type="scalar" value="%-dbuser%"/>
        <param name="-dbpwd" type="scalar" value="%-dbpwd%"/>
        <param name="-commitcount" type="scalar" value="1000"/>
        <param name="-infile" type="file" reside="local" value="%previousOutFileAsURI%"/>
        <param name="-delete" type="scalar" value=""/>
    </envelope-input>
</process>
<process name="loaderForDeleteFirstProcess"
    subsystem="com.ibm.wca.MassLoader.MassLoadSubSystem">
    <envelope-input xmlns='saf_params.xsd'>
        <param name="-dbname" type="scalar" value="%-dbname%"/>
        <param name="-dbuser" type="scalar" value="%-dbuser%"/>
        <param name="-dbpwd" type="scalar" value="%-dbpwd%"/>
        <param name="-commitcount" type="scalar" value="1000"/>
        <param name="-infile" type="file" reside="local" value="%tempFileURI%"/>
        <param name="-delete" type="scalar" value=""/>
    </envelope-input>
</process>
<process name="saveToFile"
    cmd="cmd.exe /c c:\temp\theBatchFile.bat"
    args="-infile %tempFilePath% -dbname %-dbname%"
/>
</processSet>

```

**Remarque :** Les références aux fichiers respectent les majuscules et minuscules.

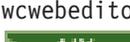
Le tableau suivant contient la liste des variables de substitution valides que l'application comprend.

| Variable de substitution % | Retour                                                                                                                                                                                                                                                                                                                                                                  |
|----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| %-dbname%                  | Nom de la base de données en cours                                                                                                                                                                                                                                                                                                                                      |
| %-dbuser%                  | Nom de l'utilisateur de la base de données                                                                                                                                                                                                                                                                                                                              |
| %-dtdname%                 | Emplacement URI du fichier DTD pour les fichiers XML                                                                                                                                                                                                                                                                                                                    |
| %-dbpwd%                   | Mot de passe associé au nom de l'utilisateur de la base de données                                                                                                                                                                                                                                                                                                      |
| %tempFilePath%             | Chemin complet du fichier temporaire                                                                                                                                                                                                                                                                                                                                    |
| %tempFile1Path%            | Il s'agit de noms de fichier temporaire uniques. Ils peuvent être placés dans le modèle de syntaxe d'une définition d'enveloppe ou d'une ligne de commande. Par exemple, si %tempFilePath% est placé dans l'attribut valeur du paramètre -infile dans un modèle d'enveloppe, Web editor écrit les données de la liste de travail à l'emplacement du fichier temporaire. |
| %tempFile2Path%            |                                                                                                                                                                                                                                                                                                                                                                         |
| Sur le sous-système        | URI temporaire                                                                                                                                                                                                                                                                                                                                                          |
| %tempFileURI%              | Les URI temporaires sont les URI des fichiers représentés par %tempFilePath%...%tempFile2Path%. Il ne s'agit pas d'un ensemble de fichiers supplémentaire, mais d'une façon d'extraire le même fichier temporaire généré, renvoyé avec une autre syntaxe.                                                                                                               |
| %tempFileURI1%             |                                                                                                                                                                                                                                                                                                                                                                         |
| %tempFileURI2%             |                                                                                                                                                                                                                                                                                                                                                                         |
| %previousOutFileAsURI%     | Fournit une représentation URI du paramètre -outfile des tâches précédentes.                                                                                                                                                                                                                                                                                            |
| %webEditorDir%             | Répertoire d'installation de Web editor                                                                                                                                                                                                                                                                                                                                 |

---

## Modification du fichier `weditor.xml`

Web editor utilise le fichier `weditor.xml` comme feuille de style XSL par défaut. Ce fichier se trouve dans le répertoire suivant :

-  `unité:\WebSphere\CommerceServer\xml\wcweditor\xml`
-  `unité:\Program Files\WebSphere\CommerceServer\xml\wcweditor\xml`
-  `/usr/WebSphere/CommerceServer/xml/wcweditor/xml`
-   `/opt/WebSphere/CommerceServer/xml/wcweditor/xml`
-  `/instroot/xml/wcweditor/xml`

En modifiant le fichier `weditor.xml`, un administrateur peut changer le format de sortie Web editor. Voici un exemple de contenu d'un fichier `weditor.xml` :

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
<xsl:output method="html"/>
  <!-- Valeur maximale des zones d'entrée sur une seule ligne. Pour les zones plus longues,
        une zone de texte TEXTAREA est créée -->
  <xsl:variable name="maxEntryFieldSize" select="80"/>
  <xsl:template match="/">
    <xsl:apply-templates/>
  </xsl:template>
  <!-- Zone en lecture seule -->
  <xsl:template match="readOnly" name="readOnly">
    <xsl:element name="input">
      <xsl:attribute name="name"><xsl:value-of select="@name"/></xsl:attribute>
      <xsl:attribute name="type">hidden</xsl:attribute>
      <xsl:attribute name="value"><xsl:value-of select="@defaultValue"/></xsl:attribute>
      <!--
      <xsl:attribute name="onFocus">this.blur()</xsl:attribute>
      <xsl:attribute name="style">border-style:groove</xsl:attribute>
      -->
    </xsl:element>
    <table border="1" cellpadding="0" cellspacing="0" width="155" bgcolor="#C0C0C0">
      <tr>
        <td>
          <xsl:value-of select="@defaultValue"/>
        </td>
      </tr>
    </table>
  </xsl:template>
</xsl:stylesheet>
```



---

## Chapitre 12. Gestion des catalogues

Les procédures suivantes expliquent comment gérer les catalogues en utilisant Web editor pour ajouter, modifier et supprimer des enregistrements dans les tables de base de données contenant les données de catalogue.

---

### Ajout d'un enregistrement à une table à l'aide de Web editor

Pour ajouter un enregistrement à une table à l'aide de Web editor, procédez comme suit :

1. A partir d'un navigateur Web, ouvrez l'adresse URL suivante :

`https://nom_hôte:8000/wcm/webeditor`

où *nom\_hôte* est le nom d'hôte HTTP qualifié complet du serveur WebSphere Application Server.

La fenêtre de connexion à la base de données Web editor s'affiche.

2. Entrez votre nom d'utilisateur et votre mot de passe de connexion à la base de données, puis cliquez sur **Connexion**.

Web editor s'affiche dans la fenêtre du navigateur et présente une liste de noms de table dans la barre de menus de gauche.

3. Cliquez sur le lien hypertexte approprié dans le sous-menu Ajout de la barre de menus de gauche.

Le formulaire approprié est alors affiché.

4. Saisissez toutes les données nécessaires dans le formulaire.

5. Cliquez sur **Transférer dans la liste de travail**.

Les résultats de la liste de travail concernant ce formulaire sont affichés. Ces résultats contiennent la totalité des modifications, ajouts et suppressions que vous avez effectués, mais ni supprimés, ni traités au cours de cette session de travail Web editor.

6. Si vous voulez supprimer des modifications d'enregistrement de la liste de travail, cochez la case figurant en regard de chaque modification d'enregistrement à supprimer, puis cliquez sur **Effacer élément sélectionné**.

7. Cliquez sur **Traiter la liste de travail** pour envoyer les modifications sélectionnées à la base de données.

Une page d'état affiche un message annonçant que le processus a aboutit.

8. Naviguez sur le site Web, cliquez sur les liens hypertexte appropriés et vérifiez que les modifications ont été effectuées.

---

## Modification d'un enregistrement dans une table à l'aide de Web editor

Pour modifier un enregistrement dans une table à l'aide de Web editor, procédez comme suit :

1. A partir d'un navigateur Web, ouvrez l'adresse URL suivante :

`https://nom_hôte:8000/wcm/webeditor`

où *nom\_hôte* est le nom d'hôte HTTP qualifié complet du serveur WebSphere Application Server.

La fenêtre de connexion à la base de données Web editor s'affiche.

2. Entrez votre nom d'utilisateur et votre mot de passe de connexion à la base de données, puis cliquez sur **Connexion**.

Web editor s'affiche dans la fenêtre du navigateur et présente une liste de noms de table dans la barre de menus de gauche.

3. Cliquez sur le lien hypertexte approprié dans le sous-menu Recherche de la barre de menus de gauche.

La page de recherche appropriée s'affiche.

4. Indiquez vos critères de recherche en procédant comme suit :

- a. Cochez la case figurant en regard de chaque attribut que vous voulez spécifier dans votre recherche.
- b. Utilisez le menu déroulant approprié pour chaque attribut sélectionné afin de choisir la logique que vous voulez utiliser dans la recherche.
- c. Dans la zone suivante de chaque attribut sélectionné, saisissez ou sélectionnez la valeur que vous voulez utiliser dans la recherche.

5. Cliquez sur **Rechercher**.

Les critères de recherche sont alors transmis à Web editor.

Une page d'état s'affiche et indique le nombre d'enregistrements répondant aux critères de recherche.

6. Effectuez l'une des opérations suivantes :

- Cliquez sur **Chargement des données** pour voir la liste des enregistrements trouvés.

Web editor affiche la liste des enregistrements extraits à partir de votre requête.

Passez à l'étape 6.

- Cliquez sur **Nouvelle recherche** pour retourner à la page de recherche.

Repassez à l'étape 3.

7. Sélectionnez l'enregistrement que vous voulez modifier.

Le formulaire approprié est alors affiché.

8. Faites défiler le formulaire jusqu'à la zone que vous voulez modifier, puis corrigez le contenu de cette dernière.

9. Cliquez sur **Transférer dans la liste de travail**.

Les résultats de la liste de travail concernant ce formulaire sont affichés. Ces résultats contiennent la totalité des modifications, ajouts et suppressions que vous avez effectués, mais ni supprimés, ni traités au cours de cette session de travail Web editor.

10. Si vous voulez supprimer des modifications d'enregistrement de la liste de travail, cochez la case figurant en regard de chaque modification d'enregistrement à supprimer, puis cliquez sur **Effacer élément sélectionné**.

11. Cliquez sur **Traiter la liste de travail** pour envoyer les modifications sélectionnées à la base de données.  
Une page d'état affiche un message annonçant que le processus a aboutit.
12. Naviguez sur le site Web, cliquez sur les liens hypertexte appropriés et vérifiez que les modifications ont été effectuées.

---

## Suppression d'un enregistrement d'une table à l'aide de Web editor

Pour supprimer un enregistrement d'une table à l'aide de Web editor, procédez comme suit :

1. A partir d'un navigateur Web, ouvrez l'adresse URL suivante :

`https://nom_hôte:8000/wcm/webeditor`

où *nom\_hôte* est le nom d'hôte HTTP qualifié complet du serveur WebSphere Application Server.

La fenêtre de connexion à la base de données Web editor s'affiche.

2. Entrez votre nom d'utilisateur et votre mot de passe de connexion à la base de données, puis cliquez sur **Connexion**.  
Web editor s'affiche dans la fenêtre du navigateur et présente une liste de noms de table dans la barre de menus de gauche.
3. Cliquez sur le lien hypertexte approprié dans le sous-menu Recherche de la barre de menus de gauche.  
La page de recherche appropriée s'affiche.
4. Indiquez vos critères de recherche en procédant comme suit :
  - a. Cochez la case figurant en regard de chaque attribut que vous voulez spécifier dans votre recherche.
  - b. Utilisez le menu déroulant approprié pour chaque attribut sélectionné afin de choisir la logique que vous voulez utiliser dans la recherche.
  - c. Dans la zone suivante de chaque attribut sélectionné, saisissez ou sélectionnez la valeur que vous voulez utiliser dans la recherche.
5. Cliquez sur **Rechercher**.  
Les critères de recherche sont alors transmis à Web editor.  
Une page d'état s'affiche et indique le nombre d'enregistrements répondant aux critères de recherche.
6. Effectuez l'une des opérations suivantes :
  - Cliquez sur **Chargement des données** pour voir la liste des enregistrements trouvés.  
Web editor affiche la liste des enregistrements extraits à partir de votre requête.  
Passez à l'étape 6.
  - Cliquez sur **Nouvelle recherche** pour retourner à la page de recherche.  
Reprenez à l'étape 3.
7. Cochez la case figurant en regard de chaque enregistrement que vous voulez supprimer.
8. Cliquez sur **Transférer dans la liste des suppressions**.
9. Cliquez sur le lien hypertexte approprié dans le sous-menu Liste de travail de la barre de menus de gauche.

Les résultats de la liste de travail concernant ce formulaire sont affichés. Ces résultats contiennent la totalité des modifications, ajouts et suppressions que vous avez effectués, mais ni supprimés, ni traités au cours de cette session de travail Web editor.

10. Si vous voulez supprimer des modifications d'enregistrement de la liste de travail, cochez la case figurant en regard de chaque modification d'enregistrement à supprimer, puis cliquez sur **Effacer élément sélectionné**.
11. Cliquez sur **Traiter la liste de travail** pour envoyer les modifications à la base de données.  
Une page d'état affiche un message annonçant que le processus a aboutit.
12. Naviguez sur le site Web, cliquez sur les liens hypertexte appropriés et vérifiez que les modifications ont été effectuées.

---

## **Partie 4. Informations de référence sur les instructions**



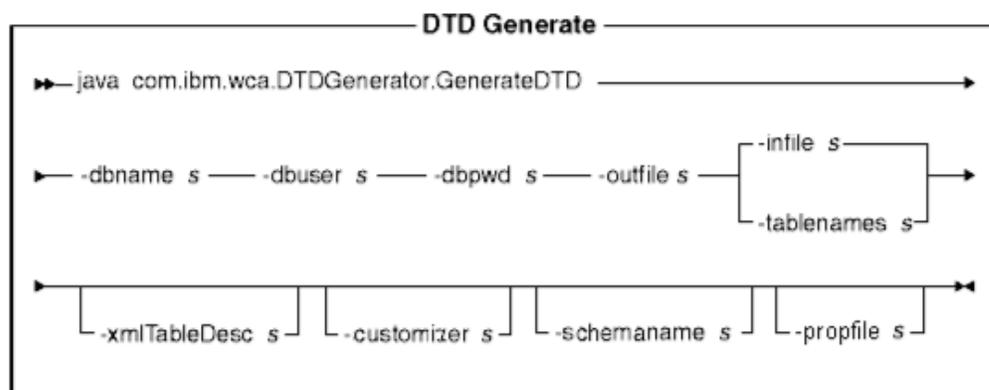
---

## Chapitre 13. Instruction DTD Generate

Cette instruction crée des fichiers DTD et des fichiers schémas à utiliser avec le module Loader.

---

### Instruction DTD Generate pour Windows, AIX, Linux et Solaris



#### Remarques :

1. Le diagramme ci-dessus constitue principalement une référence pour les paramètres de commande. Le fichier ou script de commande fourni pour cette commande et répertorié sous Chapitre 10, «Configuration des instructions et scripts du module Loader» à la page 73 fait office d'encapsuleur pour la commande Java et accepte les mêmes paramètres ; par conséquent, il est recommandé d'utiliser le fichier ou script de commande plutôt que d'appeler directement la commande Java.
2. Les noms de fichiers spécifiés en tant que paramètres pour cette commande peuvent être précédés par des chemins relatifs ou absolus.

#### Valeurs des paramètres :

##### **-dbname**

Nom de la base de données cible

##### **-dbuser**

Nom de l'utilisateur qui se connecte à la base de données

##### **-dbpwd**

Mot de passe de l'utilisateur qui se connecte à la base de données

##### **-outfile**

Nom du fichier DTD en sortie

**-infile** Nom du fichier en entrée contenant un nom de table de base de données sur chaque ligne

##### **-tablenames**

Noms des tables, séparés par des virgules

##### **-xmlTableDesc**

Chemin du fichier schéma à créer

**-customizer**

Nom du fichier de propriétés de personnalisation à utiliser.  
DB2ConnectionCustomizer.properties est le fichier par défaut. Le fichier de propriétés de personnalisation peut être spécifié comme dans l'exemple suivant :

```
-customizer d:\wc\prop\dttdgen.properties
```

Si ce fichier existe dans un répertoire spécifié dans la variable d'environnement système CLASSPATH (chemin de classe), le même fichier peut être spécifié comme dans l'exemple suivant :

```
-customizer dttdgen
```

**-schemaname**

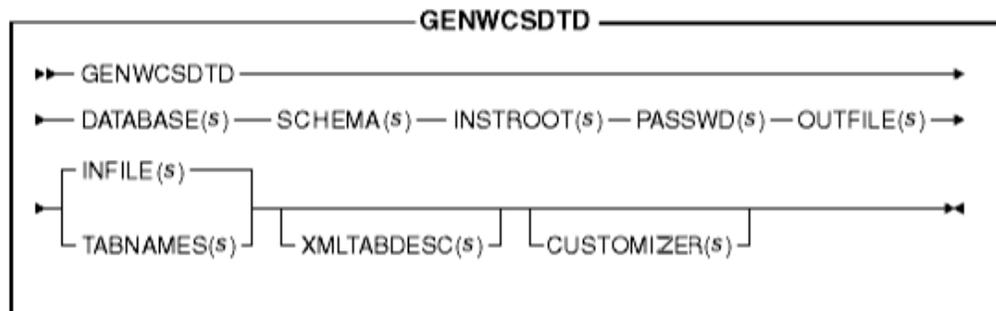
Nom du schéma de la base de données cible

**-propFile**

Fichier contenant des propriétés, tel qu'un fichier de propriétés externe dans lequel peuvent être stockés le texte d'aide, les valeurs par défaut et les informations de description de zone pour la description d'un formulaire Web editor.

---

## Instruction DTD Generate pour iSeries



**Remarque :** Les noms de fichiers spécifiés en tant que paramètres pour cette commande peuvent être précédés par des chemins relatifs ou absolus.

**Valeurs des paramètres :****DATABASE**

Nom de la base de données cible, tel qu'il est affiché dans le répertoire de la base de données relationnelle

**SCHEMA**

Nom du schéma de la base de données cible, qui est identique au nom de l'instance

**INSTROOT**

Nom complet du chemin root de l'instance WebSphere Commerce, tel que /QIBM/UserData/WebCommerce/instances/nom\_instance

**PASSWD**

Mot de passe de l'instance WebSphere Commerce

**OUTFILE**

Nom du fichier DTD en sortie

**INFILE**

Nom du fichier en entrée contenant un nom de table de base de données sur chaque ligne

**TABNAMES**

Noms des tables, séparés par des virgules

**XMLTABDESC**

Chemin d'accès du fichier schéma à créer. Ce paramètre est facultatif.

**CUSTOMIZER**

Nom du fichier de propriétés de personnalisation à utiliser. Le fichier par défaut est `ISeries_GENWCSDTD_Customizer.properties`. Le fichier de propriétés de personnalisation peut être spécifié comme dans l'exemple suivant :

```
CUSTOMIZER(/wc/prop/dtdgen.properties)
```

Si ce fichier existe dans un répertoire spécifié dans la variable d'environnement système `CLASSPATH` (chemin de classe), le même fichier peut être spécifié comme dans l'exemple suivant :

```
CUSTOMIZER(dtdgen)
```



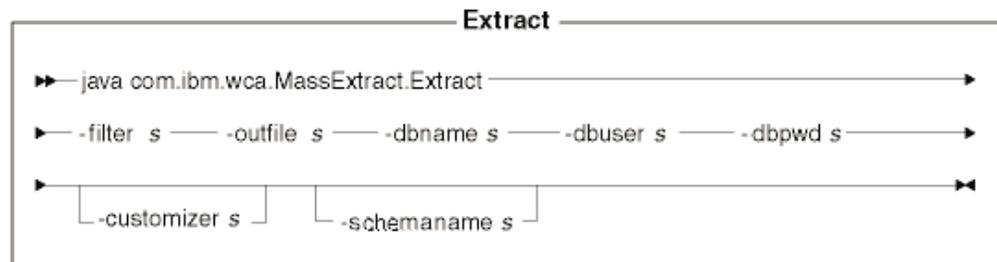
---

## Chapitre 14. Instruction Extract

Cette instruction extrait un sous-ensemble de données de la base de données sous la forme d'un fichier XML.

---

### Instruction Extract pour Windows, AIX, Linux et Solaris



#### Remarques :

1. Le diagramme ci-dessus constitue principalement une référence pour les paramètres de commande. Le fichier ou script de commande fourni pour cette commande et répertorié sous Chapitre 10, «Configuration des instructions et scripts du module Loader» à la page 73 fait office d'encapsuleur pour la commande Java et accepte les mêmes paramètres ; par conséquent, il est recommandé d'utiliser le fichier ou script de commande plutôt que d'appeler directement la commande Java.
2. Les noms de fichiers spécifiés en tant que paramètres pour cette commande peuvent être précédés par des chemins relatifs ou absolus.

#### Valeurs des paramètres :

**-filter** Nom du fichier d'extraction et de filtrage

**-outfile**

Nom du fichier XML en sortie dans lequel les données extraites vont être stockées

**-dbname**

Nom de la base de données à partir de laquelle les données sont extraites

**-dbuser**

Nom utilisateur de connexion à la base données à partir de laquelle les données sont extraites

**-dbpwd**

Mot de passe associé au nom utilisateur de connexion à la base données à partir de laquelle les données sont extraites

**-customizer**

Nom du fichier de propriétés de personnalisation à utiliser.

DB2ConnectionCustomizer.properties est le fichier par défaut. Le fichier de propriétés de personnalisation peut être spécifié comme dans l'exemple suivant :

-customizer d:\wc\prop\extract.properties

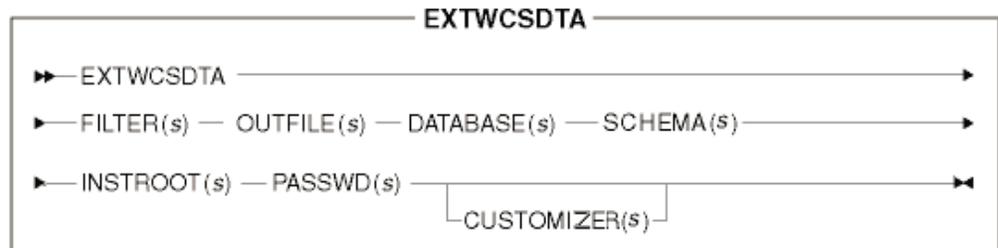
Si ce fichier existe dans un répertoire spécifié dans la variable d'environnement système CLASSPATH (chemin de classe), le même fichier peut être spécifié comme dans l'exemple suivant :

```
-customizer extract
```

**-schemaname**

Nom du schéma de la base de données cible

## Instruction Extract pour iSeries



**Remarque :** Les noms de fichiers spécifiés en tant que paramètres pour cette commande peuvent être précédés par des chemins relatifs ou absolus.

**Valeurs des paramètres :**

### **FILTER**

Nom du fichier d'extraction et de filtrage

### **OUTFILE**

Nom du fichier XML en sortie dans lequel les données extraites vont être stockées

### **DATABASE**

Nom de la base de données à partir de laquelle les données sont extraites, tel qu'il est affiché dans le répertoire de la base de données relationnelle

### **SCHEMA**

Nom du schéma de la base de données à partir de laquelle les données sont extraites, qui est identique au nom de l'instance

### **INSTROOT**

Nom complet du chemin root de l'instance WebSphere Commerce, tel que `/QIBM/UserData/WebCommerce/instances/nom_instance`

### **PASSWD**

Mot de passe de l'instance WebSphere Commerce

### **CUSTOMIZER**

Nom du fichier de propriétés de personnalisation à utiliser. Le fichier par défaut est `ISeries_EXTWCSDTA_Customizer.properties`. Le fichier de propriétés de personnalisation peut être spécifié comme dans l'exemple suivant :

```
CUSTOMIZER(/wc/prop/extract.properties)
```

Si ce fichier existe dans un répertoire spécifié dans la variable d'environnement système CLASSPATH (chemin de classe), le même fichier peut être spécifié comme dans l'exemple suivant :

```
CUSTOMIZER(extract)
```

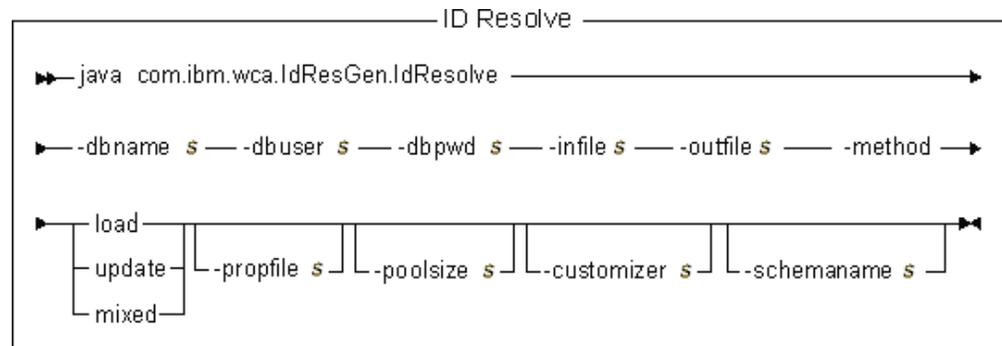
---

## Chapitre 15. Instruction ID Resolve

Cette instruction génère des identificateurs pour les éléments XML qui en ont besoin pour le chargement dans la base de données.

---

### Instruction ID Resolve pour Windows, AIX, Linux et Solaris



#### Remarques :

1. Le diagramme ci-dessus constitue principalement une référence pour les paramètres de commande. Le fichier ou script de commande fourni pour cette commande et répertorié sous Chapitre 10, «Configuration des instructions et scripts du module Loader» à la page 73 fait office d'encapsuleur pour la commande Java et accepte les mêmes paramètres ; par conséquent, il est recommandé d'utiliser le fichier ou script de commande plutôt que d'appeler directement la commande Java.
2. Les noms de fichiers spécifiés en tant que paramètres pour cette commande peuvent être précédés par des chemins relatifs ou absolus.

#### Valeurs des paramètres :

##### -dbname

Nom de la base de données cible

##### -dbuser

Nom de l'utilisateur qui se connecte à la base de données

##### -dbpwd

Mot de passe de l'utilisateur qui se connecte à la base de données

**-infile** Nom du document XML en entrée contenant les descripteurs des enregistrements de la table

##### -outfile

Nom du fichier XML en sortie devant être créé ; ce fichier peut être utilisé en entrée dans l'utilitaire Loader

##### -method

Méthode de traitement du fichier en entrée

- Utilisez la méthode de chargement pour traiter le fichier d'entrée si *aucun* enregistrement du fichier *n'existe* dans la base de données.
- Utilisez la méthode de mise à jour pour traiter le fichier d'entrée si *tous* les enregistrements du fichier *existent* dans la base de données.

- Utilisez la méthode mixte pour traiter le fichier d'entrée si *seulement certains* enregistrements du fichier *existent* dans la base de données.

La méthode par défaut est le chargement.

**-profile**

Fichier texte contenant les propriétés Java sous la forme de paires nom-valeur. Ce fichier est utilisé pour définir les noms des colonnes associatives pour les recherches d'identificateurs de clé associée et le prédicat de sélection pour les requêtes concernant les tables principales (CATEGORY et PRODUCT, par exemple). Vous pouvez omettre dans ce fichier les entrées concernant les tables qui comportent un index à entrées uniques défini ne contenant pas d'identificateur. Le fichier par défaut est `IdResolveKeys.properties`. Bien que ce fichier soit suivi de `.properties`, n'utilisez pas cette extension lorsque vous spécifiez la valeur.

**-poolsize**

Nombre d'identificateurs à réserver. La valeur par défaut est 50.

**-customizer**

Nom du fichier de propriétés de personnalisation à utiliser.

`DB2ConnectionCustomizer.properties` est le fichier par défaut. Le fichier de propriétés de personnalisation peut être spécifié comme dans les exemples suivants :

```
-customizer d:\wc\prop\idres.properties
```

```
-customizer d:\wc\prop\idres
```

Si ce fichier existe dans le répertoire en cours, le même fichier peut être spécifié comme dans l'exemple suivant :

-customizer idres.properties

Si ce fichier existe dans un répertoire spécifié dans la variable d'environnement système CLASSPATH (chemin de classe), le même fichier peut être spécifié comme dans l'exemple suivant :

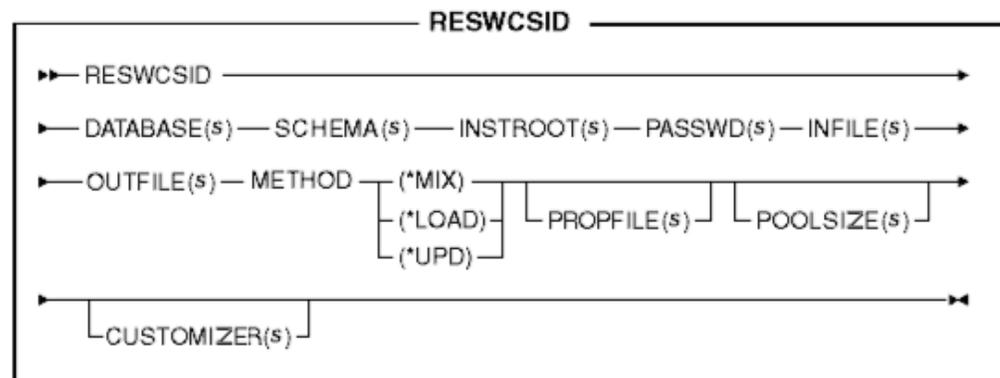
-customizer idres

**-schemaname**

Nom du schéma de la base de données cible

---

## Instruction ID Resolve pour iSeries



**Remarque :** Les noms de fichiers spécifiés en tant que paramètres pour cette commande peuvent être précédés par des chemins relatifs ou absolus.

**Valeurs des paramètres :**

**DATABASE**

Nom de la base de données cible, tel qu'il est affiché dans le répertoire de la base de données relationnelle

**SCHEMA**

Nom du schéma de la base de données cible, qui est identique au nom de l'instance

**INSTROOT**

Nom complet du chemin root de l'instance WebSphere Commerce, tel que /QIBM/UserData/WebCommerce/instances/nom\_instance

**PASSWD**

Mot de passe de l'instance WebSphere Commerce

**INFILE**

Nom du document XML en entrée contenant les descripteurs des enregistrements de la table

**OUTFILE**

Nom du fichier XML en sortie devant être créé ; ce fichier peut être utilisé en entrée dans l'utilitaire Loader

**METHOD**

Méthode de traitement du fichier en entrée

- Utilisez la méthode de chargement (\*LOAD) pour traiter le fichier d'entrée si *aucun* enregistrement du fichier *n'existe* dans la base de données.
- Utilisez la méthode de mise à jour (\*UPD) pour traiter le fichier d'entrée si *tous* les enregistrements du fichier *existent* dans la base de données.
- Utilisez la méthode mixte (\*MIX) pour traiter le fichier d'entrée si *seulement certains* enregistrements du fichier *existent* dans la base de données.

### PROFILE

Fichier texte contenant les propriétés Java sous la forme de paires nom-valeur. Ce fichier est utilisé pour définir les noms des colonnes associatives pour les recherches d'identificateurs de clé associée et le prédicat de sélection pour les requêtes concernant les tables principales (CATEGORY et PRODUCT, par exemple). Vous pouvez omettre dans ce fichier les entrées concernant les tables qui comportent un index à entrées uniques défini ne contenant pas d'identificateur. Le fichier par défaut est `IdResolveKeys.properties`. Bien que ce fichier soit suivi de `.properties`, n'utilisez pas cette extension lorsque vous spécifiez la valeur.

### POOLSIZE

Nombre d'identificateurs à réserver. La valeur par défaut est 50.

### CUSTOMIZER

Nom du fichier de propriétés de personnalisation à utiliser. Le fichier par défaut est `ISeries_RESWCSID_Customizer.properties`. Le fichier de propriétés de personnalisation peut être spécifié comme dans les exemples suivants :

```
CUSTOMIZER(/wc/prop/idres.properties)
CUSTOMIZER(/wc/prop/idres)
```

Si ce fichier existe dans le répertoire en cours, le même fichier peut être spécifié comme dans l'exemple suivant :

```
CUSTOMIZER(idres.properties)
```

Si ce fichier existe dans un répertoire spécifié dans la variable d'environnement système CLASSPATH (chemin de classe), le même fichier peut être spécifié comme dans l'exemple suivant :

```
CUSTOMIZER(idres)
```

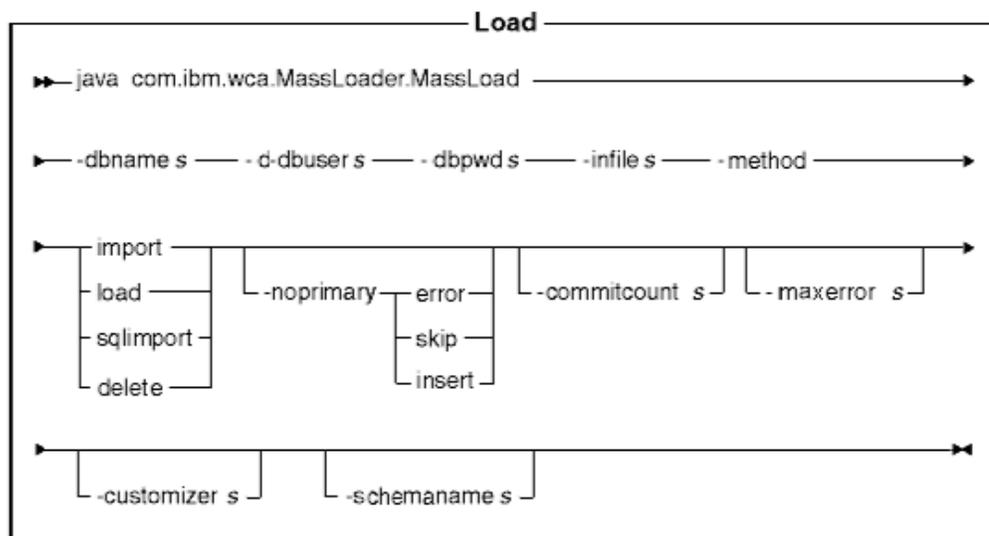
---

## Chapitre 16. Instruction Load

Cette instruction charge un fichier XML d'entrée dans une base de données cible.

---

### Instruction Load pour Windows, AIX, Linux et Solaris



#### Remarques :

1. Le diagramme ci-dessus constitue principalement une référence pour les paramètres de commande. Le fichier ou script de commande fourni pour cette commande et répertorié sous Chapitre 10, «Configuration des instructions et scripts du module Loader» à la page 73 fait office d'encapsuleur pour la commande Java et accepte les mêmes paramètres ; par conséquent, il est recommandé d'utiliser le fichier ou script de commande plutôt que d'appeler directement la commande Java.
2. Les noms de fichiers spécifiés en tant que paramètres pour cette commande peuvent être précédés par des chemins relatifs ou absolus.

#### Valeurs des paramètres :

##### **-dbname**

Nom de la base de données cible

##### **-dbuser**

Nom de l'utilisateur qui se connecte à la base de données

##### **-dbpwd**

Mot de passe de l'utilisateur qui se connecte à la base de données

**-infile** Nom du fichier XML d'entrée

##### **-method**

Mode de fonctionnement de Loader lors de l'insertion de données dans la base de données

- La méthode de chargement utilise le programme de chargement natif du fournisseur de la base de données. Vous pouvez utiliser la méthode de

chargement pour les bases de données Oracle aussi bien locales qu'éloignées, mais pour DB2, elle ne peut être utilisée que pour les bases de données locales.

- Pour charger des données dans des bases de données DB2 éloignées, utilisez la méthode d'importation. La méthode d'importation utilise l'option importation ou mise à jour, si ces dernières sont disponibles auprès du fournisseur de la base de données. Si vous spécifiez cette méthode pour une base de données dans laquelle ces options ne sont pas disponibles, des instructions SQL utilisant JDBC sont utilisées pour la mise à jour de la base de données.
- La méthode d'importation SQL (sqlimport) peut être utilisée avec des bases de données locales et éloignées.
- La méthode de suppression permet de supprimer des données de la base de données.

Si vous utilisez la synchronisation de l'espace de recherche de l'Assistant produit, vous devez utiliser la méthode sqlimport ou la méthode de suppression.

**-noprimary**

Action devant être effectuée par Loader lorsque la clé primaire fait défaut pour un enregistrement dans le fichier d'entrée. L'option error indique que l'absence de la clé primaire doit être signalée comme erreur et provoquer l'arrêt du traitement. L'option skip permet d'ignorer tous les enregistrements d'un fichier d'entrée qui ne comportent pas de clé primaire. L'option insert correspond à une tentative de traitement (insertion ou suppression) des données. L'action par défaut est error.

**-commitcount**

Nombre d'enregistrements traités avant validation de la base de données lors de l'utilisation du mode de fonctionnement mise à jour SQL. La valeur par défaut est 1.

**-maxerror**

Nombre d'erreurs dont le dépassement entraîne l'arrêt du traitement de Loader, en mode de fonctionnement mise à jour SQL.

**-customizer**

Nom du fichier de propriétés de personnalisation à utiliser. Le fichier par défaut est `MassLoadCustomizer.properties`. Le fichier de propriétés de personnalisation peut être spécifié comme dans l'exemple suivant :

```
-customizer d:\wc\prop\ml.properties
```

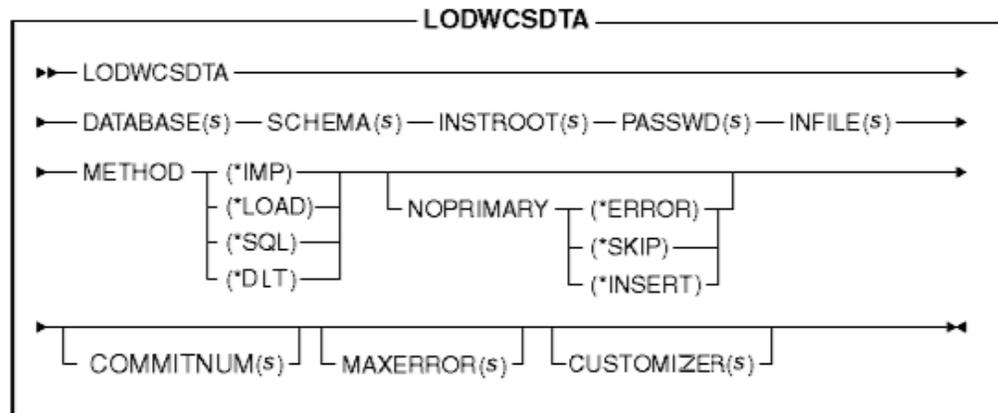
Si ce fichier existe dans un répertoire spécifié dans la variable d'environnement système CLASSPATH (chemin de classe), le même fichier peut être spécifié comme dans l'exemple suivant :

```
-customizer ml
```

**-schemaname**

Nom du schéma de la base de données cible

## Instruction Load pour iSeries



**Remarque :** Les noms de fichiers spécifiés en tant que paramètres pour cette commande peuvent être précédés par des chemins relatifs ou absolus.

### Valeurs des paramètres :

#### DATABASE

Nom de la base de données cible, tel qu'il est affiché dans le répertoire de la base de données relationnelle

#### SCHEMA

Nom du schéma de la base de données cible, qui est identique au nom de l'instance

#### INSTROOT

Nom complet du chemin root de l'instance WebSphere Commerce, tel que `/QIBM/UserData/WebCommerce/instances/nom_instance`

#### PASSWD

Mot de passe de l'instance WebSphere Commerce

#### INFILE

Nom du fichier XML d'entrée

#### METHOD

Mode de fonctionnement de Loader lors de l'insertion de données dans la base de données

- La méthode de chargement (\*LOAD) utilise le programme de chargement natif du fournisseur de la base de données. Vous pouvez utiliser la méthode de chargement (\*LOAD) pour les bases de données Oracle aussi bien locales qu'éloignées, mais pour DB2, elle ne peut être utilisée que pour les bases de données locales.
- Pour charger des données dans des bases de données DB2 éloignées, utilisez l'option d'importation (\*IMP). La méthode d'importation (\*IMP) utilise l'option d'importation ou de mise à jour lorsque ces dernières sont disponibles auprès du fournisseur de la base de données. Si ces options ne sont pas disponibles, des instructions SQL utilisant JDBC sont utilisées pour la mise à jour de la base de données.
- La méthode d'importation SQL (\*SQL) peut être utilisée avec des bases de données locales et éloignées.

- La méthode de suppression (\*DLT) permet de supprimer des données de la base de données.

**NOPRIMARY**

Action devant être effectuée par Loader lorsque la clé primaire fait défaut pour un enregistrement dans le fichier d'entrée. L'option error (\*ERROR) indique que l'utilitaire doit signaler la clé primaire manquante comme une erreur et clore l'opération. L'option skip (\*SKIP) permet d'ignorer tous les enregistrements d'un fichier d'entrée qui ne comportent pas de clé primaire. L'option insert (\*INSERT) tente de traiter (insérer ou supprimer) les données. L'action par défaut est error.

**COMMITNUM**

Nombre d'enregistrements traités avant validation de la base de données lors de l'utilisation du mode de fonctionnement mise à jour SQL. La valeur par défaut est 1.

**MAXERROR**

Nombre d'erreurs dont le dépassement entraîne l'arrêt du traitement de Loader, en mode de fonctionnement mise à jour SQL.

**CUSTOMIZER**

Nom du fichier de propriétés de personnalisation à utiliser. Le fichier par défaut est ISeries\_LODWCSDTA\_Customizer.properties. Le fichier de propriétés de personnalisation peut être spécifié comme dans l'exemple suivant :

```
CUSTOMIZER(/wc/prop/ml.properties)
```

Si ce fichier existe dans un répertoire spécifié dans la variable d'environnement système CLASSPATH (chemin de classe), le même fichier peut être spécifié comme dans l'exemple suivant :

```
CUSTOMIZER(ml)
```

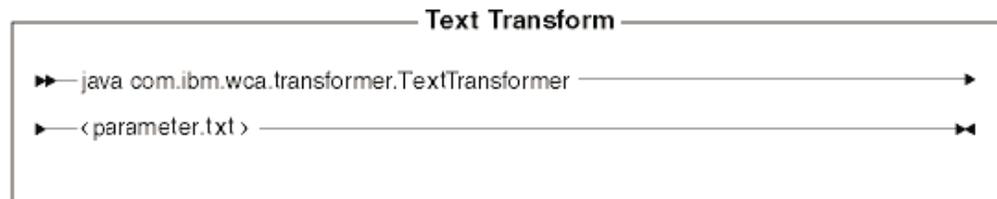
---

## Chapitre 17. Instruction Text Transform

Cette instruction convertit des données d'un format variable délimité par caractère en un format de données XML.

---

### Instruction Text Transform pour Windows, AIX, Linux et Solaris



**Remarque :** Le diagramme ci-dessus constitue principalement une référence pour les paramètres de commande. Le fichier ou script de commande fourni pour cette commande et répertorié sous Chapitre 10, «Configuration des instructions et scripts du module Loader» à la page 73 fait office d'encapsuleur pour la commande Java et accepte les mêmes paramètres ; par conséquent, il est recommandé d'utiliser le fichier ou script de commande plutôt que d'appeler directement la commande Java.

#### Valeurs des paramètres

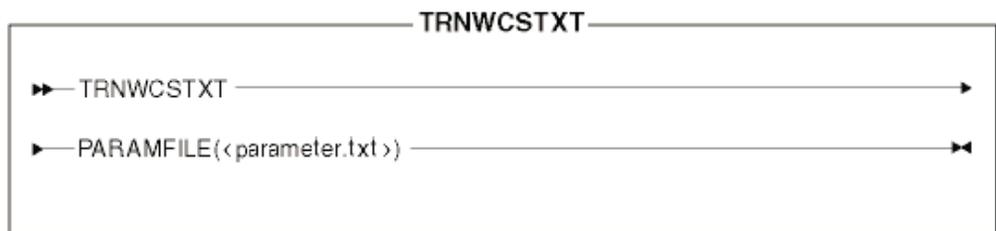
Les valeurs suivantes sont définies et séparées par des virgules dans un fichier de paramètres (*paramètres.txt*) :

- input file  
Nom du fichier à convertir
- schema file  
Nom du fichier schéma XML à utiliser pour la transformation
- output file  
Nom du fichier de sortie dans lequel les données converties vont être stockées
- transformation method  
Méthode utilisée pour ajouter les données dans le fichier de sortie. Indiquez **Create** si un nouveau fichier doit être créé ou **Append** si les données en sortie doivent être ajoutées à la fin d'un fichier de données existant.

**Remarque :** Ce fichier est également désigné par le terme fichier "manifeste" ou "commande".

---

## Instruction Text Transform pour iSeries



### Valeurs des paramètres

Les valeurs suivantes sont définies et séparées par des virgules dans un fichier de paramètres (*paramètres.txt*) :

- input file  
Nom du fichier à convertir
- schema file  
Nom du fichier schéma XML à utiliser pour la transformation
- output file  
Nom du fichier de sortie dans lequel les données converties vont être stockées
- transformation method  
Méthode utilisée pour ajouter les données dans le fichier de sortie. Indiquez **Create** si un nouveau fichier doit être créé ou **Append** si les données en sortie doivent être ajoutées à la fin d'un fichier de données existant.

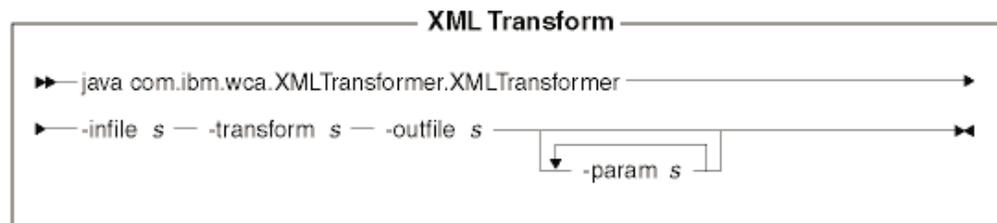
---

## Chapitre 18. Instruction XML Transform

Cette instruction permet de convertir un fichier XML dans un autre format XML.

---

### Instruction XML Transform pour Windows, AIX, Linux et Solaris



#### Remarques :

1. Le diagramme ci-dessus constitue principalement une référence pour les paramètres de commande. Le fichier ou script de commande fourni pour cette commande et répertorié sous Chapitre 10, «Configuration des instructions et scripts du module Loader» à la page 73 fait office d'encapsuleur pour la commande Java et accepte les mêmes paramètres ; par conséquent, il est recommandé d'utiliser le fichier ou script de commande plutôt que d'appeler directement la commande Java.
2. Les noms de fichiers spécifiés en tant que paramètres pour cette commande peuvent être précédés par des chemins relatifs ou absolus.

#### Valeurs des paramètres :

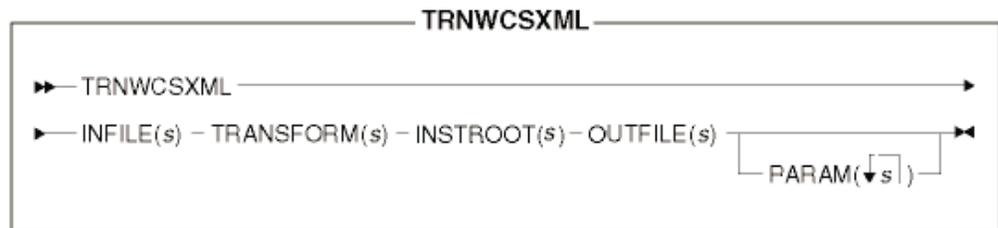
**-infile** Nom du fichier à convertir

**-transform**  
Nom du fichier de règles XSL de conversion

**-outfile**  
Nom du fichier XML en sortie dans lequel les données converties vont être stockées

**-param**  
Paramètres à transmettre au fichier de règles XSL. *Ce paramètre est facultatif.* Il peut être spécifié plusieurs fois pour transmettre plusieurs paires nom-valeur ("name=value").

## Instruction XML Transform pour iSeries



**Remarque :** Les noms de fichiers spécifiés en tant que paramètres pour cette commande peuvent être précédés par des chemins relatifs ou absolus.

### Valeurs des paramètres :

#### INFILE

Nom du fichier à convertir

#### TRANSFORM

Nom du fichier de règles XSL de conversion

#### INSTROOT

Nom complet du chemin root de l'instance WebSphere Commerce, tel que /QIBM/UserData/WebCommerce/instances/*nom\_instance*

#### OUTFILE

Nom du fichier XML en sortie dans lequel les données converties vont être stockées

#### PARAM

Paramètres à transmettre au fichier de règles XSL. *Ce paramètre est facultatif.* La chaîne indiquée peut contenir plusieurs valeurs pour transmettre plusieurs paires nom-valeur ("name=value").

---

## 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 d'informations sur les produits et services actuellement disponibles dans votre pays, contactez votre représentant 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. Cependant, il incombe à l'utilisateur d'évaluer et de vérifier lui-même le fonctionnement des produits, logiciels ou services non-IBM.

IBM peut détenir des brevets ou des demandes de brevet concernant les produits décrits 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

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 pas au Royaume-Uni, ni à tout autre pays où de telles dispositions sont incompatibles avec la loi en vigueur.** 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. Des modifications sont apportées périodiquement aux informations contenues dans le présent document et seront intégrées dans les nouvelles éditions. IBM se réserve le droit d'apporter à tout moment et sans préavis des améliorations et/ou des modifications aux produits et/ou aux programmes décrits.

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.

Pour plus d'informations sur : (i) l'échange d'informations entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci) et (ii) l'utilisation mutuelle des informations ainsi échangées, les licenciés doivent écrire à l'adresse suivante :

IBM Canada Ltd.  
Office of the Lab Director  
8200 Warden Avenue, Markham, Ontario L6G 1C7  
Canada

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'accord ICA (accord sur la clientèle d'IBM), 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 déclaration relative aux intentions d'IBM pour ses opérations à venir est susceptible d'être modifiée ou annulée sans préavis et ne constitue qu'un objectif.

Tous les tarifs indiqués sont les prix de vente actuels suggérés par IBM et sont susceptibles d'être modifiés sans préavis. Les tarifs appliqués peuvent varier selon les revendeurs.

Ces informations sont fournies uniquement à titre de planification. Elles sont susceptibles d'être modifiées avant la mise à disposition des produits décrits.

Le présent document contient des exemples de données et de rapports utilisés quotidiennement 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 SOUS COPYRIGHT :

Ces informations contiennent des exemples de programmes d'application en langue source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Les copies, modifications et distributions de ces exemples de programmes sont autorisées sous toute forme 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 à l'interface de programmation d'applications de la plateforme pour laquelle ils ont été écrits. 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. Les copies, les modifications et les distributions de ces exemples de programmes sont autorisées sous toute forme 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 d'applications IBM.

Toute copie totale ou partielle de ces exemples de programmes et des oeuvres qui en sont dérivées doit comprendre une notice de copyright, libellée comme suit :

©Copyright International Business Machines Corporation 2001.  
Portions of this code are derived from IBM Corp. Sample Programs.  
©Copyright IBM Corp. 2000, 2001. All rights reserved.

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

---

## Marques

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

AIX  
DB2  
DB2 Universal Database  
IBM  
iSeries  
OS/400  
WebSphere

Microsoft, Windows, Windows NT et Windows 2000 sont des marques de Microsoft Corporation aux Etats-Unis et/ou dans certains autres pays.

Oracle est une marque d'Oracle Corporation aux Etats-Unis et/ou dans certains autres pays.

Solaris, Java et toutes les marques et logos faisant référence à Java sont des marques de Sun Microsystems, Inc. aux Etats-Unis et/ou dans certains autres pays.

D'autres sociétés sont propriétaires des autres marques, noms de produits ou logos qui pourraient apparaître dans ce document.



**IBM**