



Guide de migration

Version 10.1



Guide de migration

Version 10.1

Important

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section «Remarques», à la page 83.

Deuxième édition - novembre 2012

Réf. US : SC19-3498-01

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

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

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

Vous pouvez également consulter les serveurs Internet suivants :

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

*Compagnie IBM France
Direction Qualité
17, avenue de l'Europe
92275 Bois-Colombes Cedex*

© Copyright IBM Corporation 2000, 2012.

Table des matières

Avis aux lecteurs canadiens	v
--	----------

Préface	vii
--------------------------	------------

Nouveautés de la version 10.1.0	vii
---	-----

Service de support logiciel IBM	xi
---	----

Chapitre 1. Migration	1
--	----------

Chapitre 2. Guide d'initiation	3
---	----------

Chapitre 3. Migration depuis des versions antérieures	5
--	----------

Migration des données maître du format interne vers le format XML.	6
--	---

Migration de V6.0.0 vers V10.1.0.	9
---	---

Migration de V9.1.0 vers V10.1.0	13
--	----

Migration de V10.0.0 vers V10.1.0	16
---	----

Migration de Global Data Synchronization de V1.3.x vers InfoSphere MDM Collaboration Server V10.1	18
---	----

Préparation à la migration de GDS 1.3.x.	19
--	----

Exportation à partir de WebSphere Product Center version 5.2.1.4	25
--	----

Importation vers InfoSphere MDM Collaboration Server V10.1	34
--	----

Vérification de votre environnement GDS migré	41
---	----

Compilation et débogage de scripts	41
--	----

Chapitre 4. Migration de scripts InfoSphere MDM Collaboration Server	45
---	-----------

Différences entre les scripts dans les diverses versions	45
--	----

Méthodes avec des signatures différentes dans V5.2.1 et V10.1.0.	46
--	----

Méthodes aux signatures différentes dans les versions 5.3.0 et 5.3.0.1	48
--	----

Méthodes avec des signatures différentes dans V5.3.0.1 et V10.1.0	48
---	----

Méthodes avec des signatures différentes dans V6.0.0 et V10.1.0.	49
--	----

Méthodes présentes dans la version 5.2.1	49
--	----

Méthodes disponibles uniquement dans la version 5.3.0.1	51
---	----

Méthodes migrées uniquement de V9.0.0 vers V10.1.0	51
--	----

Méthodes disponibles dans V5.3.2 uniquement migrées vers V10.1.0	56
--	----

Méthodes de V6.5.0 uniquement migrées vers V10.1.0	60
--	----

Modifications du comportement des opérations de script entre V5.2.1 et V10.1.0.	60
---	----

Modifications du comportement des opérations de script entre V5.3.0.1 et V10.1.0	63
--	----

Modifications du comportement des opérations de script entre V6.0.0 et V10.1.0.	64
---	----

Différences des API Java entre les diverses versions	65
--	----

Méthodes et énumérations API Java	65
---	----

Nouvelles interfaces de programme d'application (API) Java	67
--	----

Optimisation de la base de données	68
--	----

Transformation des métadonnées du modèle de données	68
---	----

Corrections par le transformateur XSLT	69
--	----

Migration manuelle des scripts	72
--	----

Chapitre 5. Identification des incidents	75
---	-----------

Identification et résolution des problèmes de migration	75
---	----

Utilisation de scripts de migration compilés	75
--	----

Erreurs de compilation de script courantes	75
--	----

Erreurs et incidents d'exécution courants	76
---	----

Résolution des erreurs et incidents d'exécution courants	77
--	----

Utilisation du script checkForCompileError.sh	78
---	----

Identification et résolution des problèmes de migration	78
---	----

Remarques	83
----------------------------	-----------

Marques	86
-------------------	----

Index	87
------------------------	-----------

Avis aux lecteurs canadiens

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

Illustrations

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

Terminologie

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

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

Claviers

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

OS/2 et Windows - Paramètres canadiens

Au Canada, on utilise :

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

Nomenclature

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

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

Brevets

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

Assistance téléphonique

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

Préface

Ces informations expliquent comment migrer d'IBM InfoSphere Master Data Management Server for Product Information Management, Versions 6.0.0 et 9.1.0, et de IBM® InfoSphere Master Data Management Collaboration Server Version 10.0.0 vers IBM InfoSphere Master Data Management Collaboration Server Version 10.1.0 en exécutant des sous-étapes et des migrations.

A propos de cette publication

Les informations du présent guide couvrent les rubriques suivantes : instructions pas à pas, pratiques recommandées, réglage des performances et résolution des incidents courants.

Public cible

Ce guide de migration est destiné aux installateurs, aux administrateurs système et au personnel des services de maintenance IBM afin de les aider à déployer correctement le produit dans votre environnement.

Informations sur les prérequis et informations associées

Avant d'utiliser ce guide de migration, familiarisez-vous avec les sources d'information suivantes :

- Le centre de documentation d'InfoSphere MDM Collaboration Server
- La documentation de votre serveur d'applications
- La documentation de votre logiciel de base de données

Envoi de commentaires

Vos commentaires sont essentiels pour nous aider à fournir des informations de première qualité et les plus précises possibles. Si vous avez des commentaires relatifs à ce guide ou à toute autre documentation InfoSphere MDM Collaboration Server, utilisez le formulaire suivant pour nous les faire parvenir :

<http://www.ibm.com/software/data/rcf/>

Nouveautés de la version 10.1.0

La version IBM InfoSphere Master Data Management Collaboration Server 10.1.0 comporte de nombreuses nouvelles fonctions.

InfoSphere MDM Collaboration Server est inclus dans InfoSphere MDM Enterprise Edition et dans InfoSphere MDM Collaborative Edition.

InfoSphere MDM Collaboration Server permet de créer un seul référentiel d'informations produit à jour, à utiliser à l'échelle d'une entreprise dans le cadre d'initiatives commerciales stratégiques.

Voici les principales modifications apportées au produit :

ACM (Advanced Catalog Management)

Advanced Catalog Management (ACM) for Commerce est une solution de InfoSphere MDM Collaboration Server. Il s'agit d'une configuration préconfigurée de InfoSphere MDM Collaboration Server prête à l'utilisation par les utilisateurs de WebSphere Commerce. Le modèle de données prêt à l'emploi couvre des workflows et des fonctions de sécurité et d'exportation pour WebSphere Commerce. La solution a pour vocation d'accélérer l'implémentation InfoSphere MDM Collaboration Server et Commerce. Cet accélérateur peut être modifié et étendu par un client ou une équipe d'implémentation. Pour plus d'informations, voir ACM for WebSphere Commerce.

Intégration IBM WebSphere Operational Decision Management (WODM)

Solution intégrée pour IBM WebSphere Operational Decision Management (WODM). Fournit des fonctions de création et d'association de règles métier avancées aux produits, ensembles, etc. La solution intégrée est un accélérateur qui peut être modifié et étendu par un client ou une équipe d'implémentation.

Edition en ligne

Fournit des fonctions d'édition en ligne pour la plupart des types d'attributs à la place des éditeurs contextuels actuels dans une édition unique ou multiple. Elle permet de passer d'un attribut à l'autre et fournit un mécanisme pour ouvrir l'éditeur de texte enrichi lorsque cela est nécessaire au lieu d'utiliser l'éditeur par défaut. Pour plus d'informations, voir Conseils d'utilisation des données maître.

Mise en cache de vue et de catalogue

Pour optimiser la mise en cache, utilisez ces valeurs de cache pour mettre en cache les ID de noeud et les chemins des noeuds. Pour plus d'informations, voir Définition des valeurs de cache pour activer la mise en cache des catalogues.

Affichage d'attribut de liaison

Permet de choisir un nom d'affichage pour les attributs de liaison. Pour plus d'informations, voir :

- Table de recherche d'attribut de relation
- Modification des valeurs d'attribut de type table de consultation
- Utilisation des attributs de lien

Prise en charge de la migration pour les utilisateurs InfoSphere MDM Collaboration Server

- Migration de V6.0.0 vers V10.1.0
- Migration de V9.1.0 vers V10.1.0
- Migration de V10.0.0 vers V10.1.0
- Conversion des données maître du format interne dans le format XML

Prise en charge de la migration pour les utilisateurs de Global Data Synchronization

- Migration de la Global Data Synchronization version 1.3.x vers InfoSphere MDM Collaboration Server version 10.1.0

Enrichissements des messages d'erreur et d'avertissement

Fournissent des messages d'erreur et d'avertissement plus explicites. Pour plus d'informations, voir Messages InfoSphere MDM Collaboration Server.

Mises à niveau du support logiciel

Les mises à niveau suivantes sont désormais prises en charge. Pour plus d'informations, voir Configuration système requise.

Mise à niveau de pile

WebSphere Application Server version 8.0

DB2 10

Enrichissements de l'interface utilisateur

La liste suivante répertorie les enrichissements apportés à l'interface utilisateur :

Réduction de l'arborescence du panneau de navigation lors de l'actualisation

Lors de l'utilisation du bouton d'actualisation, la vue du panneau de navigation de gauche est maintenue au lieu d'être réduite comme auparavant.

Recherche 'Between' sur les attributs numériques dans l'interface utilisateur

Activez la recherche 'Between' sur les attributs numériques dans la nouvelle interface utilisateur de recherche enrichie. Pour plus d'informations, voir Utilisation de la recherche dans l'interface utilisateur.

Fonctions de recherche OR dans l'interface utilisateur

Activez l'opération logique 'OR' entre deux recherches de prédicats dans la nouvelle interface utilisateur de recherche enrichie. Pour plus d'informations, voir Utilisation de la recherche dans l'interface utilisateur.

Exécution de recherches en arrière-plan

Le support de demande de recherche en arrière-plan a été ajouté. Cette fonction permet d'exécuter d'autres fonctions en utilisant l'interface graphique du produit au cours d'une recherche, sous la forme d'un travail en arrière-plan.

Navigation dans les éléments dans un écran d'édition unique

Après avoir lancé l'écran d'édition unique sur plusieurs entrées sélectionnées dans l'écran multi-édition, cette fonction permet d'accéder rapidement à ces entrées dans un écran d'édition unique sans revenir à l'écran multi-édition.

Masquage des boutons désactivés

Les boutons auxquels l'utilisateur n'a pas accès sont masqués.

Présentation dynamique des boutons

Le bouton de menu "Options" a été supprimé et toutes les options de menu qui étaient disponibles dans "Options" sont désormais disponibles comme boutons de niveau supérieur. Si tous les boutons de niveau supérieur ne peuvent pas être affichés dans la présentation par défaut dans la limite de largeur actuelle du panneau de contenu de droite, les boutons manquants sont disponibles avec le bouton de menu "Plus" à l'extrémité droite de l'écran. Pour plus d'informations, voir Ecrans d'édition des objet.

Enrichissements de l'écran de recherche

Les rubriques d'écran de recherche suivantes ont été modifiées :

- Recherche en fonction d'attributs d'indicateur
- Création de recherches de catalogue

- Création de recherches de hiérarchie
- Création de recherches dans les zones de collaboration

Tous les attributs, y compris les attributs non indexés peuvent être recherchés dans une recherche simple

La représentation des données XML a été ajoutée pour pouvoir rechercher les attributs non indexés. Cela permet d'accéder aux données InfoSphere MDM Collaboration Server dans un format plus lisible directement depuis la base de données.

Enrichissements GDS (Global Data Synchronization)

La liste suivante répertorie les enrichissements apportés à Global Data Synchronization :

Mises à niveau du pool de données GDS

Fournit le support des derniers schémas du pool de données GDSN.

Migration des correctifs depuis les versions précédentes

Les correctifs 9.1.0 IF7 et version 10.0 FP1 sont migrés vers les lignes de code GDS de la version 10.1.0.

Mise à niveau BMS V2.8 pour l'alignement sur les derniers standard GDSN :

Le composant GDS dans InfoSphere MDM Collaboration Server est mis à niveau pour le rendre conforme à la spécification BMS V2.8 qui est la spécification GDSN actuelle.

Possibilité d'abonnement en utilisant plusieurs GLN de détaillant tout en utilisant un GLN global comme identificateur dans le code XML sortant pour ce détaillant :

ainsi, les utilisateurs peuvent gérer les scénarios dans lesquels les détaillants peuvent disposer de plusieurs GLN dans leurs organisations en utilisant ceux avec lesquels ils peuvent créer des abonnements. Dans leur pool de données, cette sous-organisation dans le détaillant est identifié par un GLN d'organisation unique.

Possibilité d'effectuer des recherches en fonction du GTIN dans *Manage Message > Explore Screen* :

Cette fonction est une extension des fonctions de recherche du module **Manage Message > Explore Screen**. En utilisant GTIN comme critère de recherche, vous pouvez rechercher tous les CIN ayant le GTIN dans le niveau le plus haut de la hiérarchie logistique.

Possibilité d'effectuer des recherches en fonction de l'option *All* de la liste déroulante *Transaction Process* dans *Manage Message > Explore Screen*:

Les processus de transaction ici incluent tous les messages CIN, la synchronisation Stop et les messages RFCIN. L'option *All* implique une recherche générique pour les processus de transaction. Lorsqu'elle est utilisée avec GTIN, cette recherche permet d'accéder à une vue plus précise des divers types de messages CIN reçus pour un GTIN donné.

Possibilité de disposer d'une fenêtre de recherche contextuelle pour le fournisseur Information dans *Manage Message > Explore Screen* :

Permet à l'utilisateur d'effectuer des recherches et de sélectionner les fournisseurs Information créés dans son système GDS au lieu d'entrer la valeur.

Flexibilité de traitement des messages CIN de types ADD, CORRECT et CHANGE, même si l'article dans GDS n'existe pas :

Possibilité de traiter les CIN de charge initiaux, même si les articles n'existent pas dans GDS. Pour plus d'informations, voir Validation de notification d'élément de catalogue

Possibilité dans l'API *HashMap MessageItemHandler::createTradeItems()* de fournir des informations plus utiles sur les cas d'échec :

Amélioration du traitement des erreurs et des exceptions pour cette API de script.

Possibilité dans les API de génération CIC *sendConfirmation()* et *sendCICConfirmationReasonCodes()* de fournir des informations plus utiles sur les cas d'échec :

Amélioration du traitement des erreurs et des exceptions pour cette API de script.

Possibilité d'envoyer des confirmations CIC de n'importe quel type depuis l'interface utilisateur GDS :

APIs de script fournies par GDS pour envoyer une confirmation CIC (Catalogue Item Confirmation) qui pouvait être utilisée par l'implémentation pour automatiser le processus. Bien souvent, les utilisateurs se plaignent de ne pas pouvoir renvoyer une confirmation CIC lorsque la confirmation précédente n'a pas été reçue par le fournisseur pour diverses raisons.

Possibilité de créer ou d'enregistrer un article depuis un message dans l'interface utilisateur GDS :

Pour permettre aux utilisateurs de mieux contrôler la gestion d'article et leurs messages CIN dans le système GDS.

Service de support logiciel IBM

Le service de support logiciel IBM vous fournit une assistance en cas de défaillance d'un produit, vous aide à répondre aux FAQ et à réaliser une nouvelle reconnaissance.

Avant de commencer

Si vous ne trouvez pas une réponse ou une solution en utilisant d'autres options d'auto-assistance tels que les notes techniques, vous pouvez contacter le support IBM. Avant de contacter le service de support IBM, vérifiez que votre entreprise possède un contrat de maintenance avec IBM et que vous êtes autorisé à soumettre des incidents à IBM. Pour plus d'informations sur les différents types de support disponibles, voir la rubrique Support portfolio dans le *Software Support Handbook*.

Lorsque vous contactez le service de support logiciel IBM, veillez à distinguer les différents niveaux de support offerts via IBM :

- Questions spécifiques au produit : si vous avez suivi une formation au produit, vous pouvez soumettre vos questions et préoccupations spécifiques.
- Support d'incident : analyse et travail nécessaires requis pour la résolution d'un défaut de produit ou soumission d'une solution palliative.
- Opérations de script : pour les questions relatives à une opération de script ou le signalement d'un probable défaut avec une opération de script, vous devez limiter votre question ou incident à environ 5 lignes de code ou moins.
 - IBM ne prend pas en charge le débogage et l'analyse des scripts personnalisés.

- Demandes d'améliorations : demandes relatives à une nouvelle fonctionnalité ajoutée au produit.

Les demandes qui ne concernent pas le service de support logiciel IBM et qui requièrent que vous contactiez votre représentant de service ou commercial :

- Travail divers : travail ou aide qui requiert une implication supplémentaire d'IBM au cours de l'installation ou du processus de mise en oeuvre qui englobe les installations complexes ou les installations à travers plusieurs plateformes.
- Support pour les clients non formés : demandes d'aide IBM étape par étape mais les clients n'ont pas suivi de formation au produit IBM. IBM Education propose des formations aux clients si du personnel supplémentaire doit être formé. Consultez le site Web IBM pour obtenir des informations spécifiques sur les cours disponibles.
- Scripts personnalisés ou rédaction de scripts : aucune prise en charge n'est fournie pour les scripts que vous avez rédigés ou qui requièrent un débogage. Tous les scripts personnalisés rédigés par les services professionnels sont détenus et gérés par vous, le client, une fois le travail des services achevé. Le support technique n'est pas disponible pour la rédaction ou le débogage des scripts personnalisés.
- Récupération des données : nous vous recommandons fortement d'effectuer régulièrement une sauvegarde de vos données.
- Aide pour les performances et le réglage ou recommandations requérant des données client.

Procédure

1. Déterminez si une note technique relative à votre incident existe déjà. Pour des informations supplémentaires sur l'affichage des notes techniques, voir Recherche dans les bases de connaissances.
2. Déterminez si un collègue ou votre équipe de support interne peut vous fournir de l'aide.
3. Définissez le problème, collectez des informations générales, puis déterminez la gravité du problème. Pour obtenir de l'aide, voir «Before contacting IBM Software Support» du document *Software Support Handbook*.

Informations d'arrière-plan requises

- Tous les paramètres et informations relatifs à la configuration.
 - Le comportement attendu par rapport au comportement obtenu.
 - Tous les changements et modifications récemment mis en oeuvre.
 - Déterminer si l'incident peut être recréé.
 - Définir les étapes qui mènent à l'apparition de l'incident.
4. Rassemblez les données de diagnostic :
 - Utilisez également la fonction de collecte de données de base de l'IBM Support Assistant pour rassembler les fichiers journaux et de configuration critiques (pour plus d'informations, voir www.ibm.com/software/support/isa/).
 - Créez votre propre outil de collecte IBM Support Assistant Lite dans l'atelier.
 - a. Téléchargez le module complémentaire du produit dans le plan de travail d'IBM Support Assistant.
 - b. Cliquez sur **Collect and Send Data**. Trois options apparaissent à gauche.
 - c. Choisissez **...Using IBM Support Assistant Lite**.

- d. Créez le collecteur généré en effectuant chacune des opérations suivantes :
 - Choisissez un collecteur (si l'add-on du produit installé ne s'affiche pas, c'est qu'il ne possède pas de document Must Gather et qu'il ne peut pas créer de collecteur IBM Support Assistant Lite)
 - Spécifiez le répertoire cible
 - Indiquez le nom du fichier
 - Cliquez sur le bouton **Export**
 - e. Transférez le collecteur IBM Support Assistant Lite exporté sur un système distant via ftp ou par le biais d'un autre protocole de transfert de fichiers disponible.
 - f. Décompressez le collecteur IBM Support Assistant Lite exporté dans l'emplacement cible.
 - g. Configurez la variable JAVA_HOME ou vérifiez que Java se trouve dans l'instruction PATH.
 - h. Exécutez la commande **runISALite.bat** sous Windows ou le script **runISALite.sh** sous UNIX et LINUX à partir du répertoire d'outils d'IBM Support Assistant Lite sur le système distant.
 - i. Apportez les réponses appropriées aux messages de collecte éventuels.
 - j. Vérifiez les fichiers collectés ou envoyez-les au service de support IBM.
5. Soumettez votre problème au service de support logiciel IBM de l'une des façons suivantes :
- En ligne : (recommandé)
 - a. Cliquez sur le lien **Demandes de service**, qui figure dans la section **Choisissez votre page** de la barre de navigation gauche du site de support d'IBM InfoSphere Master Data Management Collaboration Server.
 - b. Cliquez sur **ESR / SR** et suivez les instructions fournies sur le site ESR.
- Utilisez les instructions et conseils suivants pour utiliser l'ESR et soumettre des RGP :
- Joignez les documents directement au RGP afin de les rendre disponibles pour les services de support IBM.
 - ESR gère la liste de tous les RGP que vous avez affichés afin que vous puissiez y accéder directement afin de vérifier leur statut.
 - Utilisez l'ESR pour configurer des notifications RGP automatiques.
 - Envoyez des mises à jour pour les RGP que vous avez ouverts directement aux services de support IBM depuis l'ESR.
 - Lorsque vous soumettez un RGP, vous devez indiquer un degré de gravité afin de nous aider à définir la priorité pour nos ressources par rapport aux autres RGP en cours :

Gravité 1

Impact métier majeur : ce degré indique que vous êtes dans l'impossibilité d'utiliser le programme ce qui affecte sérieusement vos opérations. Cette situation nécessite une solution immédiate.

Remarque : L'impact métier majeur signifie généralement que votre système de production est à l'arrêt, et, pour que le service de support logiciel IBM puisse travailler en dehors des heures de bureau, vous devez vous assurer qu'un contact soit disponible par téléphone 24 heures sur 24 afin de faciliter les

recherches de l'ingénieur chargé du problème, d'exécuter les tests dans votre instance de production, etc.

Gravité 2

Impact métier important : indique que le programme est utilisable mais de façon très restreinte.

Gravité 3

Impact métier quelconque : indique que le programme est utilisable et que certaines fonctions majeures (non essentielles pour les opérations) sont indisponibles.

Gravité 4

Impact métier minime : indique un problème avec un faible impact sur les opérations ou qu'un contournement au problème a été mis en oeuvre.

- Pour ouvrir une demande d'amélioration (une demande pour une nouvelle fonctionnalité produit générale non associée à une personnalisation spécifique) :
 - Ouvrez un RGP avec le degré de gravité 2, 3 ou 4 (en fonction du gain potentiel de la demande) et la requête est envoyée à la gestion produit IBM pour révision. Vous recevrez un numéro de demande d'amélioration pour le suivi qui remplace le numéro de RGP. Généralement, votre approbation est requise pour clôturer le RGP.
- Par téléphone : pour connaître le numéro de téléphone à appeler dans votre pays, accédez à la page des contacts du *Software Support Handbook*.

Que faire ensuite

Si le problème que vous soumettez concerne un défaut de produit ou une documentation manquante ou inexacte, le service de support logiciel IBM crée un rapport officiel d'analyse de programme (APAR). L'APAR décrit l'incident en détails. Lorsque cela est possible, le service de support logiciel IBM propose une solution palliative jusqu'à la résolution de l'APAR et à la découverte d'un correctif. IBM publie les APAR résolus tous les jours sur le site Web du support IBM afin que d'autres utilisateurs qui connaissent le même problème puissent bénéficier de la même résolution.

Chapitre 1. Migration

Si vous utilisez une version antérieure d'IBM InfoSphere Master Data Management Collaboration Server, vous pouvez migrer votre version vers la version la plus récente.

Ces informations d'installation sont également disponibles dans le document InfoSphere MDM Collaboration Server Migration Guide, accessible dans le format livre dans un fichier PDF. Cliquez avec le bouton droit de la souris sur le lien, puis sélectionnez **Enregistrer la cible sous** pour télécharger le fichier PDF sur votre ordinateur.

Chapitre 2. Guide d'initiation

Vous devez modifier le schéma de base de données de l'installation actuelle en utilisant les scripts de migration pour migrer des versions antérieures vers IBM InfoSphere Master Data Management Collaboration Server Version 10.1.

Vous devez posséder les droits appropriés pour créer et modifier des tables et des index dans la base de données, de même que pour créer et supprimer des fichiers dans les sous-répertoires du répertoire \$TOP.

Le processus de migration comprend les étapes de base suivantes :

- Vérifier que la table de base de données contient un espace suffisant.
- Arrêter l'instance qui utilise le schéma de base de données.
- Sauvegarder le schéma de base de données existant.
- Installez InfoSphere MDM Collaboration Server V10.1 et pointez vers le même schéma de base de données que celui qu'utilisait l'instance précédente.
- Exécutez le script de migration approprié pour modifier le schéma de base de données et amenez-le au niveau InfoSphere MDM Collaboration Server V10.1.

Chapitre 3. Migration depuis des versions antérieures

Vous pouvez migrer vers IBM InfoSphere Master Data Management Collaboration Server Version 10.1.0 depuis des versions antérieures spécifiques.

Trois chemins de migration sont pris en charge pour la version 10.1.0 :

- Migration de V6.0.0, groupe de correctifs 12 vers V10.1.0.
- Migration de V9.1.0, groupe de correctifs 9 vers V10.1.0.
- Migration de V10.0.0, groupe de correctifs 3 vers V10.1.0.

Si vous migrez vers une version suivante, et non pas un groupe de correctifs de la même édition, vous devez utiliser un nouveau répertoire d'installation pour éviter les discordances de version de fichier JAR.

Si vous effectuez la migration à partir d'une version, assurez-vous que vous n'utilisez pas d'API supprimées dans vos scripts Excel. Les API suivantes ne sont pas disponibles dans le dernier fichier JAR POI :

- `public String getExcelCellEncoding();`
- `public void setEncoding(String encoding);`
- `public String getExcelCellEncoding(Object cellObj) throws AustinException;`

Restriction :

1. La prise en charge de InfoSphere MDM Collaboration Server Excel Parser s'effectue via un fichier JAR tiers POI-3.2-FINAL.jar. Ce fichier JAR n'a pas de correctif pour la limitation d'utilisation de plus de 244 colonnes dans une ligne.
2. Excel 2010 .xlsx n'est pas pris en charge par ce fichier JAR.

Important : Avant d'exécuter les scripts de migration, vous devez vérifier que vous utilisez la version correcte de la base de données. Voir la configuration système pour plus d'informations http://www-01.ibm.com/software/data/infosphere/mdm_server_pim/requirements.html et <http://www-01.ibm.com/support/docview.wss?uid=swg27035486> en fonction de votre version du produit. Vous devez également vérifier que l'utilisateur de la base de données InfoSphere MDM Collaboration Server bénéficie des droits appropriés, comme indiqué dans le guide d'installation. Consultez l'équipe DBA pour les modifications des privilèges utilisateur de base de données pour les règles de sécurité.

Vous devez utiliser la version de base de données DB2 ou Oracle, comme indiqué dans la configuration système. Voir la configuration système pour plus d'informations http://www-01.ibm.com/software/data/infosphere/mdm_server_pim/requirements.html et <http://www-01.ibm.com/support/docview.wss?uid=swg27035486> en fonction de votre version du produit. Vous pouvez vous reporter à la documentation DB2 ou Oracle pour savoir comment mettre à niveau votre base de données DB2 et Oracle vers la version prise en charge.

Étapes de migration de InfoSphere MDM Collaboration Server lorsque des espaces table personnalisés sont utilisés dans DB2 et Oracle

Si vous utilisez des noms d'espace table personnalisés à la place de USERS ou INDX, vous devez modifier manuellement les noms d'espace table dans les fichiers SQL suivants avant de migrer vers InfoSphere MDM Collaboration Server.

Vous devez vérifier que la base de données contient un espace suffisant avant d'effectuer la migration.

Les journaux d'audit ne sont pas migrés vers V10.1.0. Suite au nouveau format des descriptions de journal d'audit introduit dans V9.1.0, les descriptions de journal d'audit dans les éditions avant V9.1.0 ne peuvent pas être affichées dans V10.1.0. Par conséquent, elles sont supprimées de la base de données pendant la migration. Les anciens journaux d'audit peuvent être affichés en utilisant l'édition depuis laquelle vous effectuez la migration.

Vous devez effectuer manuellement les étapes suivantes pour réussir la migration :

1. Procédez à la migration manuelle des fichiers de propriétés mentionnés ci-dessous. Dans la mesure où les fichiers de propriétés font partie des systèmes de fichiers locaux, la méthode de migration ne prend pas en charge l'exportation de ces fichiers. Modifiez chaque propriété personnalisée des éditions précédentes pour utiliser le fichier correspondant dans la nouvelle version. Ne remplacez pas les fichiers de propriétés d'une ancienne version par une nouvelle version, car de nouveaux paramètres ont été introduits dans V10.1.0. Pour plus d'informations sur chacun des fichiers de propriétés, consultez la documentation InfoSphere MDM Collaboration Server.
 - \$TOP/etc/default/common.properties
 - \$TOP/etc/default/logs.xml
 - \$TOP/etc/default/docstore_mount.xml
 - \$TOP/etc/default/user_jars.list
2. Copiez manuellement le contenu du répertoire \$TOP/public_html/suppliers/<cmp code>/ctg_files vers la nouvelle instance pour chaque société. Dans la mesure où ces fichiers font partie du système de fichiers local, la méthode de migration ne prend pas en charge l'exportation du répertoire ctg_files.
3. Ajoutez manuellement les fichiers JAR utilisateur nécessaires à la nouvelle instance. Comme ces fichiers font partie du système de fichiers local, la méthode de migration ne permet pas d'exporter les fichiers. Pour plus d'informations sur l'ajout des fichiers JAR au déploiement, voir Déploiement d'un fichier .jar utilisateur personnalisé.
4. Rétablissez toutes les modifications que vous aviez apportées au script Default Rich Search Results Report pour le personnaliser. La migration vers InfoSphere MDM Collaboration Server V10.1.0 remplace le script existant dans le répertoire de magasin de documents InfoSphere MDM Collaboration Server, /scripts/report. Ce script permet d'exporter les résultats de recherche d'objet InfoSphere MDM Collaboration Server vers une feuille de calcul Excel. Pour plus d'informations sur ce script, consultez la rubrique sur l'exportation des résultats de la recherche vers Excel.

Migration des données maître du format interne vers le format XML

Vous devez convertir les données maître dans le format XML lors de la migration depuis des versions antérieures à IBM InfoSphere Master Data Management Collaboration Server Version 10.1.0.

Avant de commencer

1. Créez des espaces table XML et des pools de mémoire tampon, comme indiqué dans le guide d'installation.
2. Vous devez également vérifier que l'utilisateur de la base de données de InfoSphere MDM Collaboration Server dispose des droits appropriés, comme indiqué dans le guide d'installation.
3. Arrêtez l'application InfoSphere MDM Collaboration Server sur le serveur local.
4. Exécutez le script `delete_old_versions.sh` pour supprimer les anciennes versions. Utilisez le code de la société et la date en cours comme valeurs pour les paramètres **code** et **end_date**.
5. Facultatif : étendez la mémoire disponible pour le processus Java. L'exécution du script de conversion des données utilise beaucoup de mémoire. Il est recommandé de fournir 2 Go de mémoire au processus Java en mettant à jour le paramètre **custom_java_options** dans le fichier `$TOP/bin/conf/env_settings.ini`. Exécutez `configureEnv.sh` après avoir modifié le paramètre **custom_java_options** pour appliquer les nouveaux paramètres de mémoire.
`custom_java_options=-Xmx2048m -Xms256m`

Pourquoi et quand exécuter cette tâche

Le script de migration des données nécessite en terme de connexions le double du nombre d'unités d'exécution. Par exemple, si vous utilisez 10 unités d'exécution dans l'option de ligne de commande, 20 connexions BD sont nécessaires. Vous devez configurer le nombre de connexions BD en utilisant les deux paramètres de configuration suivants :

- `db_maxConnection`
- `db_maxConnection_default`

Les deux paramètres sont disponibles dans le fichier `common.properties`. Si les valeurs des paramètres sont déjà égales ou supérieures au nombre de connexions BD nécessaires, vous n'avez pas à les modifier. Si elles sont inférieures au nombre de connexion BD, augmentez les valeurs pour améliorer la vitesse.

Si les catalogues contiennent plus de 100 000 entrées de données, augmentez le nombre d'unité d'exécution. Par exemple, vous pouvez utiliser 50 ou 100 unités d'exécution pour accélérer la migration des données. Le nombre d'unités d'exécution maximal que vous pouvez utiliser dépend des variables suivantes dans InfoSphere MDM Collaboration Server. Vous pouvez connaître ce nombre en exécutant le script de migration dans un environnement de test.

- Nombre de coeurs de processeurs disponibles
- Nombre de specs définies
- Mémoire disponible

Toutes les specs définies dans l'environnement InfoSphere MDM Collaboration Server sont extraites et utilisées dans le processus de conversion des données. Si vous disposez de plus de 100 specs, modifiez les paramètres de mise en cache des specs en leur affectant les valeurs appropriées pour éviter les lectures fréquentes de base de données et améliorer les performances générales du script.

Procédure

1. Obtenez le nombre de specs dans InfoSphere MDM Collaboration Server en utilisant la commande suivante.
`$PERL5LIB/runSQL.pl --sql_command="select count(*) from spc;"`

2. Modifiez les paramètres avec les valeurs extraites de la requête de l'étape 1. Ces paramètres se trouvent dans le fichier `mdm-cache-config.properties` dans le répertoire `$TOP/etc/default`.

```
specCache_KEY_START_VERSION_TO_VALUE.maxElementsInMemory=<spec count>
```

```
specCache_KEY_TO_CURRENT_START_VERSION.maxElementsInMemory=<spec count>
```

```
specCache_KEY_VERSION_TO_START_VERSION.maxElementsInMemory=<spec count>
```

Remarque : L'exécution du script `configureEnv.sh` après la mise à jour du fichier `mdm-cache-config.properties` remplace les paramètres par les valeurs par défaut.

3. Exécutez le script de shell de migration des données `migrateDataToXml.sh`. Ce fichier se trouve dans le répertoire `$TOP/bin/migration`.

```
migrateDataToXml.sh --company=<company name> --threads=<number of threads>
```

, où

--company

Définit le code de société utilisé pour se connecter à InfoSphere MDM Collaboration Server.

--threads

Définit le nombre d'unités d'exécution. Utilisez un grand nombre d'unités d'exécution pour améliorer la vitesse, mais vérifiez qu'un nombre suffisant de connexions DB existent. La valeur par défaut est 1. Au moins 10 unités d'exécution sont recommandées.

Par exemple,

```
migrateDataToXml.sh --company=ibm --threads=50
```

Que faire ensuite

Vous pouvez vérifier que le script fonctionne correctement en observant les erreurs qui s'affichent sur la console et qui sont consignées dans le fichier `default.log` dans le répertoire `$TOP/logs/default`. La liste des entrées qui n'ont pas pu être migrées s'affichent à la fin de l'exécution.

Le script de migration de données met à jour la base de données par lot. En cas d'erreur, l'ensemble du lot est annulé. Par conséquent, le nombre d'entrées dont la migration échoue est supérieur au nombre réel d'échecs. Affectez la valeur 1 au paramètre **index_regeneration_batch_size** dans le fichier `common.properties`, puis réexécutez le script de migration de données pour réduire le nombre d'entrées par rapport aux échecs réels. Le tableau des causes possibles des erreurs de migration et les actions recommandées sont fournis :

Tableau 1. Causes possibles des erreurs et actions recommandées

Causes des erreurs de migration	Action recommandée
La migration des données affiche l'erreur suivante : java.sql.SQLException: ORA-00600: internal error code, arguments: [KGHALP1], [0x00000000], [], [], [], [], [], [], [], [], [], []	Vérifiez que vous utilisez le niveau de correctif de base de données correct. Si nécessaire, mettez à jour la base de données, puis réexécutez le script de migration des données.

Tableau 1. Causes possibles des erreurs et actions recommandées (suite)

Causes des erreurs de migration	Action recommandée
La migration des données affiche l'erreur suivante : <pre>java.sql.SQLException: ORA-31061: XDB error: XML event error ORA-19202: Error occurred in XML processing In line 46 of orastream: LPX-00241: entity reference is not well formed</pre>	L'une des valeurs d'attribut dans l'entrée contient le caractère "&" ou la valeur n'est pas valide. Modifiez l'entrée pour supprimer la valeur non valide depuis l'interface d'édition d'élément de InfoSphere MDM Collaboration Server et réexécutez le script de migration des données.

Migration de V6.0.0 vers V10.1.0

Pour migrer vers IBM InfoSphere Master Data Management Collaboration Server Version 10.1.0 depuis IBM InfoSphere Master Data Management Server for Product Information Management Version 6.0.0, vous devez exécuter le script de migration approprié.

Avant de commencer

Avant de migrer de InfoSphere MDM Server for PIM V6.0.0 vers InfoSphere MDM Collaboration Server V10.1.0, vous devez vérifier que vous utilisez le groupe de correctifs 12 de V6.0.0. Si vous utilisez V6.0.0 avec n'importe quel niveau de groupe de correctifs de 1 à 11, vous devez migrer vers le groupe de correctifs 12 V6.0.0 en suivant les instructions dans la documentation IBM InfoSphere Master Data Management Server for Product Information Management Version 6.0.0, Fix Pack 12.

La configuration doit avoir les caractéristiques suivantes :

1. Tous les espaces table XML doivent être créés comme indiqué dans le guide d'installation et l'utilisateur de la base de données MDM Collaboration Server doit être autorisé à utiliser les nouveaux espaces table.
2. Les espaces table doivent contenir un espace libre suffisant.

Si vous utilisez des noms d'espace table personnalisés à la place des noms par défaut USERS ou INDX, vous devez modifier manuellement les noms d'espace table dans les fichiers SQL suivant avant de migrer vers InfoSphere MDM Collaboration Server.

Tableau 2. Fichiers SQL à modifier lors de l'utilisation de DB2 ou d'Oracle comme base de données dorsale

Fichiers SQL à modifier quand DB2 est la base de données d'arrière plan	Fichiers SQL à modifier quand Oracle est la base de données d'arrière plan
Remplacez les noms d'espace table USERS, INDX et BLOB_TBL_DATA par les noms d'espace table personnalisés appropriés dans les fichiers SQL suivants dans le répertoire \$TOP/src/db/schema/dbscripts/db2 :	Remplacez les noms d'espace table USERS, INDX et BLOB_TBL_DATA par les noms d'espace table personnalisés appropriés dans les fichiers SQL suivants dans le répertoire \$TOP/src/db/schema/dbscripts/oracle :
create_new_tables_of_650.sql create_new_tables_of_1000.sql create_tables_of_gds.sql recreate_coa_for_edit_flag.sql migrateItld.sql migrateDb1.sql migrate0bh.sql migrateSst.sql migrateCad.sql	create_new_tables_of_650.sql create_new_tables_of_1000.sql create_tables_of_gds.sql recreate_coa_for_edit_flag.sql migrateItld.sql migrateDb1.sql migrate0bh.sql migrateSst.sql migrateCad.sql

Remarques concernant les bases de données DB2

Si vous effectuez une mise à niveau depuis une ancienne version de DB2 vers la version prise en charge, vous devez vous assurer que le paramètre de configuration de la base de données **CUR_COMMIT** est activé. Vous devez définir manuellement ce paramètre ; en effet, il est par défaut désactivé pour les bases de données mises à niveau depuis des versions plus anciennes.

L'administrateur de base de données doit également exécuter les commandes SQL suivantes avant de migrer InfoSphere MDM Collaboration Server.

```
alter tablespace blob_tbl_data convert to large;
alter tablespace blob_tbl_data file system caching
```

Vous pouvez configurer l'emplacement du fichier journal des erreurs en modifiant la variable **_ERROR_LOGS** dans le script de migration `migrateFrom600FP.sh`. Cette variable est définie sur `$TOP/logs/errfile.log` par défaut.

Pourquoi et quand exécuter cette tâche

Il s'agit de la seule et unique méthode possible pour migrer depuis la version 6.0.0 d'InfoSphere MDM Server for PIM. Aucun autre moyen, comme utiliser l'outil d'exportation et d'importation d'environnement, ne peut être utilisé pour cette migration.

Procédure

1. Effectuez une copie de sauvegarde du schéma de base de données existant.
2. Installez InfoSphere MDM Collaboration Server V10.1.0 dans un répertoire différent.
3. Configurez la base de données.
 - a. Pour les bases de données DB2, créez des pools de mémoire tampon et des espaces table, comme indiqué dans Configuration de la base de données DB2 dans le guide d'installation.

Remarque : La taille de l'espace table XML_DATA doit être égale à l'espace occupé par l'espace table TCTG_ITD_ITEM_DETAIL. Pour connaître la taille approximative de la table, utilisez la requête suivante :

```
select (fpages * 16) / 1024 megabytes from syscat.tables
where tabschema = current schema
and tabname like 'TCTG_ITD_ITEM_DETAIL';
```

- b. Pour les bases de données Oracle, créez des espaces table et installez le composant Oracle XML DB, comme indiqué dans Configuration de la base de données Oracle dans la guide d'installation.
4. Pointez l'instance InfoSphere MDM Collaboration Server V10.1.0 vers le schéma de base de données V6.0.0, groupe de correctifs 12.
5. Exécutez le script de migration migrateFrom600FP.sh. Ce script est disponible dans le répertoire \$TOP/bin/migration.
6. Migrez les données, comme indiqué dans Conversion des données maître du format interne dans le format XML.
7. Démarrez l'instance InfoSphere MDM Collaboration Server V10.1.0.

Résultats

Tous les messages, comme l'état des composants de migration individuels et le récapitulatif de migration global, sont affichés dans la console.

Après avoir exécuté le script de migration, des messages indiquant que la migration a réussi ou échoué s'affichent.

Le message suivant s'affiche si la migration a réussi :

```
-----
Summary of the migration
-----
Migration to IBM InfoSphere Master Data Management
Collaboration Server Version 10.1.0 is complete. Check the
messages for any errors.
```

Generating database verification report...

Changed tables

```
=====
There are no changed tables
```

Missing Tables

```
=====
There are no missing tables
```

Changed Indexes

```
=====
There are no changed indexes
```

Missing Indexes

=====
There are no missing indexes

LOG FILE: \$TOP/logs/default/default.log

Le message suivant s'affiche si la migration a échoué :

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Migration Failed : xxxx
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Please see the log file for further details :
/home/pimuser/mdmcs10/logs/errfile.log

Le message suivant s'affiche si la migration de modules spécifiques a échoué :

Summary of the migration

Migration of the following modules failed :
<module names>

Ce message s'affiche s'il existe une différence dans les tables et les index de base de données.

Generating database verification report...

Changed tables

=====
<table names>

Missing Tables

=====
<table names>

Changed Indexes

=====
<index names>

Missing Indexes

=====

<index names>

LOG FILE: \$TOP/logs/default/default.log

Pour connaître les autres messages, reportez-vous au fichier `errfile.log`. Pour les erreurs SQL, trouvez le message d'erreur détaillé à partir du code d'erreur SQL, corrigez l'erreur, puis exécutez de nouveau le script de migration. Pour l'action recommandée en cas d'échec de la migration, voir Identification et résolution des problèmes de migration. Contactez le service de support IBM si le problème persiste après avoir corrigé les erreurs.

Migration de V9.1.0 vers V10.1.0

Pour migrer vers IBM InfoSphere Master Data Management Collaboration Server Version 10.1.0 depuis IBM InfoSphere Master Data Management Server for Product Information Management Version 9.1.0, vous devez exécuter le script de migration approprié.

Avant de commencer

Avant de migrer d'InfoSphere MDM Server for PIM V9.1.0 vers InfoSphere MDM Collaboration Server V10.1.0, vous devez vérifier que vous utilisez le groupe de correctifs 9 de V9.1.0. Si vous utilisez V9.1.0 avec n'importe quel niveau de groupe de correctifs de 1 à 3, vous devez migrer vers le groupe de correctifs 9 V9.1.0 en suivant les instructions dans la documentation IBM InfoSphere Master Data Management Server for Product Information Management Version 9.1.0, Fix Pack 9.

Vous devez également vérifier que tous les espaces table XML sont créés comme indiqué dans le guide d'installation et que l'utilisateur de la base de données InfoSphere MDM Collaboration Server est autorisé à utiliser les nouveaux espaces table.

Vous pouvez configurer l'emplacement du fichier journal des erreurs en modifiant la variable `_ERROR_LOGS` dans le script de migration `migrateFrom910.sh`. Cette variable est définie sur `$TOP/logs/errfile.log` par défaut.

Pourquoi et quand exécuter cette tâche

Il s'agit de la seule et unique méthode possible pour migrer d'InfoSphere MDM Server for PIM V9.1.0 vers InfoSphere MDM Collaboration Server V10.1.0. Aucune autre méthode, notamment l'utilisation de l'outil d'exportation et d'importation, ne peut être utilisée pour cette migration.

Procédure

1. Effectuez une copie de sauvegarde du schéma de base de données existant.
2. Installez InfoSphere MDM Collaboration Server V10.1.0 dans un répertoire différent.
3. Configurez la base de données.
 - a. Pour les bases de données DB2, créez des pools de mémoire tampon et des espaces table, comme indiqué dans Configuration de la base de données DB2 dans le guide d'installation.

Remarque : La taille de l'espace table XML_DATA doit être égale à l'espace occupé par l'espace table TCTG_ITD_ITEM_DETAIL. Pour connaître la taille approximative de la table, utilisez la requête suivante :

```
select (fpages * 16) / 1024 megabytes from syscat.tables
where tabschema = current schema
and tabname like 'TCTG_ITD_ITEM_DETAIL';
```

- b. Pour les bases de données Oracle, créez des espaces table et installez le composant Oracle XML DB, comme indiqué dans Configuration de la base de données Oracle dans la guide d'installation.
4. Pointez l'instance InfoSphere MDM Collaboration Server V10.1.0 vers le schéma de base de données V9.1.0, groupe de correctifs 9.
5. Exécutez le script de migration migrateFrom910FP.sh. Ce script est disponible dans le répertoire \$TOP/bin/migration.
6. Migrez les données, comme indiqué dans Conversion des données maître du format interne dans le format XML.
7. Démarrez l'instance InfoSphere MDM Collaboration Server V10.1.0.

Résultats

Tous les messages, comme l'état des composants de migration individuels et le récapitulatif de migration global, sont affichés dans la console.

Après avoir exécuté le script de migration, des messages indiquant que la migration a réussi ou échoué s'affichent.

Le message suivant s'affiche si la migration a réussi :

```
-----
Summary of the migration
-----
Migration to IBM InfoSphere Master Data Management
Collaboration Server Version 10.1.0 is complete. Check the
messages for any errors.
```

Generating database verification report...

Changed tables

```
=====
There are no changed tables
```

Missing Tables

```
=====
There are no missing tables
```

Changed Indexes

```
=====
There are no changed indexes
```

Missing Indexes

=====
There are no missing indexes

LOG FILE: \$TOP/logs/default/default.log

Le message suivant s'affiche si la migration a échoué :

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Migration Failed : xxxx
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Please see the file for further details : /home/pimuser/mdmcs10/logs/errfile.log

Le message suivant s'affiche si la migration de modules spécifiques a échoué :

Summary of the migration

Migration of the following modules failed :
<module names>

Ce message s'affiche s'il existe une différence dans les tables et les index de base de données.

Generating database verification report...

Changed tables

=====
<table names>

Missing Tables

=====
<table names>

Changed Indexes

=====
<index names>

Missing Indexes

=====
<index names>

LOG FILE: \$TOP/logs/default/default.log

Pour connaître les autres messages, reportez-vous au fichier `errfile.log`. Pour les erreurs SQL, trouvez le message d'erreur détaillé à partir du code d'erreur SQL, corrigez l'erreur, puis exécutez de nouveau le script de migration. Pour l'action recommandée en cas d'échec de la migration, voir Identification et résolution des problèmes de migration. Contactez IBM Software Support si le problème persiste après avoir corrigé les erreurs.

Migration de V10.0.0 vers V10.1.0

Pour migrer vers IBM InfoSphere Master Data Management Collaboration Server Version 10.1.0 depuis IBM InfoSphere Master Data Management Server for Product Information Management Version 10.0.0, vous devez exécuter le script de migration approprié.

Avant de commencer

Avant de migrer de InfoSphere MDM Collaboration Server V10.0.0 vers V10.1.0, vous devez vérifier que vous utilisez le groupe de correctifs 3 de V10.0.0. Si vous utilisez le groupe de correctifs V10.0.0 1 ou 2, vous devez migrer vers le groupe de correctifs 3 V10.0.0 en suivant les instructions dans la documentation IBM InfoSphere Master Data Management Collaboration Server Version 10.0.0, Fix Pack 3.

Vous pouvez configurer le fichier journal des erreurs en modifiant la variable `_ERROR_LOGS` dans le script de migration `migrateFrom910.sh`. Cette variable est définie sur `$TOP/logs/errfile.log` par défaut.

Pourquoi et quand exécuter cette tâche

Il s'agit de la seule et unique méthode possible pour migrer de InfoSphere MDM Collaboration Server V10.0.0 vers V10.1.0. Aucune autre méthode, notamment l'utilisation de l'outil d'exportation et importation, ne peut être utilisée pour cette migration.

Procédure

1. Effectuez une copie de sauvegarde du schéma de base de données existant.
2. Installez InfoSphere MDM Collaboration Server V10.1.0 dans un répertoire différent.
3. Pointez l'instance InfoSphere MDM Collaboration Server V10.1.0 vers le schéma de base de données V10.0.0, groupe de correctifs 3.
4. Exécutez le script de migration `migrateFrom1000FP.sh`. Ce script est disponible dans le répertoire `$TOP/bin/migration`.
5. Démarrez l'instance InfoSphere MDM Collaboration Server V10.1.0.

Résultats

Tous les messages, comme l'état des composants de migration individuels et le récapitulatif de migration global, sont affichés dans la console.

Après avoir exécuté le script de migration, des messages indiquant que la migration a réussi ou échoué s'affichent.

Le message suivant s'affiche si la migration a réussi :

Summary of the migration

Migration to IBM InfoSphere Master Data Management
Collaboration Server Version 10.1.0 is complete. Check the
messages for any errors.

Generating database verification report...

Changed tables

There are no changed tables

Missing Tables

There are no missing tables

Changed Indexes

There are no changed indexes

Missing Indexes

There are no missing indexes

LOG FILE: \$TOP/logs/default/default.log

Le message suivant s'affiche si la migration a échoué :

!!

Migration Failed : xxxx

!!

Please see the file for further details : /home/pimuser/mdmcs10/logs/errfile.log

Le message suivant s'affiche si la migration de modules spécifiques a échoué :

Summary of the migration

Migration of the following modules failed :

<module names>

Ce message s'affiche s'il existe une différence dans les tables et les index de base de
données.

Generating database verification report...

Changed tables

```
=====
<table names>
```

Missing Tables

```
=====
<table names>
```

Changed Indexes

```
=====
<index names>
```

Missing Indexes

```
=====
<index names>
```

LOG FILE: \$TOP/logs/default/default.log

Pour connaître les autres messages, reportez-vous au fichier `errfile.log`. Pour les erreurs SQL, trouvez le message d'erreur détaillé à partir du code d'erreur SQL, corrigez l'erreur, puis exécutez de nouveau le script de migration. Pour l'action recommandée en cas d'échec de la migration, voir Identification et résolution des problèmes de migration. Contactez IBM Software Support si le problème persiste après avoir corrigé les erreurs.

Migration de Global Data Synchronization de V1.3.x vers InfoSphere MDM Collaboration Server V10.1

Vous pouvez migrer vers IBM InfoSphere Master Data Management Collaboration Server V10.1 en utilisant des scripts qui automatisent la plupart des étapes de la migration. Pour que la migration aboutisse, vous devez exécuter quelques tâches manuelles en utilisant l'interface utilisateur InfoSphere MDM Collaboration Server.

Pourquoi et quand exécuter cette tâche

Vérifiez que vous utilisez l'édition de maintenance de base GDS 1.3.1-IF011 pour les implémentations Demande et Offre avant d'effectuer la migration GDS.

Remarque : Dans le cadre de cette rubrique et de ses sous-rubriques, GDS 1.3.x s'appelle GDS 1.3.1-IF011.

Procédure

1. **Exportation** : les métadonnées (modèle de données) et les données sont exportées de l'environnement GDS V1.3.x WebSphere Product Center V5.2.1.4 dorsal en utilisant les scripts d'exportation OOTB.
 - a. **Métadonnées** : les définitions de métadonnées sont exportées sous forme de fichiers compressés qui contiennent les fichiers suivants :
 - 1) **Fichiers de définition** : Ces fichiers XML sont les définitions réelles des différents objets (métadonnées) exportés.
 - 2) **Fichier de contrôle** : Fichier XML qui contrôle l'importation et gère les références du contenu du fichier compressé. Ce fichier se nomme `ImportEnvControl.xml`.
 - 3) **Fichiers de données** : Fichiers contenant des informations sur les articles et les catégories dans un format CSV (Comma Separated Value).

Le contenu du magasin de données propre à GDS, qui forme les métadonnées des objets GDS, sont exportées sous la forme de fichiers sur le système de fichiers.
 - b. **Données** : Les données d'InfoSphere MDM Collaboration Server peuvent être un article ou une catégorie. L'article et les catégories créés dans GDS sont exportés au format CSV et sauvegardés dans le système de fichiers. Les notifications et les transactions GDS sont exportées dans des fichiers compressés distincts qui possèdent le format du fichier compressé des métadonnées.
2. **Transformation** : Le script d'importation des données de migration comporte une étape qui permet de transformer les fichiers compressés de métadonnées exportés.

Remarque : Si les métadonnées WebSphere Product Center V5.2.1.4 sont exportées en utilisant des outils d'exportation non fournis par GDS, vous devez transformer les fichiers compressés générés en utilisant les scripts de migration disponibles dans InfoSphere MDM Collaboration Server V10.1 en traitant les fichiers compressés via l'utilitaire de transformation dans l'instance InfoSphere MDM Collaboration Server V10.1, `<MDMPIM_Install_Dir>/bin/migrateCompanyZip.sh`, qui génère des fichiers compressés compatibles InfoSphere MDM Collaboration Server V10.1.

3. **Importation** : les fichiers compressés des métadonnées transformées et les données exportées sont importées vers l'environnement InfoSphere MDM Collaboration Server V10.1 en utilisant les scripts d'importation inclus.

Préparation à la migration de GDS 1.3.x

Vous devez exécuter ces tâches préparatoires pour effectuer la migration de GDS 1.3.x vers IBM InfoSphere Master Data Management Collaboration Server V10.1.x.

Nettoyage des données GDS

Vous devez nettoyer l'instance source de WebSphere Product Center version 5.2.1.4 pour supprimer les éléments inutiles, notamment dans le magasin de documents. La suppression de ces éléments permet de réduire la durée et les ressources requises de la procédure de migration.

Avant de commencer

Avant de nettoyer les données GDS, vous devez effectuer les opérations suivantes :

- Effectuez une copie de sauvegarde de la base de données. Si des erreurs se produisent pendant la procédure de nettoyage, vous pouvez être amené à restaurer la base de données d'origine.
- Vérifiez que l'instance de GDS version 1.3.x est stable. Si c'est le cas, tous les messages de la file d'attente entrante ont été traités par GDS et les actions ou les travaux planifiés pour l'envoi des messages aux files d'attente sortantes sont terminés.
- Vérifiez que GDS n'est pas connecté à IBM WebSphere MQ et que les services du programme d'écoute GDS ne sont pas en cours d'exécution. Pour éviter l'envoi ou la réception de messages, arrêtez toutes les connexions à la file d'attente.
- Vérifiez que les variables d'environnement de GDS version 1.3.x font référence aux répertoires appropriés.
 - La variable d'environnement \$STOP fait référence au répertoire d'installation de GDS lorsqu'elle est définie à partir du répertoire d'installation de GDS, et au répertoire d'installation WebSphere Product Center lorsqu'elle est définie à partir du répertoire d'installation de WebSphere Product Center.
 - La variable d'environnement WPC_INSTALL_DIR fait référence au répertoire d'installation de WebSphere Product Center version 5.2.1.4.
- Adressez-vous à l'administrateur de la base de données pour vérifier que vous disposez de suffisamment d'espace pour des objets, tels que les espaces table, les objets INDEX, USERS et BLOB de la base de données.
- Adressez-vous à l'administrateur de la base de données pour vérifier que suffisamment d'espace est alloué aux journaux des transactions de la base de données afin de prendre en charge les transactions volumineuses.
- Vérifiez que le fichier gds.properties du répertoire <GDS1.3.x_Install_Dir>/etc/default est configuré correctement. Le fichier doit contenir des valeurs appropriées pour les variables *company_code*, *gds_app_type* et *ACTIVE_DATA_POOL_ID*.
- Vérifiez que le fichier gds.properties du répertoire <GDS1.3.x_Install_Dir>/etc/default est configuré correctement. Le fichier doit contenir des valeurs appropriées pour les variables *db_userName*, *db_password* et *db_url* et pour les paramètres de messagerie.
- Vérifiez que le fichier common.properties du répertoire <répertoire_installation_WPC5.2.1.4>/etc/default est configuré correctement. Le fichier doit contenir des valeurs appropriées pour les variables *db_userName*, *db_password* et *db_url* et pour les paramètres de messagerie.

Procédure

1. Supprimez tous les fichiers inutiles du magasin de documents. Par exemple, supprimez les anciens journaux, les sources et d'autres fichiers générés dont vous n'avez plus besoin.
2. Supprimez les travaux et les calendriers associés dont vous n'avez plus besoin.
3. Retirez tous les articles et les catégories réservés dans les zones de collaboration. Toutes les zones de collaboration doivent être vides.
4. Supprimez tous les flux de travaux GDS et les zones de collaboration liés à ces flux de travaux.
5. Supprimez toutes les notifications GDS créées pour les articles réservés dans les flux de travaux GDS en exécutant des requêtes SQL. Vous pouvez exécuter ces requêtes SQL depuis la console WebSphere Product Center V5.2.1.4 DB Admin, un client de base de données ou vous connecter à un serveur de base de données et exécuter les requêtes. Dans toutes ces requêtes, vous devez

remplacer <nom_société> par le nom réel de la société tel qu'il est défini par le paramètre *company_code* dans le fichier gds.properties. Ce fichier est disponible dans le répertoire <répertoire_installation_GDS>/etc/default.

Remarque : Vous devez remplacer <company_name> par le nom réel de la société tel qu'il est défini par le paramètre **company_code** dans le fichier gds.properties.

- a. Exécutez les requêtes SQL suivantes pour supprimer les notifications liées aux articles réservés dans les étapes des flux de travaux GDS.

Requêtes SQL côté demande

```
delete from rcp
where RCP_COMPANY_ID = (
    select CMP_COMPANY_ID from cmp
    where CMP_COMPANY_NAME='<company_name>')
and RCP_NTF_ID in (
    select NTF_ID from ntf
    where NTF_ACTION_CAUSE in (
        'SUBSCRIPTION_DELETION_APPROVAL',
        'SUBSCRIPTION_PENDING_APPROVAL',
        'REVIEW_INCORRECT_INITIAL_LOAD_PUBLICATION',
        'REVIEW_INCORRECT_ITEM_CHANGE_PUBLICATION',
        'REVIEW_INCORRECT_ITEM_CORRECT_PUBLICATION',
        'ITEM_DELETE_PUBLICATION_VALIDATION_FAILED',
        'REVIEW_INCORRECT_NEW_ITEM_PUBLICATION',
        'NEW_ITEM_CATEGORY_MAPPING_PENDING')
    )

delete from ntf
where NTF_COMPANY_ID = (
    select CMP_COMPANY_ID from cmp
    where CMP_COMPANY_NAME='<company_name>')
and NTF_ACTION_CAUSE in (
    'SUBSCRIPTION_DELETION_APPROVAL',
    'SUBSCRIPTION_PENDING_APPROVAL',
    'REVIEW_INCORRECT_INITIAL_LOAD_PUBLICATION',
    'REVIEW_INCORRECT_ITEM_CHANGE_PUBLICATION',
    'REVIEW_INCORRECT_ITEM_CORRECT_PUBLICATION',
    'ITEM_DELETE_PUBLICATION_VALIDATION_FAILED',
    'REVIEW_INCORRECT_NEW_ITEM_PUBLICATION',
    'NEW_ITEM_CATEGORY_MAPPING_PENDING')
```

Requêtes SQL côté fournisseur

```
delete from rcp
where RCP_COMPANY_ID = (
    select CMP_COMPANY_ID from cmp
    where CMP_COMPANY_NAME='<company_name>')
and RCP_NTF_ID in (
    select NTF_ID from ntf
    where NTF_ACTION_CAUSE in (
        'EDIT_ITEM_APPROVAL_PENDING',
        'ITEM_GLOBAL_ENRICHMENT_PENDING',
        'ITEM_LOCAL_ENRICHMENT_PENDING',
        'MODIFY_GLOBAL_ITEM_APPROVAL_PENDING',
        'MODIFY_ITEM_APPROVAL_PENDING',
        'MODIFY_ITEM_LINK_APPROVAL_PENDING',
        'MODIFY_TRADING_PARTNER_PENDING_APPROVAL',
        'NEW_ITEM_APPROVAL_PENDING',
        'NEW_ITEM_PENDING_PUBLICATIONS',
        'NEW_TRADING_PARTNER_PENDING_APPROVAL',
        'TRADING_PARTNER_DELETION_APPROVAL')
    )

delete from ntf
where NTF_COMPANY_ID = (
```

```

select CMP_COMPANY_ID from cmp
where CMP_COMPANY_NAME='<company_name>')
and NTF_ACTION_CAUSE in (
'EDIT_ITEM_APPROVAL_PENDING',
'ITEM_GLOBAL_ENRICHMENT_PENDING',
'ITEM_LOCAL_ENRICHMENT_PENDING',
'MODIFY_GLOBAL_ITEM_APPROVAL_PENDING',
'MODIFY_ITEM_APPROVAL_PENDING',
'MODIFY_ITEM_LINK_APPROVAL_PENDING',
'MODIFY_TRADING_PARTNER_PENDING_APPROVAL',
'NEW_ITEM_APPROVAL_PENDING',
'NEW_ITEM_PENDING_PUBLICATIONS',
'NEW_TRADING_PARTNER_PENDING_APPROVAL',
'TRADING_PARTNER_DELETION_APPROVAL')
)

```

- b. Exécutez les requêtes SQL suivantes pour supprimer les entrées de notification créées pour l'affichage utilisateur.

```

delete from rcp
where RCP_COMPANY_ID = (
select CMP_COMPANY_ID from cmp
where CMP_COMPANY_NAME='<company_name>')
and RCP_NTF_ID in (
select NTF_ID from ntf where NTF_SOURCE_USER_ID = -1
)

```

```

delete from ntf where NTF_COMPANY_ID = (
select CMP_COMPANY_ID from cmp
where CMP_COMPANY_NAME='<company_name>')
and NTF_SOURCE_USER_ID = -1

```

- c. Exécutez les requêtes SQL suivantes pour supprimer les entrées de notification qui n'ont pas de destinataires.

```

delete from rcp
where RCP_COMPANY_ID = (
select CMP_COMPANY_ID from cmp
where CMP_COMPANY_NAME='<company_name>')
and RCP_NTF_ID not in (select NTF_ID from ntf)

```

```

delete from ntf
where NTF_COMPANY_ID = (
select CMP_COMPANY_ID from cmp
where CMP_COMPANY_NAME='<company_name>')
and NTF_ID not in (select RCP_NTF_ID from rcp)

```

où *<company_name>* doit être remplacé par le nom réel de la société tel qu'il est défini par le paramètre **company_code** du fichier `<répertoire_installation_GDS>/etc/default/gds.properties`.

Suppression des scripts de mise en oeuvre associés aux hiérarchies et aux catalogues

Avant d'exporter des données, vous devez retirer manuellement les scripts de mise en oeuvre associés à une hiérarchie ou à un catalogue. Vous pouvez rattacher les scripts après l'exportation réussie des hiérarchies et des catalogues.

Ajout d'attributs SRT

Si votre implémentation comporte des attributs qui doivent être de type RTS (indexé) dans l'environnement cible IBM InfoSphere Master Data Management Collaboration Server version 9.1.x, vous devez les ajouter au fichier `Demand_Params.xml` (côté demande) ou au fichier `Supply_Params.xml` (côté fournisseur). Ces fichiers, qui contiennent tous les attributs susceptibles de faire l'objet d'une recherche et fournis dans le modèle de données GDS, se trouve dans le répertoire `$TOP/GDSMigration`.

Pourquoi et quand exécuter cette tâche

Le fichier `pkAndDspIndexedAttr.xml` est généré dans le répertoire `$TOP/GDSMigration/envInfo` pendant l'étape `FirstSteps` de la procédure d'exportation. Ce fichier contient tous les attributs de clé primaire et d'affichage indexés par défaut. Vous devez ajouter ces attributs manuellement au fichier `Demand_Params.xml` (côté demande) ou au fichier `Supply_Params.xml` (côté fournisseur) dans le répertoire `$TOP/GDSMigration`.

Procédure

1. Ouvrez le fichier `Demand_Params.xml` (côté demande) ou le fichier `Supply_Params.xml` (côté fournisseur) dans un éditeur de texte.
2. Ajoutez les balises `spec` et `attribute`. Par exemple, pour indexer les attributs `sku` et `productDetails` de la spécification `Master_Spec`, ajoutez les éléments suivants au fichier `Demand_Params.xml` ou au fichier `Supply_Params.xml`.

```
<Spec name="Master_Spec">
<Attribute>sku</Attribute>
<Attribute>productDetails/productName</Attribute>
</Spec>
```

Le fichier `Demand_Params.xml` ou le fichier `Supply_Params.xml` obtenu prend la forme suivante :

```
<?xml version="1.0" encoding="UTF-8"?>
<IndexableAttributes>
...
...
...
<Spec name=" Master_Spec ">
  <Attribute>sku</Attribute>
  <Attribute>productDetails/displayAttribute</Attribute>
</Spec>
...
...
...
</IndexableAttributes>
```

3. Enregistrez le fichier.

Mise à jour du fichier `gds.properties`

Vous devez mettre à jour le fichier `gds.properties` avant d'exporter des transactions, des notifications, des articles et des catégories GDS.

Pourquoi et quand exécuter cette tâche

Pour exporter des objets de GDS version 1.3.x, vous devez indiquer les valeurs des propriétés suivantes :

- `txn_no_of_rows` : Indique le nombre d'enregistrements de transaction qui doivent être exportés dans chaque fichier XML.
- `ntf_no_of_rows` : Indique le nombre d'enregistrements de notification qui doivent être exportés dans chaque fichier XML.
- `item_no_of_rows` : Indique le nombre d'articles ou de catégories exportés dans un fichier `.csv`. Vous pouvez indiquer une valeur plus élevée en fonction des ressources matérielles disponibles.

Procédure

1. Dans le répertoire \$TOP/etc/default, ouvrez le fichier gds.properties en mode édition dans un éditeur de texte.
2. Ajoutez les propriétés d'exportation et définissez les valeurs appropriées. Par exemple, vous pouvez définir les propriétés en utilisant les valeurs recommandées suivantes :
 - txn_no_of_rows=1000
 - ntf_no_of_rows=1000
 - item_no_of_rows=5000
3. Enregistrez le fichier.

Arrêt de l'application GDS

Avant d'exporter l'environnement GDS à partir d'une instance de GDS version 1.3.x, vous devez arrêter tous les services WebSphere Product Center et GDS, ainsi que le registre RMI en cours d'exécution en lançant les scripts dans l'ordre indiqué.

Procédure

1. Arrêtez manuellement le planificateur dans WebSphere Product Center en exécutant le script stop_scheduler.sh. Ce script se trouve dans le répertoire <répertoire_installation_WPC>/bin/go/stop.
Avertissement : Vérifiez qu'il n'y a pas de travail critique en attente et que la file d'attente des travaux est vide.
2. Arrêtez le programme d'écoute des messages GDS en exécutant les scripts disponibles dans le répertoire <répertoire_installation_GDS>/bin dans l'ordre indiqué :
 - a. Arrêtez le script gdsmsg.sh avec le paramètre stop. Ce script se trouve dans le répertoire <répertoire_installation_GDS>/bin.

```
$<répertoire_installation_GDS>/bin/gdsmsg.sh stop
```
 - b. Après avoir patienté quelques minutes, exécutez le script gdsmsg.sh avec le paramètre abort. Ce script se trouve dans le répertoire <répertoire_installation_GDS>/bin. Le délai d'attente permet au programme d'écoute des messages GDS de finir le traitement qu'il a déjà commencé.

```
$<répertoire_installation_GDS>/bin/gdsmsg.sh abort
```
3. Arrêtez l'instance GDS en exécutant le script stop_appsvr.sh. Ce script se trouve dans le répertoire <répertoire_installation_GDS>/bin/go/stop.
4. Arrêtez l'instance de WebSphere Product Center en exécutant le script abort_local.sh. Ce script se trouve dans le répertoire <répertoire_installation_WPC>/bin/go. Ce script arrête le registre RMI et interrompt tous les services WebSphere Product Center exécutés sur le serveur.
5. Supprimez les processus Java ou les processus du registre RMI qui continuent à s'exécuter même après l'arrêt de l'instance. Plusieurs tentatives peuvent être nécessaires pour supprimer tous les processus Java. Vous devez continuer à supprimer tous les processus Java démarrés par WebSphere Product Center ou Global Data Synchronization jusqu'à ce qu'il n'y ait plus de processus indésirables en cours d'exécution.

Copie de sauvegarde de la base de données

Etant donné que le processus de migration met à jour le modèle de données dans la base de données, vous devez sauvegarder la base de données GDS avant de migrer de l'environnement GDS V1.3.x vers un GDS dans l'environnement IBM

InfoSphere Master Data Management Collaboration Server V10.1.x. Si des erreurs se produisent lors de la migration, vous pouvez restaurer la version précédente de la base de données GDS.

Pourquoi et quand exécuter cette tâche

Pour les utilisateurs d'Oracle, il existe une limite de 4 ko pour l'écriture de formulaires sérialisés dans des objets BLOB lorsque vous utilisez le pilote JDBC client Oracle 9i (sur n'importe quel serveur Oracle 9i ou Oracle 10g). Lors de la migration vers InfoSphere MDM Collaboration Server V10.1.x, installez le client Oracle 10gR2 pour obtenir les fichiers JAR 10gR2. Il est préférable d'installer le client au lieu de mettre à jour manuellement les fichiers .JAR dans le répertoire ORACLE_HOME.

Exportation à partir de WebSphere Product Center version 5.2.1.4

Vous devez utiliser les scripts inclus avec GDS pour exporter les métadonnées et les données. La procédure d'exportation est divisée en phases pour permettre la gestion du traitement des erreurs et de la reprise.

Avant de commencer

Pour exporter l'environnement GDS, vous devez au préalable effectuer les opérations suivantes :

- Exécutez le script `analyze_schema.sh` pour analyser et améliorer le schéma de base de données. Ce script se trouve dans le répertoire `$TOP/src/db/schema/util`, où `$TOP` est le répertoire d'installation de WebSphere Product Center. L'exécution de ce script améliore les performances d'une exportation, en particulier pour les articles et les catégories volumineuses.
- Vérifiez qu'il y a suffisamment d'espace disque disponible dans le système de fichiers.
- Vérifiez que seuls des processus pertinents sont en cours d'exécution. Pour améliorer les performances, arrêtez tous les services GDS et WebSphere Product Center.
- Envisagez d'exporter uniquement une partie du contenu du magasin de documents, même après le nettoyage du magasin. Au lieu d'exporter l'ensemble du magasin de documents, vous pouvez exporter uniquement les documents des répertoires pertinents, tels que `scripts` et `params`. Par exemple, le script suivant exporte les documents du répertoire `scripts` :

```
envObjList = new EnvObjectList ();
envObjList.addObjectByNameToExport ("/scripts","DOC_STORE");
sDocFilePath = "archives/docstoreExport.zip" +
out.writeln (exportEnv (envObjList, sDocFilePath));
```

Pourquoi et quand exécuter cette tâche

Si le système GDS se trouve dans un état cohérent, vous pouvez exporter chaque objet séparément ou exporter l'ensemble des objets globalement en utilisant l'option `export_type=all`. Il est inutile d'exporter les objets en respectant un ordre particulier. Toutefois, il est déconseillé d'exporter l'ensemble des objets globalement car il est difficile d'identifier et de résoudre les erreurs et les exceptions qui apparaissent pendant la procédure.

Le script suivant exporte tous les objets lorsque vous l'exécutez à partir du répertoire d'installation de GDS version 1.3.x utilisé sous la forme de \$TOP dans le script. Les journaux de ces étapes sont disponibles dans le répertoire \$TOP/GDSMigration/Logs.

```
$TOP/src/db/schema/gds/migration/9.1.0_migration/MigrateExport.sh  
--company_code=<cmp_code> --export_type=all
```

Toutefois, il est recommandé d'adopter l'approche étape par étape ci-dessous pour faciliter les opérations de débogage. Les options disponibles pour l'exportation sont les suivantes :

```
MigrateExport.sh --company_code=<code>  
--export_type=<firstSteps/0/1/2/3/4/5/6/7/8/9>
```

avec les options définies de la manière suivante :

company_code

Indiquez le code de la société.

Dans les scripts suivants, *<cmp_code>* est la valeur de la propriété **company_code** dans le fichier \$TOP/etc/default/gds.properties. Les journaux de ces étapes sont disponibles dans le répertoire \$TOP/GDSMigration/Logs.

export_type

Indiquez l'option d'exportation. La liste suivante répertorie les options d'exportation.

- *firstSteps* : Recherche les scripts associés aux hiérarchies et aux catalogues, extrait les attributs à indexer, obtient les informations du modèle de données et extrait le nombre de catalogues et d'articles.
- 0 - Prépare l'instance à l'exportation.
- 1 - Exporte les métadonnées de base [consultations, spécifications, collections d'attributs, paramètres de société, hiérarchie, catalogues, etc.].
- 2 - Exporte les données de sécurité [groupe de contrôle d'accès, rôle, utilisateurs, sélections, etc.].
- 3 - Exporte d'autres métadonnées [travaux, sources, rapports, etc.].
- 4 - Exporte des messages XML GDS.
- 5 - Exporte des objets de messagerie GDS.

Remarque : Cela n'est nécessaire que côté demande.

- 6 - Exporte des transactions.
- 7 - Exporte des notifications.
- 8 - Exporte le contenu des hiérarchies.
- 9 - Exporte le contenu des catalogues.

Procédure

1. Exécutez le script `MigrateExport.sh` en associant le paramètre **export_type** à la valeur `firstSteps`. Ce script se trouve dans le répertoire \$TOP/src/db/schema/gds/migration/10.0_migration. Le script recherche les scripts associés aux hiérarchies et aux catalogues, extrait les attributs à indexer, obtient les informations du modèle de données et extrait le nombre de catalogues et d'articles.

```
$TOP/src/db/schema/gds/migration/9.1.0_migration/MigrateExport.sh
--company_code=<cmp_code> --export_type=firstSteps
```

Cette étape est une étape préparatoire qui fournit à l'utilisateur des informations sur le système en cours et sur certaines actions à effectuer avant le lancement réel de l'exportation. Elle effectue les tâches suivantes :

- Crée le fichier `$TOP/GDSMigration/envInfo/CtgAttribInfo.txt` qui détaille les scripts associés aux catalogues du système.
 - Crée le fichier `$TOP/GDSMigration/envInfo/HierAttribInfo.txt` détaillant les scripts associés aux hiérarchies du système.
 - Crée le fichier `$TOP/GDSMigration/envInfo/pkAndDspIndexedAttrs.xml` contenant les attributs de clé primaire et d'affichage à ajouter manuellement dans les fichiers `$TOP/GDSMigration/Demand_Params.xml` (côté demande) ou `$TOP/GDSMigration/Supply_Params.xml` (côté fournisseur). Ces fichiers répertorient les attributs par défaut qui peuvent faire l'objet de recherches et qui doivent être indexés (pour permettre la recherche de ces attributs) dans InfoSphere MDM Collaboration Server version 10.0
 - Crée le fichier `$TOP/GDSMigration/envInfo/metaDataInfoList.txt` qui contient la liste de tous les objets de modèle de données dans le système GDS 1.3 en cours.
 - Crée le fichier `$TOP/GDSMigration/envInfo/itemAndCategoryCount.txt` qui contient le nombre total de catalogues et d'articles figurant sur le système GDS 1.3 en cours.
2. Exécutez le script `MigrateExport.sh` en associant le paramètre **export_type** à la valeur 0. Le script prépare l'instance à l'exportation.

```
$TOP/src/db/schema/gds/migration/9.1.0_migration/MigrateExport.sh
--company_code=<cmp_code> --export_type=0
```

Cette option est obligatoire car certains types d'objet et leurs définitions ne sont pas pris en charge dans InfoSphere MDM Collaboration Server version 10.0. L'exportation des données dans le format existant entraîne l'échec de la procédure d'importation. Par exemple, il est possible d'avoir un noeud de type `STRING` en tant que noeud parent d'un autre noeud (`GROUPING`) dans l'environnement GDS 1.3.x. Cela n'est pas possible dans l'environnement InfoSphere MDM Collaboration Server version 10.0. Vous devez donc la corriger avant d'exporter ces métadonnées.

3. Exécutez le script `MigrateExport.sh` en associant le paramètre **export_type** à la valeur 1. Le script importe des métadonnées de base, telles que les consultations, les spécifications, les collections d'attributs, les paramètres de société, les hiérarchies et les catalogues.

```
$TOP/src/db/schema/gds/migration/9.1.0_migration/MigrateExport.sh
--company_code=<cmp_code> --export_type=1
```

Les métadonnées exportées lors des étapes restantes dépendent de ces objets pour effectuer une importation correcte. Les objets suivants sont considérés comme des métadonnées de base :

- `COMPANY_ATTRIBUTES`
- `LOOKUP_TABLE_SPEC`
- `LOOKUP_TABLE`
- `LOOKUP_TABLE_CONTENT`
- `PRIMARY_SPEC`
- `SUB_SPEC`
- `DESTINATION_SPEC`

- SECONDARY_SPEC
- SCRIPT_INPUT_SPEC
- FILE_SPEC
- ATTRIBUTE_COLS
- HIERARCHY
- CATALOGUE
- MAPS
- ACG

Ces objets sont exportés dans le fichier COMPANYMETADATA.zip, puis téléchargés dans le répertoire \$TOP/GDSMigration/ExportedZip.

4. Exécutez le script MigrateExport.sh en associant le paramètre **export_type** à la valeur 2. Le script exporte des définitions de sécurité, telles que CONTAINER_ACCESSPRV, MY_SETTINGS, ROLES et USERS.

```
$TOP/src/db/schema/gds/migration/9.1.0_migration/MigrateExport.sh
--company_code=<cmp_code> --export_type=2
```

Les objets ci-dessus sont exportés dans le fichier SECURITY.zip, puis téléchargés dans le répertoire \$TOP/GDSMigration/ExportedZip.

5. Exécutez le script MigrateExport.sh en associant le paramètre **export_type** à la valeur 3. Le script exporte d'autres métadonnées, telles que des travaux, des sources et des rapports.

```
$TOP/src/db/schema/gds/migration/9.1.0_migration/MigrateExport.sh
--company_code=<cmp_code> --export_type=3
```

Dans cette étape, le script d'exportation traite d'autres métadonnées, notamment :

- DATASOURCE
- FEEDS
- JOBS
- DISTRIBUTION
- EXPORTS
- REPORTS
- HIERARCHY_VIEW
- CATALOG_VIEW
- DISTRIBUTION_GROUP
- ALERT
- QUEUE
- SEARCH_TEMPLATES
- UDL
- HIERARCHY_MAPS

Les objets ci-dessus sont exportés dans le fichier OTHERMETADATA.zip, puis téléchargés dans le répertoire \$TOP/GDSMigration/ExportedZip.

6. Exécutez le script MigrateExport.sh en associant le paramètre **export_type** à la valeur 4. Le script exporte les messages XML GDS.

```
$TOP/src/db/schema/gds/migration/9.1.0_migration/MigrateExport.sh
--company_code=<cmp_code> --export_type=4
```

Tous les messages XML échangés entre un système GDS et le pool de données connecté sont stockés dans le magasin de documents de WebSphere Product

Center version 5.2.1.4. Les objets suivants sont considérés comme des messages XML GDS et les éléments suivants du magasin de documents doivent être exportés :

- Les messages XML stockés dans le chemin du magasin de documents /archive/messages/inbound/
- Les messages XML stockés dans le chemin du magasin de documents /archives/messages/outbound/

Ces messages XML sont exportés, puis téléchargés dans les répertoires \$TOP/GDSMigration/messages/inbound et \$TOP/GDSMigration/messages/outbound avec les fichiers de contrôle uplInXMLMsg.xml et uplOutXMLMsg.xml. Les deux fichiers de contrôle sont stockés dans le répertoire \$TOP/GDSMigration/messages.

7. Pour *Demand side only*, exécutez le script MigrateExport.sh avec le paramètre **type_export** défini sur 5. Le script exporte les objets de messagerie GDS.

```
$TOP/src/db/schema/gds/migration/9.1.0_migration/MigrateExport.sh  
--company_code=<cmp_code> --export_type=5
```

Tous les objets de messagerie GDS créés lors de l'échange des messages XML entre GDS et le pool de données connecté sont stockés dans le magasin de documents de WebSphere Product Center version 5.2.1.4 et doivent être exportés.

Les objets suivants sont considérés comme des objets de messagerie GDS :

- Tous les objets stockés dans le magasin de documents dans /object/messaging/

Ces objets sont exportés, puis téléchargés dans le répertoire \$TOP/GDSMigration/object/messaging avec le fichier de contrôle uplMessObj.xml dans \$TOP/GDSMigration/object.

8. Exécutez le script MigrateExport.sh en associant le paramètre **export_type** à la valeur 6. Le script exporte les transactions.

```
$TOP/src/db/schema/gds/migration/9.1.0_migration/MigrateExport.sh  
--company_code=<cmp_code> --export_type=6
```

Les transactions sont des objets GDS de base et sont considérées comme des TRANSACTIONS. Elles sont exportées dans le fichier TRANSACTIONS.zip, puis téléchargées dans le répertoire \$TOP/GDSMigration/ExportedZip.

9. Exécutez le script MigrateExport.sh en associant le paramètre **export_type** à la valeur 7. Le script exporte les notifications.

```
$TOP/src/db/schema/gds/migration/9.1.0_migration/MigrateExport.sh  
--company_code=<cmp_code> --export_type=7
```

Les notifications sont des objets GDS de base et sont considérées comme des NOTIFICATIONS. Elles sont exportées dans le fichier NOTIFICATIONS.zip, puis téléchargées dans le répertoire \$TOP/GDSMigration/ExportedZip.

10. Exécutez le script MigrateExport.sh en associant le paramètre **export_type** à la valeur 8. Le script exporte le contenu des hiérarchies.

```
$TOP/src/db/schema/gds/migration/9.1.0_migration/MigrateExport.sh  
--company_code=<cmp_code> --export_type=8
```

Cette phase ne s'applique pas aux autres objets de données de la société car le contenu des hiérarchies représente l'essentiel d'une procédure d'exportation. Si des hiérarchies comportent des données particulièrement volumineuses (catégories), cette phase peut être fractionnée au niveau des hiérarchies en lançant une procédure d'exportation distincte pour traiter le contenu de

chaque hiérarchie. Il est également possible de regrouper toutes les hiérarchies qui comportent peu de données dans un fichier .zip et d'exporter les hiérarchies plus volumineuses dans leurs fichiers zip respectifs. Dans une implémentation GDS, le contenu des hiérarchies est généralement énorme et ne peut pas être exporté en l'état à l'aide de l'environnement d'exportation WebSphere. Par conséquent, un utilitaire d'exportation personnalisé est fourni pour exporter ce type de contenu. Les objets HIERARCHY_CONTENT(Categories) sont considérés comme des contenus de hiérarchies.

Ces objets sont exportés, puis téléchargés dans le répertoire \$TOP/GDSMigration/feed_files/hierarchies avec le fichier de contrôle uplHieCSV.xml dans \$TOP/GDSMigration/feed_files.

Remarque : Le fichier de données (.csv) est généré uniquement pour les hiérarchies incluant au moins une catégorie.

11. Exécutez le script MigrateExport.sh en associant le paramètre **export_type** à la valeur 9. Le script exporte le contenu des catalogues.

```
$TOP/src/db/schema/gds/migration/9.1.0_migration/MigrateExport.sh  
--company_code=<cmp_code> --export_type=9
```

Dans cette étape, le contenu du catalogue GDS est exporté. Cette phase ne s'applique pas aux autres objets de données de la société car le contenu des catalogues représente l'essentiel d'une procédure d'exportation. Si des catalogues comportent des données particulièrement volumineuses (articles), cette phase peut être fractionnée au niveau des catalogues en lançant une procédure d'exportation distincte pour traiter le contenu de chaque catalogue. Il est également possible de regrouper tous les catalogues qui comportent peu de données dans un fichier .zip et d'exporter les catalogues plus volumineux dans leurs fichiers zip respectifs. Dans une implémentation GDS, le contenu des catalogues est généralement énorme et ne peut pas être exporté en l'état à l'aide de l'environnement d'exportation WebSphere. Un utilitaire d'exportation personnalisé est donc fourni pour exporter le contenu des catalogues. Les objets CATALOG_CONTENT sont considérés comme des contenus de catalogues.

Ces objets sont exportés, puis téléchargés dans le répertoire \$TOP/GDSMigration/feed_files/catalogs avec le fichier de contrôle uplCtgCSV.xml dans \$TOP/GDSMigration/feed_files.

Remarque : Le fichier de données (.csv) est généré uniquement pour les catalogues incluant au moins un article.

Le nombre d'articles figurant dans un fichier csv varie en fonction de la valeur définie pour item_no_of_rows dans le fichier gds.properties de l'environnement GDS version 1.3.x exporté.

Que faire ensuite

Reportez-vous aux journaux pour déterminer comment l'exportation s'est déroulée. Si certains objets n'ont pas été correctement exportés, vous pouvez réexécuter l'exportation uniquement pour les objets concernés. Le fichier compressé créé peut être extrait afin de rechercher les informations manquantes en vous basant sur les journaux des erreurs. Les scripts d'exportation peuvent être exécutés plusieurs fois.

Limitations du script d'exportation

Après la fin d'une opération d'exportation, vous devez vous assurer que tous les objets sont exportés et que toutes les dépendances sont satisfaites. Le script

d'exportation ne vérifie pas que toutes les dépendances sont exportées et ne détermine pas si le fichier compressé créé peut être importé.

Limitations au niveau de WebSphere Product Center

- Exportations de la version précédente : l'outil de migration n'exporte pas toutes les versions des métadonnées et des données marquées dans WebSphere Product Center. Seules les versions les plus récentes de ces objets sont exportables. L'annulation du catalogue sur les données importées depuis l'exportation en cours avec restauration vers un niveau précédent n'est pas prise en charge car toutes les références sur les catalogues et les catégories n'existent pas. Vous ne pouvez pas exporter les différences de version car seule la version la plus récente est disponible dans le système cible. Par exemple, si WebSphere Product Center comporte plusieurs versions d'un objet de catalogue, seul la version la plus récente du catalogue est exportée.
- Exportations d'objets par défaut : l'outil de migration n'exporte pas les objets créés par défaut lorsqu'une nouvelle société est créée. Si votre implémentation en cours contient des objets altérés comme la spécification primaire de hiérarchie par défaut, la spécification d'importation de la table de consultation et la hiérarchie de tables de consultation par défaut, les modifications ne seront pas effectuées car l'outil s'attend à les trouver dans la société cible. Avant d'importer ces objets, vous devez les modifier manuellement dans IBM InfoSphere Master Data Management Collaboration Server V10.1.x pour refléter leurs structures dans WebSphere Product Center V5.2.1.4.
- Hiérarchies d'organisation : dans WebSphere Product Center version 5.2.1.4, vous ne pouvez pas utiliser l'outil de migration pour exporter les hiérarchies d'organisation. Vous devez créer manuellement ces hiérarchies dans la société cible.
- Réorganisation des éléments/des catégories : l'exportation de la réorganisation des éléments/catégories n'est pas prise en charge.
- Mappage rôle-paramètres régionaux : l'exportation du mappage rôle-paramètres régionaux n'est pas prise en charge.
- Taille de valeur d'attribut : les attributs dont la valeur de type entier est supérieure à l'entier MAX_VALUE ne sont pas importés.
- Utilisateurs sans rôle : les utilisateurs qui n'ont pas été associés à un rôle WebSphere Product Center ne sont pas exportés.
- Langage d'interrogation de sélection avancée : le langage (règle) est modifié dans InfoSphere MDM Collaboration Server v10.1.x. Vous devez migrer manuellement toutes les requêtes qui ont été créées avant WebSphere Product Center version 5.3.0.

Dans InfoSphere MDM Collaboration Server V10.1.x, les sélections avancées ne sont pas migrées automatiquement. Vous devez réécrire les règles dans InfoSphere MDM Collaboration Server V10.1.x. Vous pouvez utiliser les opérations de script suivantes dans InfoSphere MDM Collaboration Server v10.1.x :

- new DynamicSelection(String name, String WQL)
- DynamicSelection::saveSelection()

WebSphere Product Center version 5.3.2 n'autorise pas d'opération saveSelection si le langage WQL (WebSphere Query Language) n'est pas valide. Par conséquent, vous devez exécuter ce script avec le langage WQL corrigé et valide. Vous pouvez migrer la sélection dynamique en procédant comme suit :

1. Exportez toutes les sélections de WebSphere Product Center version 5.2.1 et groupes de correctifs ultérieurs sous la forme de fichiers .csv où la première valeur sera le nom et la seconde la chaîne de règle.
2. Convertissez les règles en langage WQL valide. Pour plus d'informations sur le langage WQL, consultez la rubrique sur le langage de requête.
3. Importez des règles et des sélections à l'aide d'opérations de script. Les fichiers .csv peuvent être téléchargés en amont dans le magasin de documents ou les valeurs peuvent être lues depuis le volet d'entrée de script dans le bac à sable.

Voici un exemple de script pour des raisons de commodité.

Volet d'entrée de script

```
selection1, select item from catalog('cata')
  where item['spec1/primary key'] = '11'
selection2, select item from catalog('cata')
  where item['spec1/primary key'] = '11'
selection3, select item from catalog('cata')
  where item['spec1/primary key'] = '11'
```

Volet de script

```
var parser = newCSVParser(in);
var bDone = false;
var e = "";
catchError(e)
{
  while(!bDone)
  {
    attrs = parser.splitLine();
    if(attrs!= null)
    {
      a = new DynamicSelection(attrs[0], attrs[1]);
      a.saveSelection();
      out.writeln(attrs[0] + " created");
    }
    bDone = (null == attrs);
  }
}
out.writeln(e);
```

- Informations sur les relations : le contenu des hiérarchies qui comportent des catégories avec des données spécifiées pour des attributs de relation ne peut pas être importé. Pour contourner ce problème, procédez comme suit :
 - Extrayez le fichier compressé généré à partir de l'exportation de la société.
 - Placez-vous dans le répertoire *HIERARCHY_CONTENTS*.
 - Modifiez chaque fichier .csv et supprimez la ou les colonnes qui indiquent des données de relation. N'oubliez pas de sauvegarder les fichiers après les avoir modifiés.
 - Créez un nouveau fichier compressé et ajoutez-y les fichiers mis à jour. La structure de répertoire du fichier zip doit être identique à celle du fichier compressé que vous avez extrait. Le fichier `ImportEnvControl.xml` et le répertoire d'archives doivent être au niveau racine.
 - Importez le nouveau fichier zip à la place du fichier zip exporté.
- Migration des fichiers de propriétés : vous devez migrer manuellement tous les fichiers de propriétés WebSphere Product Center. Pour plus d'informations sur chaque fichier de propriétés, voir la documentation du produit InfoSphere MDM Collaboration Server V10.1.x.

- Migration du répertoire `ctg_files` : vous devez copier manuellement le contenu du répertoire `$TOP/public_html/suppliers/<cmp code>/ctg_files` vers la nouvelle instance de chaque société.
- Migration des fichiers `.jar` utilisateur : vous devez ajouter manuellement les fichiers `.jar` obligatoires à la nouvelle instance. Voir le centre de documentation InfoSphere MDM Collaboration Server V10.1.x pour plus d'informations sur l'ajout des fichiers utilisateur nécessaires `.jar` à une nouvelle instance.
- Exportation et importation des données héritées : vous ne pouvez pas exporter et importer de données héritées. Le contenu de catalogue et de hiérarchie qui prend en charge l'héritage ne peut pas être exporté ou importé.

Limitations de niveau système

Les opérations d'exportation à l'aide de l'outil de migration ne sont pas réalisables lorsqu'une limitation de niveau système est atteinte. Les limitations communes sont les suivantes :

- Limitation de système de fichiers : bon nombre de systèmes d'exploitation comportent des limitations de système de fichiers. Lorsque vous utilisez un système de fichiers en particulier, vous devez décider comment fractionner une exportation en fonction de la taille des métadonnées et des données.
- Limitation de format de fichier compressé : l'outil de migration `tool` comprime les fichiers exportés, puis les télécharge dans le magasin de documents. L'utilitaire de compression peut avoir une limite de taille supérieure pour un fichier compressé unique et un nombre maximal de fichiers par fichier compressé. Pour résoudre ce problème, divisez les exportations de manière logique afin que le nombre de fichiers générés soit inférieur à la limite supérieure.
- Le fichier compressé généré pendant l'opération d'exportation apparaît vide : le fichier compressé généré à l'aide de l'opération d'exportation de la fonction de migration est vide lorsque vous l'ouvrez à l'aide de l'utilitaire `Dossiers` compressés de Windows XP. Il s'agit d'un problème connu. Pour éviter ce problème, ouvrez le fichier compressé à l'aide de l'application `Winzip`.

Exportation du magasin de documents

Vous devez exporter séparément le magasin de documents.

Exportez séparément le magasin de documents car il stocke un contenu abondant et son exportation prend relativement plus de temps que celle d'un autre objet. Vous pouvez exporter certains composants ou la totalité du magasin de documents. Puisque celui-ci est souvent trop volumineux pour un fichier compressé unique, vous devrez sans doute l'exporter en plusieurs composants.

Exportation partielle/sélective du magasin de documents

Au cours de cette procédure, le magasin de documents est exporté en plusieurs composants. L'exemple de script suivant démontre l'exportation d'un composant nommé `/object/messaging/` du magasin de documents :

//Sample script

```
var docStorePath = "GDSMigration/ExportedZip/";
var sDocFilePath = docStorePath+"MESSAGINGOBJECTS.zip";
var envObjList = new EnvObjectList();
envObjList.setTypeToExport("DOC_STORE");
envObjList.addObjectByNameToExport("/object/messaging/");
//exporter les objets sélectionnés dans le magasin de documents
var result = exportEnv(envObjList, sDocFilePath);
```

```
//vérifier la bonne exécution de l'exportation
if( result.contains("Uploaded the archived file to docstore") ) {
    out.println("résultat " + result);
}
else {
    //si l'exportation échoue, le texte du résultat
    //apparaîtra sous forme de message d'erreur
    throwError(result);
}
```

Cet exemple de script exporte la section /object/messaging/ du magasin de documents et la sauvegarde dans le magasin de documents au format/GDSMigration/ExportedZip/MESSAGINGOBJECTS.zip.

Exportation complète du magasin de documents

Au cours de cette procédure, le magasin de documents est exporté dans sa totalité. Exportez tout le magasin de documents uniquement lorsque son contenu tient dans un fichier compressé unique. L'exemple de script suivant exporte tout le magasin de documents :

```
//Sample script

var docStorePath = "GDSMigration/ExportedZip/";
var sDocFilePath = docStorePath+"COMPANYDOCSTORE.zip";
var envObjList = new EnvObjectList();
envObjList.addAllObjectsToExport("DOC_STORE");
    //exporter tous les objets dans le magasin de documents
out.println("Enregistrer l'environnement sous " + sDocFilePath);
var result = exportEnv(envObjList, sDocFilePath);
//vérifier la bonne exécution de l'exportation
if( result.contains("Uploaded the archived file to docstore") ) {
    out.println("résultat " + result);
}
else {
    //si l'exportation échoue, le texte du résultat
    //apparaît sous forme de message d'erreur
    throwError(result);
}
```

Ce script exporte tout le magasin de documents et stocke le fichier compressé généré dans celui-ci au format /GDSMigration/ExportedZip/COMPANYDOCSTORE.zip.

Remarque : L'exemple de script n'est pas inclus dans le script MigrateExport.sh. L'objet DOC_STORE est également appelé magasin de documents.

Importation vers InfoSphere MDM Collaboration Server V10.1

Utilisez le script MigrateImport.sh pour importer des objets GDS dans l'environnement IBM InfoSphere Master Data Management Collaboration Server V10.1.

Avant de commencer

Pour importer l'environnement GDS, vous devez au préalable effectuer les opérations suivantes :

- Vérifiez que toutes les configurations et installations nécessaires à GDS dans InfoSphere MDM Collaboration Server V10.1 ont été effectuées. Vous pouvez vous reporter aux configurations des propriétés GDS dans une instance GDS 1.3.x active. Pour plus d'informations, voir Configuration de la fonction GDS.

- Vérifiez que les scripts de création de schéma, `create_schema.sh`, et de création de société, `create_cmp.sh`, se sont exécutés correctement. Ces deux scripts se trouvent dans le répertoire `$<répertoire_installation>/bin/db`.
- Copiez le répertoire `GDSMigration` et tout son contenu dans le répertoire d'installation GDS v1.3.x vers le répertoire d'installation InfoSphere MDM Collaboration Server V10.1.

Pourquoi et quand exécuter cette tâche

Vous devez suivre l'approche progressive dans le processus d'importation pour importer les données vers InfoSphere MDM Collaboration Server V10.1. Chaque étape requiert l'exécution du script d'importation avec le paramètre **import_type** associé à une autre valeur. Vous devez vérifier que chaque étape se déroule correctement avant de passer à l'étape suivante car les objets que vous importez au cours de certaines étapes dépendent d'autres objets que vous importez lors des étapes précédentes. Le respect de l'ordre indiqué permet d'éviter les échecs dus aux dépendances d'objets.

Pratiques recommandées : A l'heure actuelle, il n'y a aucun moyen de connaître les éléments qui ont échoué lors de l'importation car le rapport de débogage généré par l'outil ne consigne pas ce type d'informations. Les scripts qui gèrent l'importation de contenu, tel que les articles et les catégories, sont les suivants :

- `/scripts/import/ctg/environment_catalog_content_import.script`
- `/scripts/import/ctr/environment_category_tree_content_import.script`

Vous pouvez modifier ces scripts pour permettre la consignation des articles et des catégories qui ont échoué.

Dans InfoSphere MDM Collaboration Server V10.1, la propriété **debug_profile_db_connections** dans le fichier `common.properties`, et le code associé qui crée `LiteDBManager` comme gestionnaire BD, a été supprimée. Par conséquent, `ThinPoolDBManager` est le seul gestionnaire BD pour InfoSphere MDM Collaboration Server V10.1.

Le script suivant importe des objets lorsqu'il est exécuté depuis le répertoire d'installation InfoSphere MDM Collaboration Server V10.1 utilisé comme `$TOP` dans le script. Les journaux de ces étapes se trouvent dans le répertoire `$TOP/GDSMigration/Logs`.

```
$TOP/src/db/schema/gds/migration/v9/MigrateImport.sh
--company_code=<cmp_code> --import_type=<gdsprev/0/1/2/3/4/5/6/7/8/9/10>
```

avec les options définies de la manière suivante :

company_code

Indiquez le code de la société.

Dans les scripts suivants, `<cmp_code>` est la valeur de la propriété **company_code** dans le fichier `$TOP/etc/default/gds.properties`.

import_type

Indiquez l'option d'importation. La liste suivante répertorie les options d'importation.

- `gdsprev` : Charge les droits GDS
- `0` - Transforme les fichiers compressés exportés.

- 1 - Importe des métadonnées, telles que les consultations, les spécifications, les collections d'attributs, les paramètres de société, les hiérarchies et les catalogues.
- 2 - Importe les catégories de contenu des hiérarchies.
- 3 - Importe des objets de sécurité, tels que les rôles, les utilisateurs et les sélections.
- 4 - Crée des flux de travaux GDS et charge des scripts.
- 5 - Importe d'autres métadonnées, telles que des travaux, des sources et des rapports.
- 6 - Importe des messages XML GDS.
- 7 - Importe des objets de messagerie GDS.

Remarque : Cela n'est nécessaire que côté demande.

- 8 - Importe les articles de contenu des catalogues.
- 9 - Importe des transactions GDS.
- 10 - Importe des notifications GDS.

Procédure

1. Exécutez le script `MigrateImport.sh` en associant le paramètre **import_type** à la valeur `gdsprev`. Ce script se trouve dans le répertoire `$TOP/src/db/schema/gds/migration/10.0_migration`. Le script charge les droits GDS.

```
$TOP/src/db/schema/gds/migration/v9/MigrateImport.sh
--company_code=<cmp_code> --import_type=gdsprev
```

Dans cette étape, le script charge les privilèges GDS dans l'environnement InfoSphere MDM Collaboration Server V10.1. Comme il n'y a pas de modèle de données GDS chargé par défaut, il est nécessaire de charger les droits en premier.

Remarque : Cette étape est obligatoire.

2. Exécutez le script `MigrateImport.sh` en associant le paramètre **import_type** à la valeur `0`. Le script transforme les fichiers compressés exportés.

```
$TOP/src/db/schema/gds/migration/v9/MigrateImport.sh
--company_code=<cmp_code> --import_type=0
```

Cette étape convertit les fichiers compressés exportés de l'instance GDS v1.3.x dans le format InfoSphere MDM Collaboration Server V10.1. Le script lit tous les fichiers zip exportés dans le répertoire `$TOP/GDSMigration/ExportedZip`, les transforme et les place dans le répertoire `$TOP/GDSMigration/TransformedZip`.

Remarque : Une exception `SAXParseException` apparaît lorsque vous convertissez le fichier `COMPANYMETADATA.zip` car le fichier `MAPS.xml` n'est pas un fichier XML bien qu'il ait l'extension `.xml`. Vous pouvez ignorer cette exception.

3. Exécutez le script `MigrateImport.sh` en associant le paramètre **import_type** à la valeur `1`. Le script importe des métadonnées, telles que les consultations, les spécifications, les collections d'attributs, les paramètres de société, les hiérarchies et les catalogues.

```
$TOP/src/db/schema/gds/migration/v9/MigrateImport.sh
--company_code=<cmp_code> --import_type=1
```

Cette étape est la première étape de la procédure d'importation réelle des métadonnées et des données exportées. Vous devez importer le fichier compressé des métadonnées de base car les importations ultérieures dépendent des métadonnées.

Le script importe le contenu du fichier COMPANYMETADATA.zip transformé dans le répertoire \$TOP/GDSMigration/TransformedZip.

4. Exécutez le script MigrateImport.sh en associant le paramètre **import_type** à la valeur 2. Le script importe les hiérarchies et les catégories de contenu.

```
$TOP/src/db/schema/gds/migration/v9/MigrateImport.sh  
--company_code=<cmp_code> --import_type=2
```

Dans cette étape, le contenu des hiérarchies est importé après l'importation des métadonnées de société et de base vers InfoSphere MDM Collaboration Server V10.1. Cette étape charge tout le contenu des hiérarchies (dans des fichiers .csv), ainsi que les scripts d'importation dans le magasin de données et crée des travaux d'importation de hiérarchies qui doivent être démarrés manuellement après l'exécution réussie de ce script. Après l'exécution réussie du script, démarrez le travail d'importation manuellement. Voir la tâche suivante pour importer le contenu des hiérarchies vers l'instance InfoSphere MDM Collaboration Server V10.1 : Exécution immédiate des travaux d'importation du contenu des hiérarchies.

5. Exécutez le script MigrateImport.sh avec le paramètre **import_type** associé à la valeur 3. Le script importe des composants de sécurité, tels que les groupes de contrôle d'accès, les rôles, les utilisateurs et les sélections.

```
$TOP/src/db/schema/gds/migration/v9/MigrateImport.sh  
--company_code=<cmp_code> --import_type=3
```

Les composants de sécurité comprennent les groupes de contrôle d'accès, les utilisateurs, les sélections, les droits d'accès aux conteneurs et Mes paramètres. Le script importe le contenu du fichier transformé SECURITY.zip dans le répertoire \$TOP/GDSMigration/TransformedZip.

6. Exécutez le script MigrateImport.sh avec le paramètre **import_type** associé à la valeur 4. Le script crée les flux de travaux GDS et charge les scripts.

```
$TOP/src/db/schema/gds/migration/v9/MigrateImport.sh  
--company_code=<cmp_code> --import_type=4
```

Les scripts, flux de travaux et zones de collaboration ne sont pas exportés depuis l'environnement GDS version 1.3.x mais créés ou chargés pendant cette étape de migration.

7. Exécutez le script MigrateImport.sh en associant le paramètre **import_type** à la valeur 5. Le script importe d'autres métadonnées, telles que des travaux, des flux et des rapports.

```
$TOP/src/db/schema/gds/migration/v9/MigrateImport.sh  
--company_code=<cmp_code> --import_type=5
```

Le script importe toutes les métadonnées restantes qui dépendent des objets importés à l'étape précédente. Il importe le contenu du fichier transformé OTHERMETADATA.zip stocké dans le répertoire \$TOP/GDSMigration/TransformedZip.

8. Exécutez le script MigrateImport.sh en associant le paramètre **import_type** à la valeur 6. Le script importe les messages XML GDS XML vers le magasin de documents de l'instance cible InfoSphere MDM Collaboration Server V10.1.

```
$TOP/src/db/schema/gds/migration/v9/MigrateImport.sh  
--company_code=<cmp_code> --import_type=6
```

9. Pour *Demand side only*, exécutez le script `MigrateImport.sh` avec le paramètre **type_import** défini sur 7. Le script importe les objets de messagerie GDS vers le magasin de documents de l'instance InfoSphere MDM Collaboration Server V10.1.

```
$TOP/src/db/schema/gds/migration/v9/MigrateImport.sh  
--company_code=<cmp_code> --import_type=7
```

10. Exécutez le script `MigrateImport.sh` en associant le paramètre **import_type** à la valeur 8. Le script importe les articles du contenu des catalogues. Pour plus d'informations, voir décrivant l'exécution immédiate des travaux d'importation du contenu des hiérarchies et des catalogues.

```
$TOP/src/db/schema/gds/migration/v9/MigrateImport.sh  
--company_code=<cmp_code> --import_type=8
```

Importez le contenu des catalogues après tous les objets autres que les transactions et notifications et certaines parties du magasin de données vers l'instance InfoSphere MDM Collaboration Server V10.1.

Le script charge l'ensemble du contenu des catalogues (dans des fichiers .csv) et les scripts d'importation des catalogues dans le magasin de documents et crée des travaux d'importation des catalogues. Le nombre d'articles figurant dans un fichier csv varie en fonction de la valeur définie pour le paramètre **item_no_of_rows** dans le fichier `gds.properties` de l'environnement GDS 1.3.x exporté. Par exemple, chaque fichier .csv stockant le contenu des catalogues contient 1000 articles. Pour les 1000 articles suivants, un autre fichier .csv est créé.

Le format du nom des fichiers .csv est le suivant :

<nom_catalogue>[1,2,3..].csv. Par exemple, si le catalogue `Global_Catalog` contient 2000 articles, il y a deux fichiers .csv pour les données relatives aux articles. Ces fichiers sont appelés `Global_Catalog1.csv` et `Global_Catalog2.csv`.

Remarque : Vous pouvez importer le contenu du catalogue `TradeCatalog` uniquement après avoir importé le contenu des catalogues `Global_Catalog` et `Global_Local_Catalog`.

11. Exécutez le script `MigrateImport.sh` en associant le paramètre **import_type** à la valeur 9. Le script exporte les transactions GDS.

```
$TOP/src/db/schema/gds/migration/v9/MigrateImport.sh  
--company_code=<cmp_code> --import_type=9
```

Les transactions GDS dépendent des utilisateurs, des articles et des catégories. Importez les transactions uniquement après avoir importé correctement tous les objets dont elles dépendent.

Le script importe le contenu du fichier transformé `TRANSACTIONS.zip` stocké dans le répertoire `$TOP/GDSMigration/TransformedZip`.

12. Exécutez le script `MigrateImport.sh` en associant le paramètre **import_type** à la valeur 10. Le script exporte les notifications GDS.

```
$TOP/src/db/schema/gds/migration/v9/MigrateImport.sh  
--company_code=<cmp_code> --import_type=10
```

Les notifications GDS dépendent des utilisateurs, des articles, des catégories et des transactions. Vous devez les importer uniquement après avoir importé avec succès tous ces objets. Le script importe le contenu du fichier `NOTIFICATIONS.zip` converti présent dans le répertoire `$TOP/GDSMigration/TransformedZip`.

Résultats

Après l'importation, vous constaterez une différence au niveau du nombre d'articles, de transactions et de notifications entre GDS 1.3.1 et GDS dans les environnements IBM InfoSphere Master Data Management Collaboration Server V10.1. Ces éléments ont en effet des dépendances entre eux lors de l'importation.

Par exemple, si un article a été supprimé de l'environnement GDS 1.3.1.x, ses transactions et notifications ne sont pas supprimées. Lors d'une exportation, un tel article n'est pas extrait ; une importation ne peut donc pas non plus créer un tel environnement. Lorsque des transactions et des notifications sont importées, le processus vérifie que les articles auxquels ces transactions et ces notifications correspondent existent. Si l'article n'existe pas, ces transactions et ces notifications ne sont pas importées ; en effet, cela générerait des exceptions dans l'interface utilisateur si vous cliquiez sur une notification à laquelle aucun article n'est associé. Il existe donc des différences entre les deux environnements.

Que faire ensuite

- Vous devez vérifier les rapports générés par une opération d'importation pour vous assurer qu'elle s'est déroulée correctement.
- Vous pouvez également vérifier les journaux InfoSphere MDM Collaboration Server pour identifier les erreurs et les exceptions éventuelles.
- Vous devez exécuter les étapes suivantes avant de démarrer les services InfoSphere MDM Collaboration Server V10.1 sur les données importées :
 1. Ajoutez des entrées pour les tables de consultation créées par l'équipe d'implémentation dans GDS V1.3.x dans le fichier `GDSCacheRegistry.xml` dans l'instance InfoSphere MDM Collaboration Server V10.1. Ce fichier se trouve dans le répertoire `$TOP/etc/default`.
 2. Mettez à jour la liste dans le fichier `GDSCacheRegistry.xml` stocké dans le répertoire `$TOP/etc/default` :
 - Volume Unit of Measure
 - Weight Unit of Measure
 - Linear Unit of Measure
 - Language_LookUp

Exécution immédiate des travaux d'importation des hiérarchies et du contenu des catalogues

Vous devez importer le contenu des hiérarchies après l'importation des métadonnées de société et de base vers l'instance IBM InfoSphere Master Data Management Collaboration Server V10.1.x. Vous devez démarrer les travaux d'importation manuellement après avoir exécuté le script d'importation pour importer le contenu des catalogues vers l'instance IBM InfoSphere Master Data Management Collaboration Server V10.1.x.

Avant de commencer

Il est possible que le système crée des travaux supplémentaires dotés de la chaîne `CURRENT_DATE` dans la zone de description. Supprimez-les sans les exécuter.

Pourquoi et quand exécuter cette tâche

Après avoir exécuté le script d'importation `MigrateImport.sh` en affectant les valeurs suivantes au paramètre `import_type` :

- 2 pour une hiérarchie

- 8 pour le catalogue

Vous devez lancer les travaux d'importation des hiérarchies et des catalogues manuellement.

Procédure

1. Supprimez le répertoire \$TOP/logs.
2. Les importations prennent du temps ; une session utilisateur peut donc s'interrompre. Vérifiez que les bonnes valeurs sont définies pour la propriété **intervalle_inactif_max** du fichier \$TOP/etc/default/common.properties. Pour plus d'informations, voir `intervalle_inactif_max`.
3. Démarrez uniquement les services du serveur d'applications et du planificateur de InfoSphere MDM Collaboration Server. Pour savoir comment démarrer ces services, voir `Starting a service`.
4. Connectez-vous à InfoSphere MDM Collaboration Server V10.1.x comme administrateur.
5. Accédez à **Data Model Manager->Scheduler->Jobs Console**.
6. Cliquez sur **Ajouter** pour les travaux dont la description contient le nom de hiérarchie ou de catalogue à importer, par exemple, `NomHiérarchie.csv`.
7. Associez **Planifier** à **Exécuter immédiatement**.

Remarque : Si le serveur d'applications et le serveur de base de données possèdent une date ou une heure différente, vous pouvez démarrer le travail en utilisant l'option d'**exécution unique** et en antidatant le travail.

8. Cliquez sur **Soumettre** avec la valeur par défaut **Aucun**. L'état du travail apparaît dans l'écran de la console de travaux.
9. Cliquez sur l'icône Action pour afficher l'état d'exécution.
10. Recherchez les erreurs ou les échecs éventuels à l'issue de l'exécution du travail. Le travail doit disposer de suffisamment de temps pour s'exécuter.
11. Vérifiez que les journaux du planificateur et du serveur d'applications dans le répertoire \$TOP ne contiennent pas d'échecs ou d'exceptions.
12. Répétez les étapes 5 à 11 pour le reste des hiérarchies et des catalogues à importer.
13. Supprimez les fichiers de flux des sources de données (au format .csv), ainsi que le travail d'importation qui a abouti dans le cadre du nettoyage.
14. Arrêtez les services InfoSphere MDM Collaboration Server et passez à l'étape d'importation suivante.

Limitations du script d'importation

Utilisez ces instructions pour contourner les limitations du script d'importation.

- Vérifiez que le fichier compressé est compatible avec IBM InfoSphere Master Data Management Collaboration Server V10.1.x. Le script d'importation ne pourra pas importer le fichier compressé s'il n'est pas compatible avec IBM InfoSphere Master Data Management Collaboration Server V10.1.x.
- Assurez-vous que les opérations d'importation ne sont pas arrêtées par les utilisateurs ou par des défaillances au niveau du système. Si une opération d'importation est arrêtée, l'état de l'application est inconnu et la base de données risque d'être endommagée.
- Assurez-vous que le fichier compressé contient tous les objets dépendants. Si ce n'est pas le cas, vous devez alors créer ces objets manuellement dans l'application ou à l'aide des fichiers compressés importés précédemment.

Vous pouvez rechercher dans le rapport de débogage des informations sur les incidents d'importation. Cependant, le rapport de débogage risque de ne pas enregistrer la cause exacte d'un incident. Pour plus d'informations sur les incidents d'importation, consultez les journaux InfoSphere MDM Collaboration Server.

Les modifications apportées aux scripts dans l'instance GDS V1.3.x sont perdues au cours de la migration. Vous devez migrer manuellement les modifications vers InfoSphere MDM Collaboration Server V10.1.x.

Objets Collaboration Server non pris en charge par la migration GDS

Quelques objets Collaboration Server, majoritairement des personnalisations, ne sont pas gérés par la migration GDS.

Par exemple, les rapports Collaboration Server ne sont pas gérés par la migration GDS.

Les rapports attachés aux scripts d'implémentation doivent être modifiés pour fonctionner dans IBM InfoSphere Master Data Management Collaboration Server V10.1. Pendant l'exécution de l'étape de migration pour `type_import=5`, les erreurs associées à ces rapports sont émises sur la console. Ces erreurs sont associées à des types de travaux où CTGTODB est utilisé pour les importations et REPORTEXE pour les rapports. Lors d'une importation, ces rapports sont créés mais pas les travaux associés à ces rapports. Par conséquent, pour planifier ces rapports, des travaux doivent être créés manuellement dans InfoSphere MDM Collaboration Server V10.1. Les autres objets non pris en charge sont les services Web, les sélections avancées, les scripts d'implémentation et le contenu des magasins de documents par rapport à l'implémentation de Collaboration Server.

Vérification de votre environnement GDS migré

Vous devez vérifier votre environnement GDS migré après une opération de migration réussie.

Trois méthodes permettent de tester rapidement la cohérence des importations :

- Comparez les nombres d'articles, de catalogues et de spécifications dans la base de données. Par exemple, l'exécution de requêtes sur les tableaux CTG, CTR et SPC permet de vérifier le nombre de spécifications et de catalogues importés.
- Pour effectuer des tests complets, effectuer une réexportation depuis InfoSphere MDM Collaboration Server V10.1.x et exécutez une différence avec le fichier XSL transformé .zip depuis WebSphere Product Center V5.2.1.4.
- Vérifiez que les scénarios métier sont testés d'un bout à l'autre. Si des flux de travaux sont utilisés, réservez des articles et assurez-vous qu'ils fonctionnent comme prévu dans toutes les étapes du flux de travaux, puis quittez correctement le programme pour localiser des problèmes éventuels avec les scripts ou les attributs d'article.

Compilation et débogage de scripts

Vous pouvez utiliser ces méthodes simples pour compiler et déboguer des scripts.

- Pour utiliser des scripts compilés, assurez-vous que le fichier `common.properties` contient le paramètre suivant :
`script_execution_mode=compiled_only`
- Pour désactiver la compilation des scripts pris séparément, insérez la directive suivante au début du script :

`script_execution_mode=not_compiled`

Cependant, cette méthode n'est pas recommandée car elle donne lieu à une dégradation importante des performances. Si vous utilisez des scripts non compilés, vous pouvez définir ce paramètre sur **not_compiled** au lieu de désactiver la compilation au niveau des scripts.

Remarque : Si le paramètre du serveur dans le fichier `common.properties` est défini sur **not_compiled**, la compilation des scripts pris séparément ne peut pas être activée à l'aide de niveau de script.

L'utilisation d'une combinaison de scripts compilés et non compilés est déconseillée car elle altère les performances. Cependant, si vous devez utiliser cette combinaison, n'oubliez pas que, même si un script non compilé peut appeler des fonctions dans les autres scripts, l'inverse est impossible.

Erreurs communes de compilation de script

Lorsque vous utilisez des scripts compilés, un script peut être enregistré dans la console de scripts s'il se compile correctement. Si une erreur se produit, vérifiez le fichier `svc.out` dans le répertoire des journaux pour consulter la sortie complète du compilateur Java (javac) et les messages d'erreur. Voici quelques erreurs de compilation courantes :

- Une instruction `break` ou `return` à l'intérieur d'un bloc `forEach*Element()` ne se compile pas en raison d'une erreur "Code inaccessible". Pour résoudre cet incident :

```
forEachItemSetElement(itemSet, item)
{
    return item;
}
```

A remplacer par :

```
forEachItemSetElement(itemSet, item)
{
    if (item != null) {
        return item;
    }
}
```

Cela revient au même, mais convient mieux au compilateur.

- Si vous renvoyez une valeur à partir d'une fonction, vous devez renvoyer une valeur dans tous les cas. Cet exemple de code ne peut pas être compilé.

```
function sample() {
    var e = null;
    catchError (e) {
        // faire quelque chose...
        return "a string";
    } if (e != null) {
        reportError(...);
    }
}
```

Vous devez modifier la fonction car elle ne renvoie aucune valeur si une exception se produit dans le bloc `catchError`.

```
function sample() {
    var e = null;
    catchError (e) {
        // faire quelque chose...
```

```

return "a string";
}
if (e != null) {
reportError(...);
}
return null;
}

```

- Pour les problèmes de compilation plus sérieux, vous pouvez consulter les fichiers .java générés. Ces fichiers se trouvent dans un répertoire spécifié par le paramètre **tmp_dir** dans le fichier `common.properties`. La convention de dénomination des fichiers Java a été récemment modifiée afin d'inclure le nom du script ainsi qu'une séquence générée. Par exemple, `MyScript12345.java`.
- Le chemin d'accès absolu au script depuis le magasin de documents est indiqué sous forme de commentaire en haut de chaque fichier .java généré. Si vous mappez le magasin de documents au système de fichiers, vous pouvez exécuter la commande **grep** de manière réursive pour trouver le fichier .java qui correspond à un script.

Erreurs d'exécution communes

- Type d'argument non valide : vous avez sans doute transmis le type erroné d'argument à une fonction. Par exemple, une mappe de hachage au lieu d'une chaîne. Ce cas se produit également si WPC ne parvient pas à supposer le type correctement ; par conséquent, vous devrez sans doute utiliser des opérations de script comme `checkString()` pour être explicite.
- Types d'argument dans les comparaisons : si vous n'avez pas le même type de données des deux côtés d'un opérateur conditionnel comme `"=="`, `">"`, `"<"`, `"<="` etc, l'expression aura un résultat `false`. Aucun message d'erreur ne s'affichera en conséquence mais le code correspondant ne sera pas exécuté. Par exemple, le code suivant ne fonctionnera pas :

```

var id = "12345" ;
var my_id = item.getEntryAttrib(chemin d'accès à un attribut de type séquence) ;
if ( id == my_id ) {
// instructions qui doivent être exécutées mais ne le seront pas
}

```

Dans ce cas de figure, la solution consiste à utiliser de manière explicite le code suivant :

```

var id = "12345" ;
var my_id = checkString(item.getEntryAttrib(//attribut de type séquence),"") ;
if ( id == my_id ) {
// instructions à exécuter
}

```

- Analyse syntaxique XML : le code utilise le mode non compilé et également le mode compilé lorsqu'il est exécuté depuis le bac à sable :

```

new XmlDocument(xmlDoc) ;
foreachXmlNode("item") {
//faire le nécessaire
}

```

Toutefois, en mode compilé, si ce code est utilisé dans une fonction de bibliothèque de scripts appelée par plusieurs utilisateurs, les instructions dans `foreachXmlNode` ne seront pas exécutées. Aucun message d'erreur ne s'affiche. La solution palliative est :

```

var doc = new new XmlDocument(xmlDoc) ;
var xmlNode ; foreachXmlNode(doc, "item", xmlNode) {
//faire le nécessaire
}

```

Résolution des erreurs d'exécution

Pour résoudre les erreurs d'exécution sur le serveur d'applications, consultez le fichier `svc.out` dans le répertoire des journaux `appsrv`. L'examen des fichiers `exception.log` et `default.log` peut s'avérer utile dans certains cas. Grâce à la nouvelle convention de dénomination de fichier `.java`, vous pouvez aisément identifier le script défectueux. Le message d'erreur identifie également le numéro de ligne dans le fichier `.java` généré. Pour résoudre l'incident, affichez le fichier `.java` généré et accédez à la ligne où l'erreur d'exécution s'est produite. Le code Java généré inclut à présent le code de script actuel sous forme de commentaires, toutes les quelques lignes. Considérons, par exemple, la portion de code suivante issue d'un exemple de fichier `.java` généré :

```
// function checkIfPartyPartyTypeExist(party, partyType)
public static Object ScriptFunction__checkIfPartyPartyTypeExist

(HashMap hmContext, Object party, Object
partyType) throws Exception
{
// var bRet = false;
Object bRet = (java.lang.Boolean) Boolean.FALSE; //
var rootEntry = party.getRootEntryNode();
Object rootEntry = GenGetRootEntryNodeOperation.execute(hmContext , (IEntry) party);
// var entryNodes = rootEntry.getEntryNodes(getCatalogSpecName() +
"/Party Types/Party Type Code");
Object entryNodes = GenGetEntryNodesOperation.execute(hmContext ,
(EntryNode) rootEntry, (String)
BinaryOperation.execute(BinaryOperation.PLUS,
ScriptFunction__getCatalogSpecName(hmContext), "/Party
Types/Party Type Code"));
// var entryNodesSize = entryNodes.size();
Object entryNodesSize = (java.lang.Integer) GenSizeOperation.execute(hmContext ,
(HashMap) entryNodes);
```

Chapitre 4. Migration de scripts InfoSphere MDM Collaboration Server

Vous pouvez utiliser des scripts pour exécuter différentes fonctions dans IBM InfoSphere Master Data Management Collaboration Server. Les scripts offrent une grande souplesse sans compromettre l'efficacité de fonctionnement et la simplicité de développement.

Différences entre les scripts dans les diverses versions

La version de base utilisée pour IBM InfoSphere Master Data Management Collaboration Server Version 10.1.0 est WebSphere Product Center Version 5.0.1. Par conséquent, il existe des différences fonctionnelles et comportementales entre V5.2.1 et V10.1.0.

Les attributs de type de données *Flag* retournent des valeurs différentes dans V10.1.0. Dans la version 5.2.1, le type de données *Flag* renvoyait une chaîne TRUE/FALSE. Dans V10.1.0, les types de données *Flag* retournent l'équivalent d'une valeur booléenne Java qui peut avoir les valeurs true/false/null. Aussi, lorsque vous effectuez une comparaison avec un attribut flag, vous devez procéder comme suit :

```
if (flagValue==null)
```

Notez également que si la valeur d'un *Flag* est imprimée, par exemple, `out.write(flagValue);`, les valeurs imprimées seront null, false ou true.

Comme le code est compilé en Java lorsque vous utilisez le mode *compiled_only_mode*, une exception sera générée si les scripts IBM InfoSphere Master Data Management Collaboration Server utilisent des mots clés Java comme variables. Aussi, la compilation échouera si le script contient des variables qui sont déjà utilisées par InfoSphere MDM Collaboration Server dans le code généré, par exemple, *logger* et *hmContext*.

Une autre différence majeure existe entre les méthodes qui renvoient soit un *ItemSet* ou un *CategorySet* : dans la V5.2.1, si la méthode n'extrayait aucun résultat, elle renvoyait *null*. Dans V5.3.2, et donc V10.1.0, ces méthodes retournent un ensemble vide :

- Exemple de méthodes renvoyant un *ItemSet* :
 - `getCtgItemByAttributeValue(String sNodePath, String sValue)`
 - `getItemSetForCategory(Catalog ctg [, Boolean ordered])`
 - `getItemSetForCatalog()`
 - `getItemSetForUnassigned(CategoryTree ctr, boolean readOnly)`
 - `getItemSetForPrimaryKeys(Array pkeys, Boolean bOptimize)`
 - `getItemSetForSelection()`
 - `getSubset(ItemSet items, Integer start_point [, Integer end_point])`
- Here are the methods which return a *CategorySet*:
 - `getCategorySet([Boolean bReadOnly])`
 - `getDescendentCategorySetForCategory([Boolean bReadOnly])`

- getCategorySetByAttributeValue(String attribPath, Object attribValue, [Boolean bReadOnly])
- getCategorySetByPrimaryKey(String primaryKey, [Boolean bReadOnly])
- getCategorySetByLevel(Integer level, [Boolean bReadOnly])
- getCategorySetByFullPath(String[] fullPathPaths, String delimiter)
- getCategorySetByItemSecondarySpec(String specName)
- getCategorySetByStandAloneSpec(String specName)
- getLocationsAddedAvailability(CategoryTree locationHierarchy)
- getLocationsChangedToHaveData(CategoryTree locationHierarchy)
- getLocationsChangedToHaveNoData(CategoryTree locationHierarchy)
- getLocationsHavingChangedData(CategoryTree locationHierarchy)
- getAvailableLocations(Object locationOrCategoryTree)

Méthodes avec des signatures différentes dans V5.2.1 et V10.1.0

Le tableau suivant compare les opérations de script ajoutées à IBM InfoSphere Master Data Management Collaboration Server V10.1.0 Scripting Library :

Tableau 3. Comparaison des méthodes avec des signatures différentes dans V5.2.1 et V10.1.0

Opérations de la version 5.2.1	Opérations V10.1.0
checkString (String str, String defaultValue)	checkString (String str, String defaultValue [, Boolean trim])
getEntriesInfoXMLInStep (String stepName,ItemSet itemSet, HashMap hmAttrPaths[, int entriesCount])	getEntriesInfoXMLInStep (String stepName,HashMap itemSet, HashMap hmAttrPaths[, int entriesCount,String dateFormat])
userDefinedLogDeleteEntriesFor (IEntry entry)	userDefinedLogDeleteEntriesFor (Entry entry [, Entry category])
userDefinedLogAddEntry (Entry entry, [String log_message])	userDefinedLogAddEntry (Entry entry, [String log_message], [Entry category])
put (String sLkpTableName, String sStartKey, [String sEndKey,] String[] asValues)	put (String sLkpTableName, String sStartKey, String[] asValues)
createWebService (String name, String desc, String wsdlDocPath, String protocol, String implScriptPath, Boolean storeIncoming, Boolean storeOutgoing, Boolean deployed [,String style])	createWebService (String name, String implclass, String desc, String wsdlDocPath, String wsddDocPath, String protocol, String style, String implScriptPath, Boolean storeIncoming, Boolean storeOutgoing, Boolean deployed [, Boolean authRequired, Boolean skipRequestValidation, Boolean skipResponseValidation])
new\$CtgItem ([String sCtgName/Catalog ctg], [Boolean bRunEntryBuildScript], [Boolean bBuildNonPersisted])	new\$CtgItem (String sCtgName/Catalog ctg], [Boolean bRunEntryBuildScript], [Boolean bBuildNonPersisted], [Boolean bBuildEmptyEntryPicture])
printXML (String sTagName, String sValue, String sAttributes)	printXML (String sTagName, String sValue [, String sAttributes] [, boolean escape])
forEachXMLNode (String xPath [, XMLNode node])	forEachXMLNode ([XMLNode rootNode], String xPath [, XMLNode node])

Tableau 3. Comparaison des méthodes avec des signatures différentes dans V5.2.1 et V10.1.0 (suite)

Opérations de la version 5.2.1	Opérations V10.1.0
setHierarchyMapToExport (String sourceHierarchy, String destHierarchy)	setHierarchyMapToExport (String sourceHierarchy, String destHierarchy [,String sActionMode])
newUserDefinedLogEntry (Date date, Container container, Entry entry, String log)	newUserDefinedLogEntry (Date date, Container container, Entry entry, String log [, Entry category])
addObjectByNameToExport (String sEntityName[, String sObjectType])	addObjectByNameToExport (String sEntityName[, String sObjectType, [String sActionMode]])
userDefinedLogGetEntriesFor (IEntry entry)	userDefinedLogGetEntriesFor (Entry entry[, Entry category])
buildSpec (String specName, String specType)	buildSpec (String specName, String specType, [Spec specFileType])
addAllObjectsToExport ([String sObjectType])	addAllObjectsToExport ([String sObjectType, [String sActionMode]])
new Spec (String specName, String specType)	new Spec (String specName, String specType, [String specFileType])
getItemsInStepByAttribute (String stepName,String searchAttributePath,String searchAttrValue,String sSortColumn[,boolean isAscending,int startIndex, int endIndex])	getItemsInStepByAttribute (String stepName,String searchAttributePath,String searchAttrValue,String sSortColumn[,boolean isAscending,int startIndex, int endIndex,String categoryPKList])
setItemCategoryMapToExport (String sCatalog, String sHierarchy)	setItemCategoryMapToExport (String sCatalog, String sHierarchy [,String sActionMode])
getItemXMLRepresentation (Spec spec, boolean includePrimaryKeyValue[, boolean addNameSpace, String dateFormat])	getItemXMLRepresentation (Spec spec, boolean includePrimaryKeyValue[, String dateFormat])
forEachUserDefinedLogEntry (UserDefinedLog UDL, String[] entries [, Boolean bReturnMultipleEntries = true])	forEachUserDefinedLogEntry (UserDefinedLog UDL, [Entry e,] String[] logEntries [, Boolean bReturnMultipleLogEntries = true])
setEntryAttrib (String sAttribPath, Object sValue, [Boolean bDoChecks])	setEntryAttrib (String sAttribPath, Object sValue)
invokeSoapServer (String sURL, String sMethodName, Object[] aParamValues [,String[] aParamNames])	invokeSoapServer (String sURL, String sMethodName, Object[] aParamValues [,String[] aParamNames[, String userName, String password]])
importEnv (String sDocFilePath)	importEnv (String sDocFilePath, [bFromFileSystem])
getDocByPath(String sPath)	getDocByPath(String sPath , Boolean forceSync)
jmsGetQueueConnection()	jmsGetQueueConnection([String username], [String password])
sendEmail(String emailTos, String emailSubject, String emailBody, [Doc emailAttachment Doc[] emailAttachments])	sendEmail(String emailTos, String emailSubject, String emailBody, [Doc emailAttachment Doc[] emailAttachments, String ccList, String bccList])

Tableau 3. Comparaison des méthodes avec des signatures différentes dans V5.2.1 et V10.1.0 (suite)

Opérations de la version 5.2.1	Opérations V10.1.0
sendHTMLEmail(String emailTo, String emailSubject, String emailBody, [Doc emailAttachment Doc[] emailAttachments])	sendHTMLEmail(String emailTo, String emailSubject, String emailBody, [Doc emailAttachment Doc[] emailAttachments, String ccList, String bccList])
addEntryIntoColArea(Entry entry, String stepPath)	addEntryIntoColArea(Entry entry, String stepPath, [boolean waitForStatus])
getColAreaByName(String colAreaName)	getColAreaByName(String colAreaName,[boolean useCache])

Méthodes aux signatures différentes dans les versions 5.3.0 et 5.3.0.1

Le tableau suivant permet de comparer les méthodes aux signatures différentes dans WebSphere Product Center versions 5.3.0 et 5.3.0.1.

Tableau 4. Comparaison des méthodes aux signatures différentes dans les versions 5.3.0 et 5.3.0.1

Opérations de la version 5.3.0	Opérations de la version 5.3.0.1
getSpecByName (String name)	getSpecByName (String name, [Boolean bImmutable])

Pour plus d'informations sur les méthodes aux signatures différentes dans WebSphere Product Center version 5.3.0.1 et InfoSphere MDM Collaboration Server version 10.0.0, voir «Méthodes avec des signatures différentes dans V5.3.0.1 et V10.1.0».

Méthodes avec des signatures différentes dans V5.3.0.1 et V10.1.0

Le tableau suivant compare les opérations de script ajoutées à InfoSphere MDM Collaboration Server V10.1.0 Scripting Library :

Méthodes avec signatures différentes WebSphere Product Center V5.3.0.1 et IBMInfoSphere MDM Collaboration Server V10.1.0.

Tableau 5. Comparaison des méthodes avec des signatures différentes dans V5.3.0.1 et V10.1.0

Opérations de la version 5.3.0.1	Opérations V10.1.0
jmsGetQueueConnection()	jmsGetQueueConnection([String username], [String password])
createWebService(String name, String implclass, String desc, String wsdlDocPath, String wsddDocPath, String protocol, String style, String implScriptPath, Boolean storeIncoming, Boolean storeOutgoing, Boolean deployed [, Boolean authRequired])	createWebService (String name, String implclass, String desc, String wsdlDocPath, String wsddDocPath, String protocol, String style, String implScriptPath, Boolean storeIncoming, Boolean storeOutgoing, Boolean deployed [, Boolean authRequired, Boolean skipRequestValidation, Boolean skipResponseValidation])
getDocByPath(String sPath)	getDocByPath(String sPath , Boolean forceSync)

Tableau 5. Comparaison des méthodes avec des signatures différentes dans V5.3.0.1 et V10.1.0 (suite)

Opérations de la version 5.3.0.1	Opérations V10.1.0
sendEmail(String emailTos, String emailSubject, String emailBody, [Doc emailAttachment Doc[] emailAttachments])	sendEmail(String emailTos, String emailSubject, String emailBody, [Doc emailAttachment Doc[] emailAttachments, String ccList, String bccList])
sendHTMLEmail(String emailTos, String emailSubject, String emailBody, [Doc emailAttachment Doc[] emailAttachments])	sendHTMLEmail(String emailTos, String emailSubject, String emailBody, [Doc emailAttachment Doc[] emailAttachments, String ccList, String bccList])
addObjectByNameToExport (String sEntityName[, String sObjectType])	addObjectByNameToExport (String sEntityName[, String sObjectType, [String sActionMode]])
addAllObjectsToExport ([String sObjectType])	addAllObjectsToExport ([String sObjectType, [String sActionMode]])
setHierarchyMapToExport (String sourceHierarchy, String destHierarchy)	setHierarchyMapToExport (String sourceHierarchy, String destHierarchy [,String sActionMode])
setItemCategoryMapToExport (String sCatalog, String sHierarchy)	setItemCategoryMapToExport (String sCatalog, String sHierarchy [,String sActionMode])
addEntryIntoColArea(Entry entry, String stepPath)	addEntryIntoColArea(Entry entry, String stepPath, [boolean waitForStatus])
getColAreaByName(String colAreaName)	getColAreaByName(String colAreaName,[boolean useCache])

Méthodes avec des signatures différentes dans V6.0.0 et V10.1.0

Le tableau suivant compare les opérations de script ajoutées à IBM InfoSphere Master Data Management Collaboration Server V10.1.0 Scripting Library:

Méthodes avec des signatures différentes dans IBM InfoSphere MDM Server for PIM V6.0.0 et InfoSphere MDM Collaboration Server V10.1.0.

Tableau 6. Comparaison des méthodes avec des signatures différentes dans V6.0.0 et V10.1.0

Opérations de la version 6.0.0	Opérations V10.1.0
String exportEnv(EnvObjectList envObjList, String sDocFilePath)	String exportEnv(EnvObjectList envObjList, String sDocFilePath, [String sMapFilePath])

Méthodes présentes dans la version 5.2.1

La présente rubrique répertorie les méthodes uniquement disponibles dans WebSphere Product Center 5.2.1.

- populateFromPathValueMap (HashMap pathValueHashMap)
- getItemNode (String sPath)
- getCategoryPathsForInheritedValue (String sPathDelimiter, [Boolean bWithRoot])
- setTabular (String str)
- getRootItemNode ()

- setInheritanceTargets (String[][] containers)
- getItemNodes (String sPath)
- setWflStepAttributeGroups (String[]/AttrGroup[] attrGroups)
- getMappedAttributeGroups ()
- dummy ()
- deleteInheritanceRule ()
- getInheritanceTargets ()
- getItemNodePath ()
- getCtgItemAttribs ()
- getItemSetAndCountForSelection ()
- getItemNodeValue ()
- getEntryAttribModifier (String sAttribPath)
- getChangedAttributes (Entry secondEntry)
- massUpdate (Selection sel, String scriptPath, String parametersPath)
- isEntryNodeInheritedDataFromItem ()
- getChangedAttributesForMultiOccurrence (Entry oldEntry)
- new\$AdvancedSelection (Catalog catalog, String name, String expression)
- getInheritanceRuleByName (String sRuleName)
- getCtgItemRelatedItemInfo (Integer iItemId)
- disableInheritance ()
- setItemSetFetchCategorySpecificAttributes (Boolean b)
- setItemNode (String sPath)
- debug (String s)
- getEntryNodeInheritedValueSourceEntryUniqueID ()
- getSelectionRule ()
- getItemsInheritingDataForPath (String sAttribPath)
- setItemNodeRelationshipValueUsingItem (Item relatedItem)
- new\$InheritanceRule (Container container, String ruleName)
- getItemNodeChildren ()
- getCtgItemAttributesStatus ()
- getEntryNodeInheritedDataContainerName ()
- addSubNode (Node node)
- setItemNodeRelationshipValue (Catalog relatedItemCtg, String sRelatedItemPrimaryKey)
- reflattenAllInheritanceRules ()
- getEntryAttribModificationTime (String sAttribPath)
- removeAttributeGroup (String attributeGroupName)
- saveRule ()
- setItemNodeValue (Object value)
- addAttributeGroup (String attributeGroupName)
- setEntryAttributesFromXMLRepresentation (String xPath [,String dateFormat)
- resetAttributeGroupsToProcess ()

Méthodes disponibles uniquement dans la version 5.3.0.1

Cette rubrique répertorie les méthodes uniquement disponibles dans WebSphere Product Center v5.3.0.1.

- loadCatalog(String docStorePathForFileToLoad, Spec fileSpec, SpecMap specMap, String feedType [itm | icm | ctr])
- getEntryRelatedItemInfo(int itemID)

Méthodes migrées uniquement de V9.0.0 vers V10.1.0

Cette rubrique répertorie les méthodes disponibles uniquement dans IBM InfoSphere Master Data Management Collaboration Server V9.0.0 migrées vers V10.1.0.

- commitDB()
- new LinkHandler(String wflName)
- updateLinks(IEntrySet entrySet)
- New PathSelectionHandler(ICollaborationArea colArea)
- selectAndSetPath(IEntrySet entrySet,String propertyName)
- MultiMoveHandler(String wflName)
- moveEntries(IEntrySet entrySet,String stepPath)
- loadWF(String filePath)
- checkoutItem(String wfName, String itemID)
- checkoutCategory(String wfName, String categoryId,String categoryTreeName)
- new ChildCheckinHandler(String wflName)
- checkInChilds(IEntrySet entrySet, String stepPath)
- TIAttrVerifier(String wflName)
- verifyAttrGroupsAndSetExitValues(IEntrySet entrySet,String [] attrCollectionNames)
- new GDSXMLDocument(Doc doc/String str)
- getXMLNodeValue(Boolean bRequired)
- setXMLNodeValue (String sPath, String value)
- getGDSXMLNodeName()
- getXMLNodePath()
- getGDSXMLNode()
- addXMLNode(GDSXMLNode xmlNode [,String path])
- removeXMLNode(Object obj)
- getXMLNode(String path)
- cloneXMLNode()
- setValue(String path, String value)
- getGDSXMLNodes(String path)
- removeAustinNodes(String path)
- getXMLString()
- getGDSXMLNodeFromXMLNode(XMLNode xmlnode)
- getNewPublicationMessageIds()
- getNewPublicationMessageIdsByType(String[] messageTypes)
- getPublicationMessageIdsByDate(String fromDate, String toDate [,String recipientGLN])
- MessageItemHandler(String messageId)

- getTradeItems()
- sendConfirmation(String confirmationType,String[] messageIds)
- getLinks()
- changeState()
- getHierarchyTree()
- getAssociatedLinks(String gtin, Boolean fetchChildLinks)
- getInternalCode()
- setNodeType(String nodeType)
- sendCICConfirmationReasonCodes(String confirmationType,String MsgId,HashMap hmCICReasonCodeData)
- new CICStatus(String confirmationStatusCode, String confirmationStatusCodeDescription, String languageISOCode [,String additionalConfirmationStatusDescription] [,String correctiveAction] [,String expectedCorrectiveInformation])
- getParent(int index)
- getChild(int index)
- getNumberOfChildren()
- getNumberOfParents()
- getRootTradeItem()
- isLeaf()
- getLinkQuantity()
- createNotification(IBusinessObject iObj, String notifType, String userName)
- new TradeItemSS(String sGtin, String sGTINName, String sProductType, String sInternalCatCode, [String sDataPoolCatCode])
- getTradeItemSS(String sGTIN,String sTM,String sIP)
- setTradeItemAttributesSS(HashMap hmAttributePathsToValues)
- setTradeItemAttributeSS(String attributePath,String attributeValue)
- getTradeItemAttributeValueSS(String sAttrPath)
- toBoolean(String str)
- validateGTIN(String sGTIN)
- validateGLN(String sGLN)
- getGTINCategory(String sGTIN)
- getGTINTree(String sGTIN, String sProductType, Integer iLinkQuantity, String sLinkStatus)
- addChild(GTINTree tree)
- removeChild(String sGTIN)
- saveTradeItemSS()
- searchTradeItems(HashMap hmSearchParamsToValues)
- getTradeItemParentLinks()
- getTradeItemChildLinks()
- getModifiedTradeItems(String transactionType, Date startDate, Date endDate)
- getItems()
- setUDEXCategoryCode(String code)
- setInternalCategoryCode(String code)
- validateTradeItem()
- validateTradeItemAttribute(String sAttrPath)

- validateTradeItemAttributeGroup(String sAttributeGroup)
- new TradeItemDS(String sEANUCCCode, String sEANUCCCodeType, String sExtensionName, String sProductType,String sGPCCCode[,String sInternalCatCode ,String sGTIN ,String sGTINName])
- getExchangeSpecificAttribute()
- getTradeItemDS(String sEANUCCCode,String sTM,String sIP,[String sGTIN],[String recipientGLN])
- setTradeItemAttributesDS(HashMap hmAttributePathsToValues)
- setTradeItemAttributeDS(String attributePath,String attributeValue)
- getTradeItemAttributeValueDS(String sAttrPath)
- saveTradeItemDS()
- getAllTradeItemAttributes()
- searchTradeItemsDS(HashMap hmSearchParamsToValues)
- deleteTMIPVariant()
- deleteGTIN()
- isTradeItemCheckedOut()
- getWWREExchangeSpecificAttribute()
- isWWREInteroperable()
- addItemSS(HashMap sEntryMap, String strDate)
- synchronizeItemChangesSS(HashMap sEntryMap, String strDate, String operationType)
- publishNewItemSS(HashMap sEntryMap, HashMap tradingPartners,String strDate, String operationType)
- sendInitialLoadSS(HashMap sEntryMap, HashMap tradingPartners, String strDate, String operationType)
- addItemLinksForMassTransactionsSS(HashMap sEntryMap, String strDate)
- getCtgItemAttribValues()
- readGDSPropertiesSS(String sPropkey)
- searchTradeItems(HashMap hmSearchParamsToValues,String start, String end)
- processMassTransaction(String massJobID,HashMap hmSearchParamsToValues)
- HashMap setMassImportSS(boolean bImport)
- publishPublicationList(HashMap hmSearchParamsToValues)
- new TradeItemLink(String parentGTIN, String childGTIN [,Integer linkQuantity])
- setLinkAttributes(HashMap hmAttributeNameToValues [,String informationProvider])
- setLinkAttribute(String attributeName,String attributeValue [,String informationProvider])
- getTradeItemLink(String parentGTIN, String childGTIN)
- saveTradeItemLink()
- deleteTradeItemLink()
- getLinkAttributeValue(String sAttrName)
- getParentGTIN()
- getChildGTIN()
- new TradingPartner(String sGLN, String sPartnerName, String sCountry, String sGLNID, String itemType)

- new TradingPartner(String sGLN, String sPartnerName, String sCountry, String itemType)
- getTradingPartner(String sGLN)
- setPartnerAttributes(HashMap hmAttributePathsToValues)
- setPartnerAttribute(String attributePath,String attributeValue)
- saveTradingPartner()
- searchTradingPartners(HashMap hmSearchParamsToValues)
- getPartnerAttributeValue(String sAttrPath)
- createTransaction(IBusinessObject iObj, String txnType, String userName)
- isString()
- iterator()
- hasNext()
- next()
- isLangSpecificAttr(String sAttrPath)
- getGDSLanguages()
- flushGDSCache()
- new WWREMessageHandler(String wflName)
- sendMessage(IEntrySet entrySet,String messageType[, List transactionVOList])
- new MessageValidationResult(String sMsgID, String sMessage)
- new TradeItemValidationResult(String sKey)
- new AttributeValidationResult(String sKey)
- new ValidationResultObject(String sRuleName)
- addValidationResult(ValidationResultObject refValidationResultObject, String sMessage, String sStatus, String sType, Object objHolder)
- addValidationResultObject(AbstractValidationResult refAbstractValidationResult, ValidationResultObject refValidationResultObject)
- addTradeItemValidationResult(MessageValidationResult refMessageValidationResult, TradeItemValidationResult refTradeItemValidationResult)
- addAttributeValidationResult(TradeItemValidationResult refTradeItemValidationResult,AttributeValidationResult refAttributeValidationResult)
- getValidationResultObjectList(AbstractValidationResult refAbstractValidationResult)
- getTradeItemValidationResultList(MessageValidationResult msgValidationResult)
- getAttributeValidationResultList(TradeItemValidationResult tradeItemValidationResult)
- setValidationResultStatus(boolean bStatus)
- getValidationResultStatus()
- setValidationResultMessage(String message)
- setLocalizedValidationResultMessage(String messageID [,String[] arguments])
- addLocalizedValidationResultMessage(String messageID [,String[] arguments])
- getValidationResultMessage()
- setValidationResultHolder(Object holder)
- getValidationResultHolder()
- setValidationResultType(String sType)
- getValidationResultType()

- getTradeItemsInMsgItemDS(Item msgItem)
- getValidationResult(AbstractValidationResult refAbstractValidationResult)
- getMessageTypeName(Item msgItem)
- getDataPoolNameForMsgItem(Item msgItem)
- new TransactionHandlerDS()
- insertTransactionDS(ICollaborationArea colArea,IEntrySet entrySet,String stepPathString,String trasactionType,String status,String stepTrasition)
- updateTransactionDS(ICollaborationArea colArea,IEntrySet entrySet,String stepPathString,String trasactionType,String status,String stepTrasition)
- new StatusUpdaterDS()
- updateTradeItemGlobalStatusDS(IEntrySet entrySet,String status)
- updateTradeItemLocalStatusDS(IEntrySet entrySet,String status)
- updateStatusDS(IEntrySet entrySet,String status)
- new NotificationHandlerDS(String wflName)
- notifyUsersForActionDS(IEntrySet entrySet,String action)
- notifyUsersForActionWithStatusDS(IEntrySet entrySet,String action)
- new InternalSyncHandler()
- synchronize(IEntrySet entrySet)
- new EnrichmentHandler()
- canStartEnrichment(IEntrySet entrySet)
- startEnrichment(IEntrySet entrySet)
- new InitialLoadSaver()
- saveInitLoadEntries(IEntrySet entrySet)
- new MessageValidator()
- checkInitLoadAndSetExitVal(IEntrySet entrySet)
- new CategoryMappingValidator()
- verifyAndSetExitValues(IEntrySet entrySet)
- getTransactionDetails(ICollaborationArea colArea,IEntrySet entrySet,String stepPathString,String trasactionType,String status,String stepTrasition)
- runWFTTest(String ar1,String arg2, String arg3)
- createTestData(String ar1,String arg2)
- GDSCreateMassPublicationItems(IEntrySet entrySet)
- GDSDeleteMassPublicationItems(IEntrySet entrySet)
- GDSCheckFilters(IEntrySet entrySet)
- LogInboundTransaction(IItem msgItem,String transactionType, String senderGLN, String msgFilePath)
- TransactionHandlerSS()
- insertTransactionSS(ICollaborationArea colArea,IEntrySet entrySet,String stepPathString,String trasactionType,String status,String stepTrasition)
- updateTransactionSS(ICollaborationArea colArea,IEntrySet entrySet,String stepPathString,String trasactionType,String status,String stepTrasition)
- new StatusUpdaterSS()
- updateStatusSS(IEntrySet entrySet,String status)
- updateTradeItemGlobalStatusSS(IEntrySet entrySet,String status)
- updateTradeItemLocalStatusSS(IEntrySet entrySet,String status)
- new NotificationHandlerSS(String wflName)

- notifyUsersForActionSS(IEntrySet entrySet,String action)
- notifyUsersForActionWithStatusSS(IEntrySet entrySet,String action,String status)
- invokeWorkflowSS(HashMap entryMap, String sWFName)

Méthodes disponibles dans V5.3.2 uniquement migrées vers V10.1.0

Cette section répertorie les méthodes de WebSphere Product Center V5.3.2 uniquement migrées vers IBM InfoSphere Master Data Management Collaboration Server V10.1.0.

- publishEntriesToSrcContainer (EntrySet entrySet)
- setLdapOperation (LdapOperation operation)
- setViewableAttributeGroups (String subViewType, String[]/AttrGroup[] attrGroups, [String locationHierarchyName])
- isItemAvailableInLocation (Category location)
- isBinary ()
- reset ()
- getCurrentLocation ()
- getEntryInStep (String stepPath)
- deleteLookupTable (LookupTable lkp)
- assertFalse (Boolean condition, [String message])
- getEntryAttribValues (String sAttribPath)
- getSearchTemplateName ()
- getLdapObjectclasses ()
- getCategoryParentsUsingCache (CategoryCache cat_cache)
- getCtgItemLocationAttribsForKeys (Category location, Object[] aAttribPath [, String sDelimiter])
- getAddedAttributePathsNewEntry ([Category location])
- getItemRootEntryNodesHavingLocationData ()
- getEntryChangedDataSinceLastSave ()
- getDate (int columnIndex)
- addLdapAttribute (LdapAttribute attribute)
- getLocationPathForInheritedValue (String delimiter)
- getModifyLocationHierarchyAvailability (String locationHierarchyName)
- makeItemAvailableInLocations (Category[] locations, [Boolean bRecursive])
- encodeUsingCharset (String charset)
- getLocationsAddedAvailability (CategoryTree locationHierarchy)
- getEntryRelationshipAttrib (String sAttribPath)
- new\$LdapObjectclass (String objclass)
- getWflStepView (String subViewType, [String locationHierarchyName])
- getLocationsChangedToHaveData (CategoryTree locationHierarchy)
- setLdapDistinguishedName (LdapAttribute dn)
- getLocation ()
- xmlDocToString ()
- getAvailableLocations (Object locationOrCategoryTree)
- writeFileUsingReader (Reader r)

- `setEditableAttributeGroups (String subViewType, String[]/AttrGroup[] attrGroups, [String locationHierarchyName])`
- `getDynamicSelectionQueryString ()`
- `setActionModeToExport (String sActionMode)`
- `addLdapEntry (LdapEntry entry)`
- `getLdapOperation ()`
- `getCategory (int columnIndex)`
- `getHierarchy (int columnIndex)`
- `removeLocationSpecificData (CategoryTree ctr)`
- `parseDate (String value, String format[,Locale locale])`
- `new$SearchQuery (String queryString)`
- `getStepsForEntry (Entry entry)`
- `getWsddDocPath ()`
- `setRequiredAttributeGroups (String subViewType, String[]/AttrGroup[] attrGroups, [String locationHierarchyName])`
- `getWflStepViews ()`
- `startExportByName (String sName)`
- `getErrorsForLocation ()`
- `getInt (int columnIndex)`
- `parseXMLNodeWithNamespace (String sXMLSubPath)`
- `initializeKeyValueMapping (Object[] aKeyset)`
- `assertTrue (Boolean condition, [String message])`
- `getCtgItemAttribsForKeys (Object[] aAttribs [, String sDelimiter])`
- `getNodeDisplayName ([Locale locale])`
- `getLocationErrors ()`
- `getLdapEntries ()`
- `getRequiredAttributeGroups ([String subViewType], [String locationHierarchyName])`
- `getEntryChangedData (Entry oldEntry, Entry newEntry)`
- `getLocaleNode (Locale locale)`
- `locationHasData (Category location)`
- `execute ()`
- `makeItemUnavailableInLocations (Category[] locations, [Boolean bRecursive])`
- `getLdapAttributes ()`
- `assertNotNull (Object obj, [String message])`
- `getLocationsHavingChangedData (CategoryTree locationHierarchy)`
- `getLocationsHavingData (Object locationOrCategoryTree)`
- `getSpec (int columnIndex)`
- `getItemLocationAttrib (Category location, String sAttribPath)`
- `runQuery (String qryString)`
- `getLdapAttributeValue ()`
- `defineLocationSpecificData (CategoryTree ctr, Spec spc, AttrGroup[] inhAttrGrps)`
- `new$SearchTemplate (String name, Array attrGroupNames, Container container, String desc, [String colAreaName, String stepPath])`
- `getBoolean (int columnIndex)`

- `setItemLocationData` (Item item, CategoryTree locationTree, String[] locFullPaths, String delimiter, Boolean rootIncluded[, Boolean append])
- `setImplclass` (String implclass)
- `isInheriting` (Category location, String sAttribPath)
- `new$LdapEntry` ()
- `getLdapAttributeType` ()
- `setXMLNodeValue` (String sPath, String value)
- `parseLDIFFFile` (String filename)
- `dropEntry` (Entry entry)
- `new$DynamicSelection` (String selectionName, String queryString)
- `makeItemAvailableInLocation` (Category location, [Boolean bRecursive])
- `getCategoryChildrenUsingCache` (CategoryCache cat_cache)
- `isDefined` (int columnIndex)
- `getEntryNodeType` ()
- `setAuthRequired` (Boolean authRequired)
- `flushScriptCache` ()
- `getLocationsRemovedAvailability` (CategoryTree locationHierarchy)
- `getScriptExecutionMode` ()
- `getModifiedAttributePathsOldEntry` ([Category location])
- `copySearchItemData` (Item searchItem[, Boolean append])
- `copySearchItemLocationTreeData` (Item searchItem, CategoryTree locationTree[, Boolean append])
- `createLDIFFFile` (String docstore_filename, LdapEntrySet entrySet)
- `getLong` (int columnIndex)
- `decodeUsingCharset` (String charset)
- `moveEntryToNextStep` (Entry entry, String stepPath, String exitValue)
- `isAuthRequired` ()
- `getSearchTemplateByName` (String name)
- `getItemRootEntryNodeForLocation` (Category location)
- `getViewableAttributeGroups` ([String subViewType], [String locationHierarchyName])
- `moveEntryToColArea` (Entry entry, String destColAreaName)
- `getCtgViewType` ()
- `getPossibleEntryNodeValues` ()
- `getItemUsingEntryRelationshipAttrib` (String sAttribPath)
- `javaArrayFromScriptArray` (Array scriptArray, String type)
- `getXMLNode` (String nodePath)
- `assertEquals` (Object expectedValue, Object actualValue, [String message])
- `addLdapObjectclass` (LdapOperation objectClass)
- `new$LdapEntrySet` ()
- `setNodeIndexed` (Boolean)
- `isNodeIndexed` ()
- `getProductCenterURL` - Renvoie la propriété `trigo_web_url` définie dans le fichier `common.properties`, qui détient l'URL absolue, notamment le numéro de port, du site Web où les utilisateurs doivent faire pointer leurs navigateurs pour accéder à cette instance d'InfoSphere MDM Collaboration Server.

- `assertNull (Object obj, [String message])`
- `setXMLNodeValues (String sPath, String[] values)`
- `setModifyLocationHierarchyAvailability (String locationHierarchyName, Boolean canModifyAvailability)`
- `getFlatEntryNodesOf (EntryNode en)`
- `checkoutEntry (Entry entry, [String stepPath], [boolean waitForStatus])`
- `getLocationHierarchyNames ([Boolean canModifyAvailability])`
- `getGlobalErrors ()`
- `getItem (int columnIndex)`
- `getModifiedAttributePathsNewEntry ([Category location])`
- `deleteCtgView (CtgView ctgView)`
- `getLocationsChangedToHaveNoData (CategoryTree locationHierarchy)`
- `new$searchSelection (Catalog catalog, String name)`
- `getDouble (int columnIndex)`
- `moveCursor (int position)`
- `getXMLNodes (String sPath)`
- `getLdapObjectclass ()`
- `new$LdapAttribute (String attributeType, Object attributeValue [, Boolean isBinary, Boolean isExternal])`
- `getImplclass ()`
- `getAllUsers ()`
- `getEditableAttributeGroups ([String subViewType], [String locationHierarchyName])`
- `getXMLNodeValues (String nodePath [, Boolean bRequired])`
- `getFloat (int columnIndex)`
- `makeItemUnavailableInLocation (Category location, [Boolean bRecursive])`
- `deleteSearchTemplate ()`
- `getDeletedAttributePathsOldEntry ([Category location])`
- `isExternal ()`
- `setInheriting (Category location, String sAttribPath, [Boolean flag])`
- `setWsddDocPath (String wsddDocPath)`
- `getString (int columnIndex)`
- `setItemLocationAttrib (Category location, String sAttribPath, Object sValue)`
- `scriptArrayFromJavaArray (OneDimensionalJavaArray)`
- `setEntryAttribValues (String sAttribPath, Object[] sValues)`
- `getLdapDistinguishedName ()`
- `executeInBackground (String selectionName)`
- `getLocationForErrors ()`
- `disableEntryValidation(Entry collabAreaEntry, String workflowStepAtPath, Boolean disableValidation)`
- `isEntryNew()`
- `item.getCtgItemRelatedItemInfo()`

Remarque :

- Cette opération de script est obsolète dans WebSphere Product Center v5.2.1.

- Cette opération de script a été remplacée par **entry::getEntryRelationshipAttrib(string attribpath)** dans InfoSphere MDM Collaboration Server version 9.0.0.
- moveUserToOrganization(Category srcOrg,Category dstOrg)
- copyUserToOrganizations(Category[] organizations)
- removeUserFromOrganization(Category organization)
- invocationCacheGet(String key)
- invocationCachePut(String key, Object value)
- invocationCacheRemove(String key)
- invocationCacheClear()
- setDynamicSelectionQueryString(String queryString)
- getExactAttributePath()

Méthodes de V6.5.0 uniquement migrées vers V10.1.0

Cette rubrique indique une méthode de IBM InfoSphere Master Data Management Collaboration Server V6.5.0 uniquement qui est migrée vers V10.1.0.

- String ExcelCell::getExcelCellEncoding()

Modifications du comportement des opérations de script entre V5.2.1 et V10.1.0

Cette rubrique répertorie les modifications du comportement des opérations de script dans WebSphere Product Center V5.2.1 et IBM InfoSphere Master Data Management Collaboration Server V10.1.0 Scripting Libraries.

setEntryNodeValue()

Comportement dans la version 5.2.1

1. Crée un nouveau noeud s'il n'en existe pas.
2. Renvoie la valeur qui a été définie.

Comportement dans V10.1.0

1. Génère des exceptions si le noeud n'est pas déjà présent. Vous devez créer un nouveau noeud à l'aide de la méthode `entry.setEntryAttrib()`.
2. Renvoie 1/0/-1 en fonction de l'action entreprise (succès, aucune modification, erreur).

getEntryNode()

Comportement dans la version 5.2.1

1. Lorsqu'un(e) nouvel(le) article/catégorie est créé(e), une occurrence du noeud MultiAttribute est présente par défaut.
`itemNode.getEntryNode("/SpecForMigTest/Multi#0/Sub-Multi#0")`

Renvoie le noeud

2. Si le chemin d'accès spécifié n'est pas trouvé, la version 5.2.1 le crée et renvoie le noeud. Renvoie Null si le chemin d'accès n'est pas valide.

Comportement dans V10.1.0

1. V10.1.0 émet une exception si le chemin n'est pas présent.
2. V10.1.0 ne crée pas de noeud.

setEntryNode()

Comportement dans la version 5.2.1

1. La version 5.2.1 crée et renvoie le noeud, autrement, elle renvoie la valeur Null si le chemin d'accès n'est pas valide.

Comportement dans V10.1.0

1. V10.1.0 émet une exception si le chemin n'est pas présent.

getEntryNodeChildren()

Comportement dans la version 5.2.1

1. La version 5.2.1 renvoie les enfants actuels du noeud de groupe.

```
itemNode = item.getRootEntryNode(); node = itemNode.setEntryNode("/SpecForMigTest/Multi");
```

output:

SpecForMigTest/Multi/Sub-Multi

Comportement dans V10.1.0

1. V10.1.0 retourne les occurrences de Multi et non pas son enfant, par exemple,

```
itemNode = item.getRootEntryNode(); node =  
itemNode.setEntryNode("/SpecForMigTest/Multi");
```

output: /SpecForMigTest/Multi#0 /SpecForMigTest/Multi#1

getEntryNodes()

Comportement dans la version 5.2.1

1. Dans la version 5.2.1, les enfants sont également renvoyés car les chemins d'accès des enfants contiennent le chemin d'accès spécifié dans l'opération de script.

```
itemNode = item.getRootEntryNode(); nodes =  
itemNode.getEntryNodes("/SpecForMigTest/Multi"); out.println(nodes);
```

Comportement dans V10.1.0

1. V10.1.0 ne retourne pas d'enfant.

getEntryNodeParent()

Comportement dans la version 5.2.1

1. La version 5.2.1 renvoie les parents actuels.

```
itemNode = item.getRootEntryNode(); node =  
itemNode.getEntryNodes("/SpecForMigTest/Multi#0/Sub-Multi#0");  
out.println("Node:: " + node[0].getEntryNodePath());  
out.println("Node Parents:: "  
+ node[0].getEntryNodeParent().getEntryNodePath());
```

Output:

Node:: SpecForMigTest/Multi/Sub-Multi Node Parents:: SpecForMigTest/Multi

Comportement dans V10.1.0

1. V10.1.0 retourne le noeud de groupe au lieu du parent.

```
itemNode = item.getRootEntryNode(); node =  
itemNode.getEntryNodes("/SpecForMigTest/Multi#0/Sub-Multi#0");  
out.println("Node:: " + node[0].getEntryNodePath());  
out.println("Node Parents:: "  
+ node[0].getEntryNodeParent().getEntryNodePath());
```

Sortie :

Node:: SpecForMigTest/Multi/Sub-Multi Node Parents::
SpecForMigTest/Multi/Sub-Multi

getFlatEntryNodes()

Comportement dans la version 5.2.1

1. Cette méthode renvoie un tableau des EntryNodes non hiérarchiques de cette entrée.

Comportement dans V10.1.0

Cette méthode générait des exceptions dans les versions précédentes qui étaient provoquées par le script qui effectuait un appel vers **getEntryNodeValue** sur le **entryNode** renvoyé par **getFlatEntryNodes**. Cette exception est provoquée, car avec le formulaire sérialisé, nous disposons d'un nouvel **entrynode** renvoyé par **getFlatEntryNodes** : le noeud *directory*. Pour les attributs individuels qui apparaissent plusieurs fois, un élément **entrynode** supplémentaire est renvoyé et son **node::type** est *STRING* ou le type de l'attribut. Le script tente d'obtenir la valeur et de la signaler. Un correctif a été fourni dans la version 5.3.2. A présent, une fois que **getFlatEntryNodes** a renvoyé les noeuds, une nouvelle méthode **getEntryNodeType()** vérifie son type. Si le type s'avère être *V*, seul le noeud de l'entrée est traité. Le prototype de la méthode est le suivant :

```
<prototype>
  <![CDATA[String EntryNode::getEntryNodeType()]]>&gt;
</prototype>
<description>
  <![CDATA[return "V" = valeur, "G" = Groupement ou niveau supérieur du répertoire de spécifications,
  "M" = répertoire multiple (contient plusieurs occurrences
  des valeurs ou groupements)]]>&gt;
</description>
```

L'exemple suivant illustre l'implémentation.

```
ctgl = getCtgByName( "<catalog name>" );
itm1 = ctgl.getCtgItemByPrimarykey( "<clé primaire de l'article>" );
ents1 = itm1.getFlatEntryNodes();

for( i=0; i<ents1.size(); i++ )
{
  var val1 = "----";
  var ent1 = ents1[i];

  var ePath1 = ent1.getEntryNodeExactPath();
  out.write( ePath1 + " " );

  var type1 = ent1.getEntryNodeType();
  out.write( type1 + " " );

  if( type1 == "V" )
  {
    val1 = "" + ent1.getEntryNodeValue();
  }
  else
  {
    val1 = "" + ent1.getEntryNodeChildren().size();
  }
  out.writeln( val1 );
}
```

setTypeToExport(String sObjectType)

Comportement dans la version 5.2.1

- COLLABORATION_AREA_CONTENT est une valeur acceptable pour sObjectType

Comportement dans V10.1.0

- COLLABORATION_AREA_CONTENT a été retiré de la liste des valeurs acceptables pour sObjectType

addObjectByNameToExport (String sEntityName[, String sObjectType, [String sActionMode]])

Comportement dans la version 5.2.1

- COLLABORATION_AREA_CONTENT est une valeur acceptable pour sObjectType

Comportement dans V10.1.0

- COLLABORATION_AREA_CONTENT a été retiré de la liste des valeurs acceptables pour sObjectType

addAllObjectsToExport ([String sObjectType, [String sActionMode]])

Comportement dans la version 5.2.1

- COLLABORATION_AREA_CONTENT est une valeur acceptable pour sObjectType

Comportement dans V10.1.0

- COLLABORATION_AREA_CONTENT a été retiré de la liste des valeurs acceptables pour sObjectType

Modifications du comportement des opérations de script entre V5.3.0.1 et V10.1.0

Cette rubrique répertorie les modifications dans le comportement des opérations de script dans WebSphere Product Center V5.3.0.1 et IBM InfoSphere Master Data Management Collaboration Server V10.1.0 Scripting Libraries.

setContainerProperties()

Comportement dans la version 5.3.0.1

- Les propriétés spécifiées dans le mappe de hachage sont définies pour le conteneur en question.

Comportement dans V10.1.0

- Les propriétés spécifiées dans le mappe de hachage sont définies pour le conteneur en question. La clé de mappe de hachage "SCRIPT_NAME" est désormais obsolète et a été remplacée par "POST_SCRIPT_NAME".

getCategoryAttrib()

Comportement dans la version 5.3.0.1

- Renvoie la valeur de l'attribut sAttribPath (nom_spécification/nom_attribut) de cette catégorie.

Comportement dans V10.1.0

- Renvoie la valeur de l'attribut sAttribPath (nom_spécification/nom_attribut) de cette catégorie, uniquement lorsque l'attribut est inclus dans la collection de vues ou d'attributs. Sinon, la valeur null est renvoyée.

getEntryAttrib()

Comportement dans la version 5.3.0.1

- Renvoie la valeur de l'attribut sAttribPath (nom_spécification/nom_attribut) de cette entrée.

Comportement dans V10.1.0

- Renvoie la valeur de l'attribut sAttribPath (nom_spécification/nom_attribut) de cette entrée, uniquement lorsque l'attribut est inclus dans la collection de vues ou d'attributs. Sinon, la valeur null est renvoyée.

getCtgItemAttrib()

Comportement dans la version 5.3.0.1

- Renvoie la valeur de l'attribut sAttribPath (nom_spécification/nom_attribut) de cet article.

Comportement dans V10.1.0

- Renvoie la valeur de l'attribut sAttribPath (nom_spécification/nom_attribut) de cet article, uniquement lorsque l'attribut est inclus dans la collection de vues ou d'attributs. Sinon, la valeur null est renvoyée.

removeNode()

Comportement dans la version 5.3.0.1

- Renvoie une valeur booléenne.

Comportement dans V10.1.0

- Renvoie une valeur vide.

setTypeToExport()

Comportement dans la version 5.3.0.1

- Définit le type d'objet à exporter.

Comportement dans V10.1.0

- Définit le type d'objet à exporter. COLLABORATION_AREA_CONTENT et INHERITANCE_RULES sont retirés de la liste des valeurs acceptables pour sObjectType.

addObjectByNameToExport()

Comportement dans la version 5.3.0.1

- COLLABORATION_AREA_CONTENT est une valeur acceptable pour sObjectType.

Comportement dans V10.1.0

- COLLABORATION_AREA_CONTENT est retiré de la liste des valeurs acceptables pour sObjectType.

addAllObjectsToExport()

Comportement dans la version 5.3.0.1

- COLLABORATION_AREA_CONTENT est une valeur acceptable pour sObjectType.

Comportement dans V10.1.0

- COLLABORATION_AREA_CONTENT est retiré de la liste des valeurs acceptables pour sObjectType.

disableContainerProcessingOptions(String[] options)

Comportement dans la version 5.3.0.1

- Désactive les options de traitement indiquées.

Comportement dans V10.1.0

- Désactive les options de traitement indiquées. Autres valeurs possibles pour le paramètre Options : ENTRY_BUILD_SCRIPT et UNIQUE_VALIDATION.

Modifications du comportement des opérations de script entre V6.0.0 et V10.1.0

Cette rubrique répertorie les modifications du comportement des opérations de script dans IBM InfoSphere Master Data Management Collaboration Server V6.0.0 et V10.1.0 Scripting Libraries.

exportEnv ()

Comportement dans la version 6.0.0

- Exporte les objets InfoSphere MDM Collaboration Server spécifiés dans envObjList au chemin du magasin de documents spécifié. La variable *sDocFilePath* contient le chemin du fichier .zip qui sera exporté dans le magasin de documents. Cette méthode renvoie le journal sous la forme d'une chaîne.

Comportement dans V10.1.0

- Exporte les objets InfoSphere MDM Collaboration Server spécifiés dans envObjList au chemin du magasin de documents spécifié. La variable *sDocFilePath* contient le chemin du fichier .zip qui sera exporté dans le magasin de documents. Si *sDocFilePath* est indiqué sous la forme "file:/// " suivi du chemin sur le système de fichiers, le fichier .zip sera

exporté directement dans le système de fichiers, et non dans le magasin de documents. Le paramètre facultatif *sMapFilePath* est le chemin du nom du fichier de mappage sur le système de fichiers qui fournit l'objet aux mappages de nom de fichier pour les objets en cours d'exportation. Cette méthode renvoie le journal sous la forme d'une chaîne.

Différences des API Java entre les diverses versions

La version de base utilisée pour l'API Java dans IBM InfoSphere Master Data Management Collaboration Server Version 10.1.0 est WebSphere Product Center Version 5.3.2.

Méthodes et énumérations API Java

La présente rubrique répertorie les nouvelles méthodes et énumérations ajoutées aux interfaces existant dans IBM InfoSphere MDM Server for PIM version 6.0.0, groupe de correctifs 5, version 9.1.0 et IBM InfoSphere Master Data Management Collaboration Server version 10.0.

AttributeDefinition.java

enum: AttributeDefinition.Type: RICH_TEXT et EXTERNAL_CONTENT_REFERENCE ajoutés

method: public int getId()

AttributeDefinitionProperty.java

enum AttributeDefinitionProperty.Name: GROUP et LOCALE_NODE obsolètes

enum AttributeDefinitionProperty.Name: RICH_TEXT ajouté

AttributeInstance.java

method: public void setValueUsingJSON(Object value)

AttributeOwner.java

method: public AttributeInstance getAttributeInstance(int[] nodeIdArray, int[] indexArray, boolean bFollowInheritance, boolean createOccurrencesIfNeeded)

Item.java

method: public OrderedJSONObject toJSON(boolean includeAttributeType)

method: public void addStatusAndAvailability(OrderedJSONObject itemJSONObj)

Catalog.java

method: public PIMCollection<Item> getItems(int[] itemIds)

method: public PIMCollection<Item> getItems(Object[] itemIds)

method: public void setIgnoreCache(boolean ignoreCache)

method: public boolean getIgnoreCache()

method: public boolean isCached()

method: public void setCached(boolean setCached)

method: public OrderedJSONObject toJSON()

CatalogManager.java

method: public Catalog getCatalog(String catalogName, boolean returnItem)

method: public Catalog getCatalog(int catalogId)

method: public Catalog getCatalog(int catalogId, boolean returnItem)

CategoryCollaborationArea.java

method: public CollaborationCategory createCollaborationCategory()

CollaborationCategory.java

enum: CollaborationCategory.SaveResult

method: public SaveResult getSaveResult()

method: public Collection<SecondarySpec> getItemSecondarySpecs()

method: public OrderedJSONObject toJSON(boolean includeAttributeType)

method: public void addStatusAndAvailability(OrderedJSONObject

collaborationCategoryJSONObject)

method: public void addItemSecondarySpec(SecondarySpec spec, Collection<Catalog> catalogs, boolean addToChildCategories, boolean addAcrossMapping)

method: public void addSecondarySpec(SecondarySpec spec)

method: public void removeItemSecondarySpec(SecondarySpec spec)

CollaborationItem.java

method: public OrderedJSONObject toJSON(boolean includeAttributeType)

method: public void addStatusAndAvailability(OrderedJSONObject collaborationItemJSONObject)

CollaborationStep.java

method: public CollaborationObject getCollaborationObject(int id)

enum ValidationError.Type

Les éléments suivants sont ajoutés : TOTAL_DIGITS, MIN_EXCLUSIVE, MIN_INCLUSIVE, MAX_EXCLUSIVE, MAX_INCLUSIVE

Context.java

method: SearchQuery createSearchQuery(String queryString, Collection<Item> items, Catalog catalog)

method: public String setSavepoint()

method: void rollbackToSavepoint(String name)

method: public SearchHelper getSearchHelper()

method: public HistoryManager getHistoryManager()

method: public MessageBundle getMessageBundle()

method: public MessageBundle getMessageBundle(Locale locale)

Category.java

enum: Category.SaveResult

method: public Collection<String[]> getFullDisplayPaths()

method: public Collection<Integer> getSecondarySpecIdsForChildren()

method: public OrderedJSONObject toJSON(boolean includeAttributeType)

method: public void addStatusAndAvailability(OrderedJSONObject categoryJSONObject)

Hierarchy.java

method: public Category createCategory()

method: public PIMCollection<Category> getCategories(int[] categoryIds)
method: public PIMCollection<Category> getCategories(Object[] categoryIds)
method: public OrderedJSONObject toJSON()

HierarchyManager.java

method: public Hierarchy getHierarchy(int hierarchyId)

OrganizationManager.java

method: public OrganizationHierarchy getOrganizationHierarchy(int id)

SearchQuery.java

enum SearchQuery.ColumnType : LOGENTRY added enum:
SearchQuery.SearchQueryOp

SearchResultSet.java

method: public UserDefinedLogEntry getUserDefinedLogEntry(int columnIndex) throws PIMSearchException

SelectionManager.java

method: public Selection getSelection(int selectionId)

Spec.java

method: public AttributeDefinition getAttributeDefinition(int attributeDefinitionId)

SpecManager.java

method: public Spec getSpec(int specId)

EnvironmentExporter.java

method: public String export(ExportList exportList, String documentPath, String mappingPath)

method: public String export(ExportList exportList, String documentPath, boolean

checkForRequisites, String mappingPath)

method: public String exportAll(String documentPath, String mappingPath)

Nouvelles interfaces de programme d'application (API) Java

La présente rubrique répertorie les nouvelles interfaces introduites dans IBM InfoSphere MDM Server for PIM version 9.1.0 et IBM InfoSphere Master Data Management Collaboration Server version 10.0.

- Filtrage
- PIMObject
- HistoryManager
- CMSContentURN
- CMSInstance
- CMSManager
- AuthorizationHelper
- SavedSearch
- SearchHelper
- LeftPaneDataObject
- LeftPaneDataObjectFactory
- RichTextEditor
- MessageBundle

- PackagingTool

Optimisation de la base de données

Vous devez optimiser la base de données en augmentant la taille de pile si le serveur de base de données utilisé est IBM DB2.

Avant de commencer

Pour optimiser les paramètres de la base de données DB2, vous devez d'abord vous connecter en tant qu'administrateur DB2.

Procédure

Exécutez les commandes suivantes pour augmenter la taille de pile.

```
db2 update database configuration for database <nom_base_de_données>
using applheapsz 8192
db2 update database configuration for database <nom_base_de_données>
using app_ctl_heap_sz 8192
db2 update database configuration for database <nom_base_de_données>
using LOGFILSIZ 4000
```

où *nom_base_de_données* est le nom réel de l'instance de base de données DB2. Vous pouvez connaître les valeurs en cours de ces paramètres en utilisant les commandes suivantes :

```
db2 get dbm cfg
db2 get db cfg for <nom_base_de_données>
```

Remarque : Les valeurs des commandes DB2 ci-dessus sont fournies à titre indicatif. Vous devez demander les valeurs appropriées pour votre environnement à l'administrateur de la base de données.

Transformation des métadonnées du modèle de données

Vous devez transformer les métadonnées de WebSphere Product Center version 5.2.1.4 à utiliser dans IBM InfoSphere Master Data Management Collaboration Server version 9.1.0.

Pourquoi et quand exécuter cette tâche

WebSphere Product Center version 5.2.1.4 et InfoSphere MDM Collaboration Server version 9.1.0. ne stockent pas les données et ne traitent pas les modèles de données de la même manière. Pour garantir la conformité des données aux normes d'InfoSphere MDM Collaboration Server version 9.1.0, vous devez transformer les données.

Pour transformer les données, vous pouvez utiliser le script `migrateCompanyZip.sh`. Ce script lit le fichier compressé généré par la procédure d'exportation, recherche les problèmes d'incompatibilité des modèles de données dans le fichier compressé généré par la procédure d'exportation et crée un fichier compressé portant le même nom que vous pouvez exporter dans une instance d'InfoSphere MDM Collaboration Server version 9.1.0. Le script shell utilise en interne des transformations XSL lancées en fonction du contenu du fichier compressé.

Lorsque vous exécutez ce script, vous pouvez également indiquer un fichier XML qui définit les attributs RTS pour la société cible. Cette fonction a été ajoutée pour améliorer les performances.

Avant de migrer les données, vérifiez que tous les attributs que vous souhaitez inclure dans les recherches sont définis en tant qu'attributs RTS. Si un attribut est modifié pour devenir un attribut RTS après la migration via la console des spécifications, tous les articles migrés ou créés avant cette modification doivent être une nouvelle fois sauvegardés pour avoir le type RTS. Si le nombre de ces articles est très élevé, cette tâche peut s'avérer très fastidieuse. Il est conseillé de rechercher ces attributs avant la migration.

Procédure

Exécutez le script `migrateCompanyZip.sh`. Ce script se trouve dans le répertoire `$TOP/bin`.

```
migrateCompanyZip.sh [--debug=yes] [--verbose=yes]
[--param=<filename>] --out=<output directory> <input zip>
```

Dans cet exemple, `migrateCompanyZip.sh` transforme le contenu du fichier `companySpecs.zip` et utilise le fichier `params.xml` pour identifier tous les attributs RTS (RunTime Searchable) dans la société cible.

```
$TOP/bin/migrateCompanyZip.sh --debug=yes --verbose=yes
--param=$TOP/params.xml --out=$TOP/out $TOP/input/companySpecs.zip
```

où `$TOP` indique le répertoire d'installation d'InfoSphere MDM Collaboration Server.

Corrections par le transformateur XSLT

Le transformateur XSLT est conçu pour corriger certaines erreurs.

Modifications des spécifications

IBM InfoSphere Master Data Management Collaboration Server version 9.1.0 ne prend pas en charge ce qui suit :

- Type de noeud SpecNode PERIOD - Si vous effectuez une migration depuis WebSphere Product Center V5.2.1.x et utilisez le type PERIOD SpecNode, l'utilitaire convertira ce dernier au format GROUP.
- Noeud SpecNode avec la valeur d'attribut NULL. Par exemple, les attributs SpecNode MAX_INCLUSIVE, SCALE, MAX_EXCLUSIVE et MIN_EXCLUSIVE ne peuvent pas avoir de valeur NULL. L'utilitaire de transformation supprime ces propriétés d'attribut.

Modifications du flux de travaux

InfoSphere MDM Collaboration Server version 9.1.0 ne possède pas le type StepType PARTIAL_UNDO. Le remplacement équivalent INTERIM_CHECKOUT. Ainsi, l'utilitaire de transformation convertira toutes les occurrences du type d'étape PARTIAL_UNDO en INTERIM_CHECKOUT.

Attributs à associer aux étapes

Dans WPC version 5.2.1.x, seules les étapes PARTIAL_UNDO et MODIFY devaient être associées à des collections d'attributs.

Dans InfoSphere MDM Collaboration Server version 9.1.0, les étapes AUTOMATED, INTERIM_CHECKOUT, AND_APPROVAL, MODIFY, INTERIM_CHECKIN, GENERAL, OR_APPROVAL et DISPATCH doivent être associées à une collection d'attributs.

Avant d'importer un flux de travaux, vous devez modifier le XML du fichier zip pour les types d'étapes décrits ci-avant en incluant une collection d'attribut pertinente. Le XML présentera l'aspect suivant :

```
<Step>
....
  <StepType>AUTOMATED</StepType>
....
  <EditableAttributesCollections>
    <ItemEdit>
      <AttributeCollection>Attribute Collection 1</AttributeCollection>
    </ItemEdit>
  </EditableAttributesCollections>
....
</Step>
```

Volet gauche de la zone de collaboration

Dans InfoSphere MDM Collaboration Server version 9.1.0, des zones de collaboration sont disponibles dans le volet gauche exactement comme des catalogues et des hiérarchies. Il s'agit d'un procédé rapide pour naviguer vers une étape différente sans avoir à accéder à la console de la zone de collaboration.

Rétablissement du moteur de flux de travaux au démarrage

A partir de WPC version 5.3, lorsque le moteur de flux de travaux démarre, tout événement de flux de travaux toujours marqué comme étant en COURS DE TRAITEMENT est à présent défini sur ERREUR pour indiquer qu'un problème est survenu. Cela est d'autant plus important maintenant que la plupart du traitement du flux de travaux est effectué sur le moteur de flux de travaux au lieu d'être effectué sur le serveur d'applications.

Validation d'attributs dans le flux de travaux

Dans InfoSphere MDM Collaboration Server version 9.1.0, les collections d'attribut peuvent être affectées à chaque étape d'un flux de travaux. Dans les versions précédentes de WPC, l'utilisateur sélectionnait une collection d'attributs pour tout le flux de travaux.

Attributs à valider

A partir de WPC version 5.3, chaque étape est affectée à trois collections d'attributs au maximum :

- Afficher – permet d'afficher les valeurs de ces attributs.
- Modifier – permet d'afficher et de modifier les valeurs de ces attributs. Lors de l'enregistrement, ces attributs sont vérifiés par rapport aux règles définies dans la spécification.
- Requis – permet d'afficher et de modifier ces attributs. Vous devez fournir une valeur pour tous les attributs requis. Lors de l'enregistrement, ces attributs sont vérifiés par rapport aux règles définies dans la spécification.

Règles de validation de flux de travaux InfoSphere MDM Collaboration Server version 9.1.0

- Un article ne sera autorisé à passer à l'étape suivante que s'il est considéré comme valide par les règles de l'étape que vous achevez. Il s'agit de la règle la plus importante. En d'autres termes, les règles de validation du conteneur source, et plus particulièrement les attributs requis, seront remplacées par les paramètres de validation de flux de travaux. Par exemple, les attributs obligatoires comme MinOccurs=1 seront remplacés par les paramètres de validation de flux de travaux.
- Les attributs requis et pouvant être modifiés dans le flux de travaux doivent être ajoutés à une collection d'attributs requise dans les étapes dans lesquelles ils peuvent être modifiés, même si une règle est spécifiée dans la spécification.
- Chaque article doit posséder une clé primaire unique.
- Chaque objet doit être unique au sein de la zone de collaboration jusqu'à sa restitution, moment où l'unicité de la clé du conteneur source est vérifiée.
- Une entrée en conflit avec une entrée du conteneur source sera acheminée vers FIXIT.
- Seuls les attributs éditables ou requis au sein d'une étape sont validés.

Enregistrer comme brouillon

Dans WPC version 5.3, vous pouvez enregistrer des données non valides au sein d'une étape de flux de travaux et revenir ultérieurement à cette étape. Dans les versions précédentes, les données étaient perdues lorsque vous quittiez l'écran et que des attributs non valides étaient présents. Ce cas se produit si vous appuyez sur Enregistrer ou Terminé, ce qui entraîne l'affichage d'erreurs de validation à l'écran. L'option Enregistrer en tant que brouillon fonctionne uniquement avec les valeurs qui réussissent la conversion du type. Pour que cette option fonctionne, la clé primaire doit être valide et unique au sein de la zone de collaboration. Vous ne pouvez pas faire passer une entrée contenant des données non valides à l'étape suivante tant que ce problème n'est pas résolu, car une entrée doit être validée par les règles de l'étape pour que le programme puisse être quitté correctement.

Pièges de l'enregistrement en tant que brouillon

Lorsqu'une entrée quitte une étape suite à un dépassement de délai ou est envoyée en réparation, aucune validation n'est effectuée. Veillez à définir tous les attributs concernés par le flux de travaux comme des attributs éditables à valider au cours de l'étape de réparation. Autrement, vous ne pourrez pas afficher ou corriger les attributs non valides. Les seules options seront alors les suivantes :

- Vérifier les données non valides
- Retirer l'article de la zone de collaboration

Veillez à ce que l'étape cible d'un délai d'attente englobe tous les attributs qui peuvent être modifiés avant le déclenchement du dépassement de délai. Cette opération exige d'être très vigilant avec les chaînes de délai d'attente ou les escalades de délai de flux de travaux imbriqués. En cas de doute, ajoutez tous les attributs.

Prise en charge des articles fantômes

Dans WPC version 5.2.1.1, la plupart des causes des articles fantômes ont été éliminées. Vous remarquerez que les catégories fantômes n'ont pas été abordées. Plusieurs modifications ont été apportées depuis :

- Améliorations apportées à la logique transactionnelle lors des transitions entre les étapes afin de garantir une approche tout ou rien des mises à jour de table de flux de travaux, effectuées en une seule transaction.
- Une limite d'un moteur de flux de travaux est désormais appliquée pour limiter les conflits.
- Isolement de la connexion des scripts d'étape.
- Les scripts d'étape sont désormais exécutés dans des connexions de base de données distinctes afin que leurs incidents n'affectent pas les mises à jour de table de flux de travaux.
- Le traitement de flux de travaux est à présent effectué de manière asynchrone sur le moteur de flux travaux au lieu d'être effectué de manière synchrone sur le serveur d'applications. Cette modification garantit également qu'une transition de flux de travaux est enregistrée dans la base de données avant d'être exécutée ce qui permet une reprise optimale et une meilleure gestion des erreurs.

Modifications du magasin de documents

Les fichiers suivants sont supprimés du fichier `DOCSTORE.xml` du fichier zip exporté WPC version 5.2.1 pendant l'exécution de l'utilitaire de transformation, `migrateCompanyZip.sh`, sur ce même fichier zip. Ainsi, ce fichier ne sera pas migré vers InfoSphere MDM Collaboration Server version 9.1.0. Vous devez le migrer manuellement si vous devez apporter les modifications des fichiers dans InfoSphere MDM Collaboration Server version 9.1.0.

```
/scripts/import/ctg/environment_catalog_content_import.script
/scripts/import/ctr/environment_category_tree_content_import.script
/scripts/logout/Logout.wpcs
/scripts/login/Login.wpcs
/scripts/triggers/LDAPLibrary.wpcs
/scripts/triggers/LdapLibraryLogger.wpcs
```

Les scripts associés aux étapes par défaut dans un flux de travaux sont renommés en leurs contreparties en majuscules. Par exemple, le script `Fixit` est renommé `FIXIT`.

Migration manuelle des scripts

Avant d'importer les données pour la société cible, vous devez vérifier le bon fonctionnement des scripts que vous avez migrés.

Pourquoi et quand exécuter cette tâche

Vous pouvez identifier et corriger la plupart des erreurs de syntaxe, telles que les modifications de signature et les valeurs de renvoi manquantes, en compilant les scripts. Vous pouvez utiliser le script `checkForCompileError.sh` pour compiler les scripts automatiquement. L'exécution de ce script génère un rapport qui répertorie les erreurs pour chacun des scripts. Pour obtenir la liste complète des modifications de signature et de comportement au niveau des scripts et connaître les erreurs les plus courantes et les solutions disponibles, reportez-vous à la rubrique `Compilation et débogage de scripts` InfoSphere MDM Collaboration Server.

Procédure

1. Exécutez le script `checkForCompileError.sh`. Ce script se trouve dans le répertoire `$TOP/bin`.

```
$TOP/bin/checkForCompileError.sh --company_code=<code>
--script_dir=<path/to/wpc/script/directory>
--logfile=<chemin_fichier_journal>
```

Ce script utilise le répertoire dans lequel tous les scripts sont stockés en tant qu'argument et tente de compiler chaque script dans le répertoire et ses sous-répertoires. Dans cet exemple, le script `checkForCompileError.sh` est exécuté pour compiler les scripts dans le répertoire `/scripts`. Pour vérifier tous les scripts dans le répertoire `/scripts` en utilisant le mode `script compiled_only`, vous devez modifier le fichier `common.properties` pour définir **`script_execution_mode=compiled_only`**, puis exécuter le script `checkForCompileError.sh`.

```
$TOP/bin/checkForCompileError.sh --company_code=test
--script_dir=/scripts
--logfile=logfile.txt
```

Le fichier `logfile.txt` est généré. A la fin du fichier, vous visualisez un rapport au format suivant :

```
Fri Jan 07 15:25:06 IST 2011 - *****
Fri Jan 07 15:25:06 IST 2011 - Script Mode : 'compiled_only'
Fri Jan 07 15:25:06 IST 2011 - Total Scripts : '10'
Fri Jan 07 15:25:06 IST 2011 - Successful : '8'
Fri Jan 07 15:25:06 IST 2011 - Failed      : '2'
Fri Jan 07 15:25:06 IST 2011 - /scripts/import/ctr/abhitest -
The script could not be compiled.
Fri Jan 07 15:25:06 IST 2011 - /scripts/import/ctr/AbhiFailureWrongScript -
The script object could not be built.
[[Opération getEntryAttribs non définie à la ligne 3]]
Fri Jan 07 15:25:06 IST 2011 - *****
```

2. Exécutez `checkChangedScriptOps.sh` pour obtenir la liste des scripts dont la signature et le comportement ont été modifiés par rapport à WebSphere Product Center version 5.2 et ses Fix Packs, ou qui ont été supprimés de WebSphere Product Center version 5.3.x. Ce script est disponible dans le répertoire `$TOP/bin`. script. Pour plus d'informations sur les modifications du fonctionnement des scripts, voir le centre de documentation InfoSphere MDM Collaboration Server. Ce script vous aide à effectuer une première estimation et, ultérieurement, lors de la phase de mise en oeuvre vous pouvez obtenir une liste de scripts qui requièrent davantage votre attention, ainsi que l'emplacement exact du script où des modifications sont susceptibles d'être apportées.

```
$TOP/bin/checkChangedScriptOps.sh --company_code=<company code>
--use_docstore=<yes/no> --src_path=<source path>
[--dst_path=<destination path>] --logfile=<log file path>
```

dst_path

Indique le chemin de destination. Ce paramètre est facultatif. Si vous n'indiquez pas cet argument, le script analyse et remplace tous les scripts du dossier source indiqué par le paramètre **`src_path`**.

logfile Indique les détails du fichier journal au format suivant :

<chemin complet du fichier script > <numéro de ligne> <nom de l'opération du script obsolète>

Commentaire

Indique un commentaire sur l'opération d'un script obsolète.

Lorsque l'argument **`dst_path`** est indiqué dans le script de shell, il modifie le script source en ajoutant un commentaire juste au dessus des occurrences de l'opération de script obsolète et copie ce script modifié dans l'emplacement

cible. Toutes les modifications apportées au script sont balisées par le texte *//modified_by wpc_automated_tool* pour en faciliter l'identification.

Lorsque le paramètre **use_docstore** correspond à no, le paramètre **company_code** est facultatif. Si vous indiquez une valeur, celle-ci est ignorée.

Les erreurs d'exécution incluent des incohérences de type car InfoSphere MDM Collaboration Server version 9.1.0 applique un scriptage qui s'apparente davantage à un langage type dans lequel une variable ne peut pas stocker des types non liés. Comme il y a une modification du comportement dans le fonctionnement de quelques scripts, les valeurs renvoyées peuvent générer des erreurs inattendues. Vous devez examiner ces erreurs au cas par cas. Pour plus d'informations sur la modification des scripts, voir Compilation et débogage de scripts InfoSphere MDM Collaboration Server.

Remarque : Bien que l'utilitaire ne prenne pas en charge les modifications d'API des scripts apportées après WPC version 5.3.2, vous pouvez être amené à identifier les différences existants entre le fonctionnement des scripts de la version 5.2.1.4 et de la version 5.3.2 de WPC.

Chapitre 5. Identification des incidents

Vous pouvez utiliser les solutions recommandées pour résoudre les problèmes de migration courants.

Identification et résolution des problèmes de migration

Vous devez compiler les scripts de migration avant de pouvoir les afficher dans la console de script d'IBM InfoSphere Master Data Management Collaboration Server et les exécuter. Vous pouvez également utiliser les fonctions d'InfoSphere MDM Collaboration Server pour identifier et résoudre les incidents liés à vos scripts.

Utilisation de scripts de migration compilés

Vous devez modifier le fichier `common.properties` avant d'utiliser des scripts compilés.

Avant de pouvoir utiliser des scripts compilés, assurez-vous que le fichier `common.properties` du répertoire `$TOP/etc/default` contient le paramètre suivant :

```
script_execution_mode = compiled_only
```

Vous pouvez désactiver la compilation de script pour les scripts individuels en insérant la directive suivante au début du script :

```
script_execution_mode=not_compiled
```

Cependant, la désactivation de la compilation de script au niveau du script n'est pas recommandée, car elle entraîne une altération importante des performances. Pour éviter toute altération des performances lorsque vous utilisez des scripts non compilés, définissez le paramètre de votre serveur sur *not_compiled* au lieu d'utiliser l'option de désactivation de la compilation au niveau du script.

Si le paramètre de serveur du fichier `common.properties` est défini sur *not_compiled*, la compilation de script pour des scripts individuels ne peut pas être activée à l'aide de directives de niveau de script.

Remarque : L'utilisation d'une combinaison de scripts compilés et non compilés altère les performances et EST DECONSEILLÉE. Si vous devez néanmoins utiliser une telle combinaison, sachez qu'il existe une limitation : un script non compilé peut appeler des fonctions dans un script compilé, mais un script compilé ne peut pas appeler une fonction dans un script non compilé.

Erreurs de compilation de script courantes

Lorsque vous utilisez des scripts compilés, un script peut être enregistré dans la console de scripts s'il se compile correctement. Si une erreur se produit, vérifiez `svc.out` dans le répertoire des journaux `appsrv` afin de consulter la sortie `javac` complète et le message d'erreur.

Voici quelques erreurs de compilation courantes :

1. Une instruction `break` ou `return` à l'intérieur d'un bloc `forEach*Element()` ne se compile pas en raison d'une erreur "Code inaccessible". Pour résoudre cet incident :

```
forEachItemSetElement(itemSet, item)
{
    return item;
}
```

A remplacer par :

```
forEachItemSetElement(itemSet, item)
{
    if (item != null)
    {
        return item;
    }
}
```

Cela revient au même, mais convient mieux au compilateur.

2. Si vous renvoyez une valeur à partir d'une fonction, vous devez renvoyer une valeur dans tous les cas. En d'autres termes, ceci ne se compile pas :

```
function sample() {
    var e = null;
    catchError (e) {
        // faire quelque chose...
        return "a string";
    } if (e != null) {
        reportError(...);
    }
}
```

Aucune valeur n'est renvoyée si une exception se produit dans le bloc `catchError`. Vous devez le remplacer par :

```
function sample() {
    var e = null;
    catchError (e) {
        // faire quelque chose...

        return "a string";
    }
    if (e != null) {
        reportError(...);
    }
    return null;
}
```

3. Pour les incidents de compilation plus sérieux, vous pouvez consulter les fichiers Java générés. Ces fichiers Java générés se trouvent dans le répertoire spécifié par le paramètre *tmp_dir* dans `common.properties`. La convention de dénomination des fichiers Java a été récemment modifiée afin d'inclure le nom du script ainsi qu'une séquence générée, par exemple : `MyScript12345.java`.
4. En outre, le chemin d'accès absolu au script depuis le magasin de documents est indiqué sous forme de commentaire en haut de chaque fichier Java généré. Si vous mappez le magasin de documents au système de fichiers, vous pouvez exécuter une commande `grep` récursive pour trouver le fichier Java qui correspond à un script.

Erreurs et incidents d'exécution courants

Vous pouvez rencontrer certains problèmes lorsque vous exécutez des scripts. Les exemples suivants illustrent certains problèmes connus et vous donnent un aperçu des contournements et solutions possibles.

Type d'argument non valide :

Cette erreur se produit lorsque vous transférez le mauvais type d'argument à une fonction (par exemple, `HashMap` lorsque `String` est requis). Cette

erreur se produit également si IBM InfoSphere Master Data Management Collaboration Server ne parvient pas à supposer le type correctement. Pour résoudre cet incident, vous devrez peut-être utiliser une opération de script telle que *checkString()* afin de rendre le type explicite.

Non concordance de type d'arguments dans les comparaisons :

Si un même type de données, tel que `==`, `>`, `<`, `<=`, etc., n'apparaît pas aux deux extrémités d'un opérateur conditionnel, l'expression aura un résultat "false". Aucun message d'erreur ne s'affichera, mais le code correspondant ne sera pas exécuté.

Par exemple, le code suivant ne fonctionnera pas :

```
var id = "12345" ;
var my_id = item.getEntryAttrib(chemin d'accès à un attribut de type séquence) ;
if ( id == my_id ) {
// instructions qui doivent être exécutées mais ne le seront pas
}
```

Dans ce cas de figure, la solution consiste à utiliser de manière explicite le code suivant :

```
var id = "12345" ;
var my_id = checkString(item.getEntryAttrib("//attribut de type séquence"),"") ;
if ( id == my_id ) {
// instructions à exécuter
}
```

Analyse XML :

Le code suivant utilise le mode non compilé et également le mode compilé lorsqu'il est exécuté depuis l'écran Script Sandbox :

```
new XmlDocument(xmlDoc) ;
foreachXmlNode("item") {
//faire le nécessaire
}
```

Toutefois, en mode compilé, si ce code est utilisé dans une fonction de bibliothèque de scripts appelée par plusieurs utilisateurs, les instructions dans *foreachXmlNode* ne seront pas exécutées. Aucun message d'erreur ne s'affiche. La solution consiste à utiliser le code suivant :

```
var doc = new new XmlDocument(xmlDoc) ;
var xmlNode ; foreachXmlNode(doc, "item", xmlNode) {
//faire le nécessaire
}
```

Résolution des erreurs et incidents d'exécution courants

Pour résoudre les erreurs d'exécution sur le serveur d'applications, consultez le fichier `svc.out` dans le répertoire `appsvr log`. L'examen des fichiers `exception.log` et `default.log` peut s'avérer utile dans certains cas.

Grâce à la nouvelle convention de dénomination de fichier Java, vous pouvez aisément identifier le script défectueux. Le message d'erreur identifie également le numéro de ligne dans le fichier Java généré. Pour résoudre l'incident, affichez le fichier Java généré et accédez à la ligne qui a engendré l'erreur d'exécution. Le code Java généré inclut à présent le code de script actuel sous forme de commentaires, toutes les quelques lignes.

Considérons, par exemple, la portion de code suivante issue d'un exemple de fichier Java généré :

```
// function checkIfPartyPartyTypeExist(party, partyType)
public static Object ScriptFunction__checkIfPartyPartyTypeExist(HashMap hmContext, Object party, Object
partyType) throws Exception
{
// var bRet = false;
Object bRet = (java.lang.Boolean) Boolean.FALSE; //
var rootEntry = party.getRootEntryNode();
Object rootEntry = GenGetRootEntryNodeOperation.execute(hmContext , (IEntry) party);
// var entryNodes = rootEntry.getEntryNodes(getCatalogSpecName() + "/Party Types/Party Type Code");
Object entryNodes = GenGetEntryNodesOperation.execute(hmContext , (EntryNode) rootEntry, (String)
```

```
BinaryOperation.execute(BinaryOperation.PLUS, ScriptFunction__getCatalogSpecName(hmContext), "/Party
Types/Party Type Code"));
// var entryNodesSize = entryNodes.size();
Object entryNodesSize = (java.lang.Integer) GenSizeOperation.execute(hmContext , (HashMap) entryNodes);
```

Chaque ligne commençant par // dans les commentaires ci-dessus correspond au code actuel issu du script IBM InfoSphere Master Data Management Collaboration Server correspondant. Il est ainsi plus facile d'identifier la ligne qui a engendré l'incident dans le script.

Utilisation du script **checkForCompileError.sh**

Vous devez vérifier les scripts à la recherche d'erreurs de compilation.

Veillez à corriger toutes les erreurs détectées au cours de la compilation avant de démarrer l'importation dans la société cible.

Le script **checkForCompileError.sh** est fourni afin d'activer la compilation de script automatique.

Utilisez-le comme suit :

```
checkForCompileError.sh --company_code=<code> --script_dir=<path/to/wpc/script/directory>
--logfile=<chemin d'accès au fichier journal>
```

Cette commande prend `script_dir` (chemin d'accès au magasin de documents) en tant qu'argument dans lequel tous les scripts sont présents. Ce script tente de compiler tous les scripts présents dans un répertoire de scripts spécifié et dans ses sous-répertoires et génère un rapport qui contiendra le nom du script et les erreurs.

Si vous souhaitez vérifier tous les scripts disponibles dans le répertoire du magasin de documents `/scripts/import/ctr/` à l'aide du mode de script *compiled_only*, modifiez tout d'abord le fichier `common.properties` sur la valeur `script_execution_mode=compiled_only` et exécutez le script `checkForCompileError.sh` depuis le répertoire `$TOP`.

```
./bin/checkForCompileError.sh --company_code=test --script_dir=/scripts --logfile=logfile.txt
```

Un fichier `logfile.txt` sera généré. A la fin de ce fichier, le rapport au format suivant sera présent :

```
Ven 21 Jui 15:25:06 IST 2006 - *****
Ven 21 Jui 15:25:06 IST 2006 - Mode de script : 'compiled_only'
Ven 21 Jui 15:25:06 IST 2006 - Nombre total de scripts : '10'
Ven 21 Jui 15:25:06 IST 2006 - Succès : '8'
Ven 21 Jui 15:25:06 IST 2006 - Echec : '2'
Ven 21 Jui 15:25:06 IST 2006 - /scripts/import/ctr/abhitest - Impossible de compiler
le script.
Ven 21 Jui 15:25:06 IST 2006 - /scripts/import/ctr/AbhiFailureWrongScript - Impossible
de générer l'objet de script. [[Opération getEntryAttribs non définie à la ligne 3]]
Ven 21 Jui 15:25:06 IST 2006 - *****
```

Identification et résolution des problèmes de migration

Décrit certains problèmes courant de résolution des problèmes qui peuvent apparaître lors de la migration dans IBM InfoSphere Master Data Management Collaboration Server.

Problème : index manquants

Lors de la migration vers InfoSphere MDM Collaboration Server, Version 10.0.0, des index peuvent ne pas figurer dans le rapport de la console de migration.

Les erreurs suivantes liées aux index manquants peuvent se produire lors de la migration :

Index manquants dans la table SCA

```
Missing Indexes
=====
SCA_0_PK SCA_CATEGORY_IDSCA_SELECTION_IDSCA_COMPANY_IDSCA_
CATALOG_IDSCA_CAT_TREE_ID
```

You can run this PERL script to resolve this issue. Choose directory db2 or oracle appropriately.

```
perl $PERL5LIB/runSQL.pl --sql_file=$TOP/src/db/schema
/dbscripts/<db2 or oracle>/sca_pk.sql
```

Index manquants dans la table DBV

```
Missing Indexes
=====
DBV_0_UK DBV_VERSION
```

You can run this PERL script to resolve this issue. Choose directory db2 or oracle appropriately.

```
perl $PERL5LIB/runSQL.pl --sql_file=$TOP/src/db/schema
/gen/<db2 or oracle>/ddl_ver_synchronize.sql
. $TOP/bin/compat.sh
$JAVA_RT com.ibm.ccd.synchronize.DBSchemaVersion --autoupd
```

Index manquants dans les tables DOA et CTG (DB2 uniquement)

```
Missing Indexes
=====
CTG_1_UK CTG_COMPANY_IDCTG_NAME
DOA_0_UK DOA_DOC_IDDOA_CMP_IDDOA_NAME
```

You can run this PERL script to resolve this issue:

```
perl $PERL5LIB/runSQL.pl
--sql_command="alter table tctg_ctg_catalog drop constraint ctg_1_uk ;"
perl $PERL5LIB/runSQL.pl
--sql_command="drop index ictg_ctg_2;"
```

```
perl $PERL5LIB/runSQL.pl
--sql_command="alter table tctg_ctg_catalog
add constraint ctg_1_uk unique (ctg_company_id, ctg_name);"
```

```
perl $PERL5LIB/runSQL.pl
--sql_command="create index ictg_ctg_2 on
tctg_ctg_catalog ( ctg_name, ctg_company_id)
ALLOW REVERSE SCANS;"
```

```
perl $PERL5LIB/runSQL.pl
--sql_command="alter table TDOC_DOA_DOC_ATTRIBUTES
drop constraint doa_0_uk ;"
```

```
perl $PERL5LIB/runSQL.pl
--sql_command="drop index idoc_doa_0;"
```

```
perl $PERL5LIB/runSQL.pl
--sql_command="alter table tdoc_doa_doc_attributes
add constraint doa_0_uk unique (doa_doc_id, doa_cmp_id, doa_name);"
```

```
perl $PERL5LIB/runSQL.pl
--sql_command="create index idoc_doa_0
on tdoc_doa_doc_attributes ( doa_name, doa_doc_id, doa_cmp_id)
ALLOW REVERSE SCANS;"
```

Problème : tables ITX et CAX manquantes

Lors de la migration vers InfoSphere MDM Collaboration Server, Version 10.0.0, l'erreur suivante peut figurer dans le rapport de vérification BD :

```
Missing Tables
=====
TCTG_CAX_CATEGORY_CONTENT
TCTG_ITX_ITEM_CONTENT

Missing Indexes|
CAX_0_PK CAX_ENTRY_IDCAX_NEXT_VERSION_ID
ITX_0_PK ITX_ENTRY_IDITX_NEXT_VERSION_ID
```

Cette erreur indique un problème de création d'espace table, d'installation XDB ou de paramètres BD. Vérifiez le fichier `errfile.log` dans le répertoire `$TOP/logs` pour connaître le message exact. Vérifiez que la base de données est correctement configurée et exécutez de nouveau le script de migration.

Problème : privilèges insuffisants

Lorsque vous exécutez un script de migration vers InfoSphere MDM Collaboration Server, Version 10.0.0, une erreur de privilèges insuffisants peut figurer sur la console. Exemples d'erreurs types :

Base de données Oracle

Lorsque vous utilisez une base de données Oracle, l'erreur peut se présenter comme suit :

```
create index icnt_eem_2 on tcnt_eem_entry_entry_map (
*
ERROR at line 1:
ORA-01031: insufficient privileges
```

DB2 database

Dans le cas d'une base de données DB2, l'erreur se présente comme suit :

```
SQL0551N  "USERNAME" does not have the privilege to perform operation
"CREATE INDEX" on object "USERNAME.TWFL_WFE_WORKFLOW_EVENT".
SQLSTATE=42501
```

L'utilisateur de la base de données ne possède pas de privilèges suffisants lui permettant de créer un index dans la base de données. Vous devez accorder le privilège "create index" à l'utilisateur de la base de données, puis exécuter de nouveau le script de migration. Vous devez également accorder l'accès à l'utilisateur afin qu'il puisse créer et modifier des tables.

Problème : échec du script de migration

Lorsque vous exécutez un script de migration vers InfoSphere MDM Collaboration Server, Version 10.0.0, le script peut échouer. Scénario type d'échec de script :

Failure in module data_maintenance_reports

```
-----
Summary of the migration
-----
La migration des modules suivants a échoué :
data_maintenance_reports
```

The `$TOP/logs/errfile.log` file contains the following content:

```
net.sf.ehcache.distribution.  
MulticastKeepaliveHeartbeatReceiver$MulticastReceiverThreadrun  
SEVERE: Multicast receiver thread caught throwable. Cause was null. Continuing...
```

Cette erreur indique un problème au niveau des paramètres de configuration de cache, mais le script de migration s'est poursuivi et a abouti. Vérifiez les paramètres de configuration de cache. Il n'est pas nécessaire de réexécuter le script de migration.

Remarques

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Contactez votre interlocuteur IBM pour plus d'informations sur les produits et services disponibles dans votre région. Toute référence à un produit, programme ou service IBM n'implique pas que seul ce produit, programme ou service IBM peut être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevets couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Pour le Canada, veuillez adresser votre courrier à :

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

Les informations sur les licences concernant les produits utilisant un jeu de caractères double octet peuvent être obtenues par écrit auprès d'IBM à l'adresse suivante :

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

Le paragraphe suivant ne s'applique ni au Royaume-Uni, ni dans aucun pays dans lequel il serait contraire aux lois locales. LE PRESENT DOCUMENT EST LIVRE «EN L'ETAT» SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et logiciels décrits dans ce document.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans ces informations et tous les éléments sous licence disponibles s'y rapportant sont fournis par IBM conformément aux dispositions de l'ICA IBM Customer Agreement), 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. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

LICENCE DE COPYRIGHT :

Le présent logiciel contient des exemples de programme d'application en langage source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation des plateformes pour lesquels ils ont été écrits ou aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM, ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes. Les exemples de programmes sont fournis "en l'état", sans garantie d'aucune sorte. IBM ne sera en aucun cas responsable des dommages liés à l'utilisation de ces programmes.

Toute copie totale ou partielle de ces programmes exemples et des oeuvres qui en sont dérivées doit comprendre une notice de copyright, libellée comme suit :

© (nom de votre entreprise) (année). Des segments de code sont dérivés des Programmes exemples d'IBM Corp. © Copyright IBM Corp. _entrez l'année ou les années_. 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

IBM, le logo IBM et ibm.com sont des marques d'International Business Machines Corporation aux Etats-Unis et/ou dans certains autres pays.

IBM, le logo IBM et ibm.com sont des marques ou des marques déposées d'International Business Machines Corp. dans de nombreux pays. Les autres noms de produits et de services peuvent appartenir à IBM ou à des tiers. Vous trouverez une liste des marques IBM sur le Web dans la section "Copyright and trademark information" à l'adresse www.ibm.com/legal/copytrade.shtml.

Adobe est une marque d'Adobe Systems Incorporated aux Etats-Unis et/ou dans certains autres pays.

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

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

Linux est une marque de Linus Torvalds aux Etats-Unis et/ou dans certains autres pays.

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

UNIX est une marque enregistrée de The Open Group aux Etats-Unis et/ou dans certains autres pays.

Les autres noms de produits et de services peuvent être des marques d'IBM ou d'autres sociétés.

Index

Nombres

10.0 65
10.1.0 48, 49
5.3.0.1 48
6.0.0 49
9.1.0 65, 67

A

ajout 23
api java
 10.0 67
 6.0.0 65
 énumérations 65
 interfaces 67
 méthodes 65
attributs pouvant faire l'objet de
 recherches
 gds 23

B

base de données
 gds 68
base de onnées
 gds 25

C

commentaires
 envoi vii
compilation 41
contenu des catalogues
 gds 39
contenu des hiérarchies
 gds 39
copie de sauvegarde 25

D

débogage 41
 gds 41
différences des API Java 65
différences entre les scripts 45

E

environnement
 GDS 25, 34
environnement migré
 gds 41
envoi de commentaires vii
exporter 25, 33

F

fonctions
 nouvelles dans version 10.1 vii

G

gds 24
GDS 18
 données 19
gds.properties
 gds 23
GDS version 1.3.x 18

I

identification et résolution des problèmes
 problèmes de migration 78
importation de données 78
importer 34, 39
informations connexes vii
informations sur les prérequis vii

L

limitations 31, 40

M

magasin de documents 33
marques 86
métadonnées 68
méthodes
 dans la version 9.0.0 60
 uniquement dans V6.5.0 migré vers
 V10.1.0 60
mettre à jour 23
migration
 identification et résolution des
 incidents liés aux scripts de
 migration 75
 initiation 3
 migration de la version 6.0.0 vers la
 version 10.0.0 9
 migration de v10.0.0 vers v10.1.0 16
 migration de v9.1.0 vers v10.1.0 13
 migration depuis des versions
 antérieures 5
 présentation 1
 problèmes 75
 scripts 45, 75
 vérification des erreurs de
 compilation 78
migrer 18
 GDS 1.3.x 19
modèle de données
 gds 68

N

nettoyage 19

O

optimisation 68

P

préparer 19

S

script
 5.3.0 48
 différences 48, 49
 méthodes dans V10.1.0 51
 méthodes de la version 5.3.0.1 51
script d'exportation
 GDS 31
script d'importation
 gds 40
Script MigrateExport.sh 25
scripts
 changement dans le comportement de
 V5.2.1 vers V10.1.0 60
 changement dans le comportement de
 V5.3.0.1 vers V10.1.0 63
 changement dans le comportement de
 V6.0.0 et V10.1.0 64
 compilation et débogage 75
 différences dans V5.2.1 et V10.1.0 46
 erreurs d'exécution courantes 76
 erreurs de compilation courantes 75
 méthodes dans la version 5.2.1 49
 méthodes dans V10.1.0 56
 résolution des erreurs d'exécution 77
scripts de migration
 gds 72
scripts de mise en oeuvre
 GDS 22
stop 24
supplémentaire 23
supprimer 22

T

traitement des incidents
 contacter le support logiciel IBM xi
transformateur
 GDS 69
transformation 68

V

vérifier 41, 72

X

XSLT 69



SC11-7034-01

