



Peregrine | **Connect-It**
Utilisation



© Copyright 2002 Peregrine Systems, Inc.

Tous droits réservés.

Les informations contenues dans ce document sont la propriété de Peregrine Systems, Incorporated, et ne peuvent être utilisées ou communiquées qu'avec l'autorisation écrite préalable de Peregrine Systems, Inc. La reproduction de tout ou partie de ce manuel est soumise à l'accord écrit préalable de Peregrine Systems, Inc. Cette documentation désigne de nombreux produits par leur marque. La plupart de ces citations sont des marques déposées de leurs propriétaires respectifs.

Peregrine Systems® et Connect-It® sont des marques déposées de Peregrine Systems, Inc.

Ce produit contient des composants logiciels développés par Apache Software Foundation (<http://www.apache.org>).

Les logiciels décrits dans ce manuel sont fournis avec un contrat de licence entre Peregrine Systems, Inc., et l'utilisateur final ; ils doivent être utilisés suivant les termes de ce contrat. Les informations contenues dans ce document sont susceptibles d'être modifiées sans préavis et sont fournies sans engagement aucun de la part de Peregrine Systems, Inc. Contactez le support client de Peregrine Systems, Inc. pour contrôler la date de la dernière version de ce document.

Les noms de personnes et de sociétés cités dans le manuel, dans la base d'exemple ou dans les visites guidées sont fictifs et sont destinés à illustrer l'utilisation des logiciels. Toute ressemblance avec des sociétés ou personnes existantes ou ayant existé n'est qu'une pure coïncidence.

Pour toute information technique sur ce produit ou pour faire la demande d'une documentation sur un produit dont vous possédez la licence, veuillez contacter le support client Peregrine Systems, Inc. en envoyant un e-mail à l'adresse suivante : support@peregrine.com.

Pour tout commentaire ou suggestion à propos du présent document, veuillez contacter le département des publications techniques de Peregrine Systems, Inc. en envoyant un e-mail à l'adresse suivante : doc_comments@peregrine.com.

Cette édition s'applique à la version 3.2.0 du programme sous contrat de licence
Connect-It

Peregrine Systems, Inc.
Worldwide Corporate Campus and Executive Briefing Center
3611 Valley Centre Drive San Diego, CA 92130
Tel 800.638.5231 or 858.481.5000
Fax 858.481.1751
www.peregrine.com



Table des matières

Introduction	11
A quoi sert Connect-It	12
A qui s'adresse Connect-It	12
Comment utiliser ce manuel	13
Chapitre 1. Principes généraux	17
Traitement des données	17
Chapitre 2. Installation	19
Environnements supportés	19
Installer Connect-It	20
Saisir le certificat d'autorisation	23
Contenu de l'installation	24
Chapitre 3. Editeur de scénarios	31
Fenêtre principale	31
Menus	40
Favoris	48
Journaux	54
Options	69

Chapitre 4. Mise en place d'un scénario d'intégration	79
Mettre en place un scénario avec un assistant	79
Mettre en place manuellement un scénario	80
Editer les types de document produits ou consommés	84
Enregistrer un scénario	93
Ouvrir un scénario existant	94
Chapitre 5. Mappings des types de document	97
Boite de mapping	98
Editer des mappings	100
Types de mapping	112
Chapitre 6. Scripts de mapping	121
Référence de programmation	122
Fonctions Basic	122
Aide à la rédaction des scripts de mapping	132
Chaîne de caractères	138
Edition des fichiers associés	140
Fonctions et variables globales	146
Tables de chaînes de caractères	146
Tables de correspondance	147
Formats utilisateur	149
Collection - exemples	151
Chapitre 7. Types de document pivots	157
Principe de fonctionnement des types de document pivot	158
Créer un scénario d'intégration en utilisant les types de document pivots	159
Liste des types de document pivots	160
Modifier les mappings Type de document publié - type de document pivot	161
Changer de connecteur destination	162
Chapitre 8. Test et débogage d'un scénario d'intégration	163
Tester un type de document produit	163
Utiliser le journal des documents	164
Utiliser les fichiers de cache	165
Travailler hors connexion	167
Chapitre 9. Documentations des scénarios	171
Contenu d'une documentation de scénario	171
Visualisation de la documentation d'un scénario au format HTM	174

Propriétés des documentations de scénario	175
Création d'une documentation de scénario	178
Chapitre 10. Exploitation d'un scénario d'intégration	179
Créer une programmation	179
Créer un service Connect-It	190
Gérer le suivi d'un scénario en utilisant la console des services	196
Optimiser les performances d'un scénario	197
Chapitre 11. Bilans de traitement	213
Contenu d'un bilan de traitement	214
Connecteurs pouvant utiliser les bilans de traitement	215
Assistant Gestion des bilans de traitement	218
Bilan de traitement - exemple d'utilisation	219
Didacticiel	223
Chapitre 12. Kit de développement Java de Connect-It	229
Contenu du kit de développement java	229
Implémentation de la norme JCA	230
Norme UDC (Unified Document Content)	238
Création d'un connecteur Java	241
Création d'un connecteur événementiel	261
Création d'un connecteur JCA - exemple	264
Création d'un assistant de configuration amélioré	267
A. Comment signaler un problème	275
Informations générales	275
Informations spécifiques au problème	277
B. Glossaire	279
Termes Connect-It	279
Termes transversaux	295
Index	299

Liste des illustrations

1. Connect-It - composants de l'application	11
1.1. Transfert de données d'une base de données vers une autre base avec Connect-It	17
3.1. Editeur de scénarios - fenêtre principale	32
3.2. Journal de Connect-It	55
3.3. Editeur de scénarios - onglet du journal des documents	57
3.4. Fenêtre de configuration du journal des documents	57
3.5. Signal de problème de traitement	60
3.6. Filtre sur les lignes de suivi du journal des documents	61
3.7. Filtrage des lignes de suivi	61
3.8. Une chaîne de description dans le document	62
3.9. Une collection dans le détail d'un type de document et le détail d'un document	63
3.10. Détail du document pcsoftware consommé par le connecteur ServiceCenter	64
3.11. Ligne de suivi grisée signalant une ligne de suivi sous le noeud	67
3.12. Filtrage des lignes de suivi dans le détail des documents	68
4.1. Boîte à outils de l'éditeur de scénarios	81
4.2. Liaison des composants d'un scénario	82
4.3. Créer un lien ou déplacer un connecteur	83
4.4. Fenêtre d'édition des type de documents consommés par le connecteur ServiceCenter	86

4.5. Détail de la zone de destination	87
4.6. Fenêtre de visualisation des données d'un type de document produit	90
5.1. Fenêtre d'édition d'un mapping	101
5.2. Résumé des scripts de mapping sur le plan de travail	102
5.3. Boîte de sélection de vos composants et du type de document source	105
5.4. Élément source dans le script du mapping	114
5.5. Collection	114
5.6. Mapping collection à document	116
5.7. Mapping champ à collection	117
6.1. Glisser-déplacer plusieurs champs	134
6.2. L'éditeur des tables de correspondance	141
6.3. Configuration de l'éditeur de texte	145
7.1. Utilisation d'un type de document pivot	158
7.2. Affichage des connecteurs internes	160
8.1. Connecteur - Icône signalant l'utilisation d'un cache	166
8.2. Connecteur - Icône signalant le travail hors connexion	168
10.1. L'éditeur de programmeurs	181
10.2. Fenêtre d'édition de la programmation	186
10.3. Classement des mappings	187
10.4. Console des services	191
10.5. Traitement des documents - barre de progression	202
11.1. Schéma du scénario <code>idd\iddac36\iddac.scn</code>	220
12.1. Kit de développement Java - Classes CCI	232
12.2. Kit de développement Java - Classes SPI	235
12.3. Kit de développement Java - classes événementielles	237
B.1. Connecteur Asset Management - Types de documents publiés	282
B.2. Éléments d'un type de document	282
B.3. Collection d'un type de document	284
B.4. Arborecence d'un type de document	285

Liste des tableaux

3.1. Editeur de scénarios - icônes de la barre d'outils générique	32
3.2. Editeur de scénarios - icônes de la barre d'outils des favoris	34
3.3. Editeur de scénarios - icônes de la barre d'outils du schéma du scénario	34
3.4. Editeur de scénarios - icônes de la barre des vues	34
3.5. Editeur de scénarios - icônes de la barre d'outils du Studio	35
3.6. Fonctions des commandes du menu Fichier	40
3.7. Fonctions des commandes du menu Edition	41
3.8. Fonctions des commandes du menu Affichage	42
3.9. Fonctions des commandes du menu Scénario	42
3.10. Fonctions des commandes du menu Composants	44
3.11. Fonctions des commandes du menu Journal	45
3.12. Fonctions des commandes du menu Administration	45
3.13. Fonctions des commandes du menu Java	46
3.14. Fonctions des commandes du menu Aide	46
6.1. Editeur de texte - barre d'outils	143
6.2. Exemples de formats de type date	151
6.3. Exemples de formats de type numérique	151
11.1. Informations contenues dans un bilan de traitement	214
11.2. Détail du mapping entre le bilan de traitement produit par la boîte de mapping Mapping et le type de document DirectoryPoolerAction consommé par le connecteur InfraTools Desktop Discovery	220

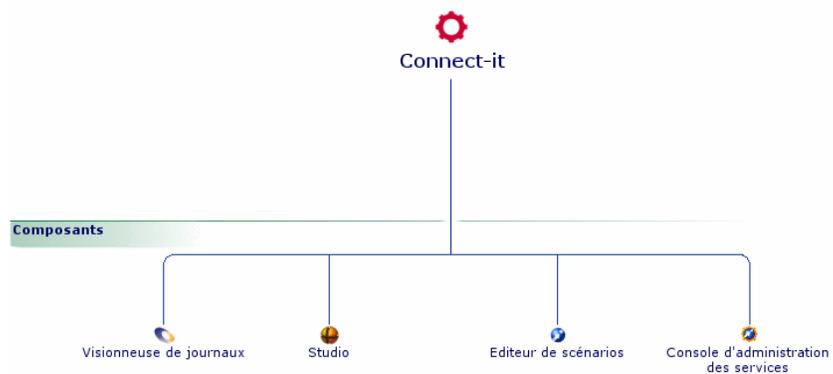
11.3. Détail du mapping entre le bilan de traitement produit par le connecteur Asset Management et le type de document DirectoryPoolerAction consommé par le connecteur InfraTools Desktop Discovery	221
12.1. Contenu du kit de développement java de Connect-It	229

Introduction

PREFACE

Connect-It est une application Peregrine System dont les composants sont détaillés dans le schéma suivant.

Figure 1. Connect-It - composants de l'application



A quoi sert Connect-It

Connect-It est une plateforme d'intégration. Elle appartient au groupe des EAI (Enterprise Application Integration). Une solution EAI permet à une entreprise d'intégrer les différentes applications qui lui permettent de d'obtenir ou de fournir des données en interne (progiciel de gestion des équipements, support technique interne, annuaire LDAP) ou en externe (ERP, B2B, B2C).

Connect-It intègre autant les données que les processus applicatifs de l'entreprise.

Connect-It peut être utilisé pour :

- Transférer des données d'une base de données vers une autre base de données
- Répliquer des données entre deux bases de données différentes
- Importer des données contenues dans des e-mails, des fichiers texte délimité, des fichiers XML ou d'autres formats dans une base de données
- Exporter les données d'une base de données vers des e-mails, des fichiers texte délimité, des fichiers XML ou d'autres formats
- Importer des informations de sécurité NT dans une base de données
- Etc.

A qui s'adresse Connect-It

Connect-It s'adresse aux techniciens informatiques chargés de l'intégration des différentes applications de leur entreprise.

L'utilisation de Connect-It nécessite les compétences suivantes :

- la connaissance avancée des applications devant être intégrées
- la connaissance du langage Basic utilisé dans les scripts de mappings
- la connaissance de la norme JCA (Java Connector Architecture) dans le cas de la création de connecteurs à l'aide du kit de développement Java.

Comment utiliser ce manuel

Chapitre Principes généraux

Ce chapitre présente les principes généraux de fonctionnement de Connect-It.

Chapitre Installation

Ce chapitre détaille l'installation de Connect-It.

Chapitre Editeur de scénarios

Ce chapitre présente l'ergonomie de l'éditeur vous permettant de créer des scénarios d'intégration.

Chapitre Mise en place d'un scénario d'intégration

Ce chapitre présente la mise en place d'un scénario d'intégration. Cette mise en place qui consiste à :

- choisir des connecteurs source et destination
Les connecteurs communiquent avec les applications externes qu'un scénario Connect-It permet d'intégrer : connecteur de type Base de données, connecteur e-mail, etc.
- créer les types de documents produits ou consommés par les connecteurs
Les types de document correspondent à la traduction au format XML des ensembles de données gérées par les applications externes : tables de base de données, e-mails, fichiers texte, formats propriétaires, etc.

Cette mise en place est logiquement complétée par l'étape de mapping présentée dans le chapitre suivant.

Chapitre Mappings des types de document

Ce chapitre présente l'édition des mappings entre les types de document produits et consommés par les connecteurs source et destination.

Un mapping définit la transformation des données d'un type de document produit par un connecteur source pour qu'un connecteur destination puisse le consommer.

Chapitre Types de document pivots

Ce chapitre présente les types de document pivot. Les types de document pivot permettent le transfert de données entre un connecteur source et un connecteur destination sans que la création d'un mapping soit nécessaire.

Chapitre Test et débogage d'un scénario d'intégration

ce chapitre présente comment tester et débogger les scénarios d'intégration avant que vous les utilisiez en production.

Chapitre Documentations des scénarios

Ce chapitre présente la fonctionnalité vous permettant de créer une documentation dynamique d'un scénario d'intégration. Cette documentation présente les connecteurs utilisés par le scénario, les types de documents source et destination, le détail des mappings et de nombreuses autres informations.

Chapitre Exploitation d'un scénario d'intégration

Ce chapitre présente la mise en exploitation d'un scénario d'intégration : sa programmation (fréquence d'exécution d'un scénario), son utilisation en tant que service sous Windows ou démon sous Unix.

Chapitre Bilans de traitement

Ce chapitre présente les bilans de traitement utilisés par les connecteurs. Ces bilans sont des types de documents rendant compte du traitement des données par les connecteurs. Les informations de réussite ou d'échec de traitement de données peuvent être utilisées pour déclencher des actions de la part d'autres connecteurs : arrêt de traitement, envoi de messages d'alerte au support, etc.

Chapitre Kit de développement Java de Connect-It

Ce chapitre présente le kit de développement Java livré avec Connect-It. Ce kit de développement vous permet de créer vos propres connecteurs Java en respectant la norme JCA (Java Connector Architecture).

Annexe Comment signaler un problème

Cette annexe présente la manière de signaler un problème au support de Peregrine Systems. Outre la procédure commune à tous les produits Peregrine

Systems, la section [Si vous signalez un problème relatif à Connect-It](#) [page 278] présente la procédure propre à Connect-It.

Glossaire

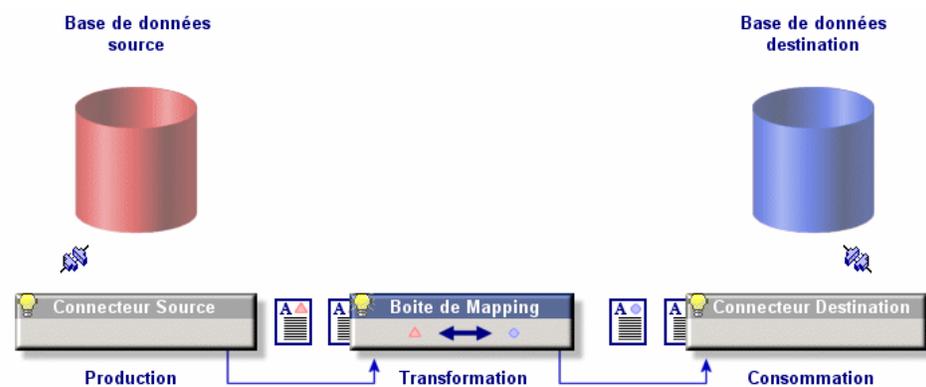
Ce glossaire regroupe une liste non exhaustive des termes spécifiques à Connect-It.

1 | Principes généraux

CHAPITRE

Traitement des données

Figure 1.1. Transfert de données d'une base de données vers une autre base avec Connect-It



Connect-It interagit avec des applications externes par le biais de connecteurs. Dans le cas d'un transfert de données d'une base de données vers une autre :

- un connecteur source produit des documents de type XML.
Chaque document correspond à un ensemble de données de l'application source.
- une boîte de mapping réorganise la structure des documents produits par le connecteur source pour que le connecteur destination puisse les consommer.
- un connecteur destination consomme des documents de type XML.
Chaque document correspond à un ensemble de données de l'application destination.

Pour un connecteur de type base de données, chaque document correspond à un enregistrement d'une table (et éventuellement à des liens vers d'autres tables). Exemple : pour une base de données AssetCenter, un document contient les données de la table des biens.

Pour les autres connecteurs, les ensembles de données peuvent correspondre à des fichiers texte délimité, des messages e-mail, des informations de sécurité, etc.

Le choix des connecteurs, leur liaison, la définition de mappings entre types de document source et type de document destination constituent un scénario d'intégration. Après une phase de test et de débogage, un scénario est associé à une programmation et mis en exploitation.

Comme dans tout processus d'intégration, des documents sont rejetés partiellement ou en totalité. L'utilisation de bilans de traitement et d'un journal des documents permet de retraiter ces documents rejetés sans avoir à redéfinir l'ensemble de votre scénario d'intégration.

2 | Installation

CHAPITRE

Ce chapitre contient des informations vous permettant d'installer la plateforme d'intégration Connect-It.

Environnements supportés

Connect-It est compatible Windows.

Connect-It peut être installé également sous Unix (Linux, Solaris, AIX) mais ne fonctionne qu'en mode non graphique.

Connect-It doit être installé sur un poste ayant accès aux applications concernées par les scénarios que vous souhaitez utiliser. Ce poste n'a généralement pas besoin de l'installation complète de la partie cliente des applications concernées par les scénarios. Par contre, cela peut être utile, notamment en phase de test, pour vérifier le résultat de l'écriture des données dans les applications externes.

La gestion du suivi des documents implique la sauvegarde sur votre disque dur des éléments suivants :

- les documents (fichiers **.dat**)
- les lignes de suivi accompagnées de messages explicatifs (fichiers **.msg**)

- une indexation complète de ces documents vous permettant d'y accéder rapidement (fichier **.idx**)

La mémoire allouée au journal de l'application peut être fixée grâce à la commande **Options** dans le menu **Edition** de l'éditeur de scénarios. Par défaut, la mémoire allouée au journal de l'application est fixée à 5 Mo.

Pour une présentation complète du journal des documents et de la gestion du suivi des documents, consultez le chapitre [Editeur de scénarios](#) [page 31], section [Journaux](#) [page 54], sous-section [Journal des documents](#) [page 56].

Installer Connect-It

Vous pouvez installer Connect-It sous Windows 32 bit ou UNIX.

Pour installer Connect-It sous Windows

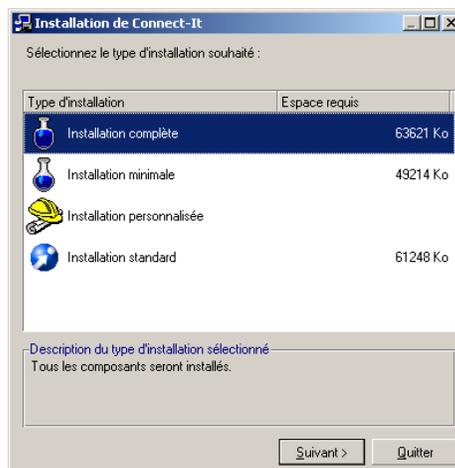
- 1 Vérifiez que vous disposez de l'environnement requis
- 2 Démarrez Windows avec des droits d'administration

Après avoir inséré le CD ROM dans votre lecteur, le programme d'installation se lance automatiquement.

Si ce n'est pas le cas, ouvrez le dossier d'installation de Connect-It et lancez le programme **setup.exe**.

Le programme d'installation vous offre le choix entre quatre installations différentes :

- Une installation complète
- Une installation minimale
- Une installation personnalisée
- Une installation standard



Installation minimale

L'installation minimale comprend les composants suivants :

- Editeur de scénarios (Connect-It)
- Console Connect-It qui vous permet de créer des services associés à vos scénarios
- Visionneuse de fichiers journaux qui vous permet de lire les fichiers LOG
- Connecteurs de base
- Kit de données Connect-It comprenant notamment :
 - les scénarios fournis liés aux connecteurs installés
 - les fichiers associés nécessaires à l'utilisation des scénarios fournis

Installation standard

L'installation standard ajoute à l'installation minimale la documentation Connect-It au format PDF et CHM.

Installation complète

L'installation complète ajoute à l'installation standard les composants suivants :

- **Connecteurs optionnels** [page 26]
Vous pouvez utiliser ces connecteurs optionnels si votre certificat d'autorisation vous en donne le droit.
- Les scénarios fournis liés aux connecteurs optionnels

Installation personnalisée

Cette installation personnalisée vous permet de procéder à une installation minimale et d'ajouter les connecteurs optionnels de votre choix.

L'installation d'un composant optionnel s'accompagne de celle des scénarios fournis l'utilisant.

Pour installer Connect-It sous UNIX

Pour installer Connect-It sous UNIX, vous devez décompresser le fichier **.tgz** fourni en exécutant la ligne de commande suivante :

```
tar xvfz cnit-[système d'exploitation]_[version].[build]_[préfixe de la langue].tgz
```

Les préfixes disponibles sont les suivants : **fr** pour la version française, **en** pour la version anglaise, **it** pour la version italienne, **es** pour la version espagnole et **de** pour la version allemande.

L'installation de Connect-It sous UNIX est toujours une installation complète. Cette installation comprend les composants suivants :

- Connecteurs de base [page 25]
- Connecteurs optionnels [page 26]
- Kit de données [page 27]
- Documentation
- Les scénarios d'exemple

Note :

Certains connecteurs ne fonctionnent que sous un environnement Windows 32 bit. Pour un détail complet des compatibilités des connecteurs, nous vous invitons à consulter la matrice de compatibilité présente sur le site support de Peregrine Systems, à l'adresse suivante : <http://support.peregrine.com>

Librairies dynamiques (.so) sous UNIX

Connect-It nécessite l'utilisation de librairies dynamiques (.so) localisées dans le dossier **bin** du dossier d'installation de Connect-It. Le chemin **[dossier d'installation de Connect-It]/bin** doit être ajouté à la variable d'environnement **LD_LIBRARY_PATH**.

Pour un interpréteur de commande C (csh), exécutez la ligne de commande suivante :

```
setenv LD_LIBRARY_PATH=[dossier d'installation de Connect-It]/bin
```

Pour un interpréteur de commande K (ksh) ou Bourne (sh), exécutez la ligne de commande suivante dans le fichier **.profile**:

```
LD_LIBRARY_PATH=[dossier d'installation de Connect-It]/bin export LD_LIBRARY_PATH
```

Saisir le certificat d'autorisation

Quand vous avez terminé d'installer Connect-It, vous devez saisir dans l'éditeur de scénarios, le certificat d'autorisation fourni par Peregrine Systems.

Ce certificat d'autorisation est un fichier texte contenant :

- la liste des connecteurs optionnels que vous avez le droit d'utiliser
- la date limite d'utilisation de chaque connecteur optionnel
- une clé d'autorisation cryptée

Pour saisir le certificat d'autorisation

- 1 Lancez l'éditeur de scénarios de Connect-It.
- 2 Sélectionnez **Editer le certificat d'autorisation** dans le menu **Administration**.
- 3 Coller votre certificat d'autorisation dans la boîte de dialogue qui s'affiche.
- 4 Cliquez sur **OK**.

Cette action crée un fichier **license.txt** dans le dossier d'installation de Connect-It.

 **Note :**

Dans sa version Unix, Connect-It ne possède pas d'interface graphique. La saisie du certificat d'autorisation s'effectue suivant la procédure décrite ci-dessous :

- 1 Copiez le contenu du fichier de licence fourni par Peregrine Systems dans un nouveau fichier texte. Ne modifiez pas le contenu de ce fichier.
- 2 Sauvegardez ce fichier sous le nom **license.txt**
- 3 Copiez le fichier **license.txt** dans le dossier d'installation de Connect-It

Contenu de l'installation

Cette section détaille les composants et les fichiers installés.

La présence de ces fichiers sur votre ordinateur dépend du type d'installation choisi : installation minimale, complète ou personnalisée.

Fichiers du dossier d'installation

Le tableau suivant vous présente l'organisation des fichiers du dossier d'installation de Connect-It

Nom des dossiers	Contenu principal
bin32	<ul style="list-style-type: none"> • Exécutables (conitgui.exe, conitsvc.exe, console.exe, logview.exe) • Librairies dynamiques (DLL) • Fichier de configuration de l'éditeur de texte propre à Connect-It (codeedit.cfg) • Divers fichiers nécessaires au fonctionnement des scénarios fournis et au fonctionnement de Connect-It.
config	<p>Fichiers associés aux connecteurs dans tout scénario :</p> <ul style="list-style-type: none"> • .mpt (tables de correspondance) • .str (tableau de chaînes de caractères) • .bas (Fonctions Basic et variables globales)

Nom des dossiers	Contenu principal
datakit	Kit de données utilisées par les outils de Connect-It
doc	Documentations au format .pdf et .chm
fsf	Répertoire de stockage des copies de sauvegarde des fichiers .fsf .
scenario	<ul style="list-style-type: none"> Fichiers SCN contenant les scénarios d'exemple de Connect-It Fichiers associés aux connecteurs pour les scénarios d'exemple.
lib	Fichiers JAR contenant : <ul style="list-style-type: none"> classes Java développées par Peregrine Systems classes Java tierces
wizards	Fichiers nécessaires au fonctionnement de Connect-It.
Studio	Fichiers nécessaires au fonctionnement du composant Studio de Connect-It.

Connecteurs de base

Les connecteurs de base sont les connecteurs installés et utilisables quel que soit le mode d'installation de Connect-It.

Les connecteurs de base sont :

- Connecteur Asset Management
- Connecteur Action Request System
- Connecteur Ligne de commande
- Connecteur InfraTools Desktop Discovery
- Connecteur InfraTools Network Discovery
- Connecteur Peregrine Desktop Inventory
- Connecteur Desktop Administration
- Connecteur InfraTools Management
- Connecteur FacilityCenter
- Connecteur Sécurité NT
- Connecteurs passerelles 3.x
- Connecteur PDA Inventory
- Connecteur ServiceCenter

- Connecteur TeleCenter
- Connecteur Texte délimité
- Connecteur XML

Connecteurs optionnels

Les connecteurs optionnels sont installés si vous choisissez une installation complète de Connect-It ou une installation personnalisée durant laquelle vous choisissez les connecteurs optionnels de votre choix.

Les connecteurs optionnels sont :

- Connecteur Asset Insight
- Connecteur Base de données
- Connecteurs e-mail
- Connecteur d'écoute XML
- Connecteurs Intel LANDesk
- Connecteur LDAP
- Connecteur Lotus Notes
- Connecteur MQSeries
- Connecteur d'écoute SCAuto
- Connecteurs SMS 1.x et SMS 2.x
- Connecteur Tivoli Enterprise Console (Envoi)
- Connecteur Tivoli Enterprise Console (Réception)
- Connecteur TS.Census
- Connecteur Tivoli Inventory (versions 3.1 et 3.6)
- Connecteur Tivoli Inventory CM for Inventory 4.2
- Connecteur Unicenter AMO
- Connecteur Winpark Actima
- Connecteur Web Services
- Connecteur JDBC
- Connecteur SAP BAPI
- Connecteur SAP IDoc

Connecteurs additionnels

Ce sont des connecteurs qui ne sont pas installés par défaut avec l'éditeur de scénario. Vous devez les installer séparément à l'aide d'un CD-ROM fourni par Peregrine Systems. Pour chaque connecteur additionnel, un certificat d'autorisation vous est délivré par Peregrine Systems.

Kit de données

Le tableau suivant détaille le contenu du kit de données installé.

Kit de données

Nom des fichiers	Fonction
Dossier idd *.fsf	Donnent des exemples d'inventaire réalisé sur un ordinateur grâce à Desktop Discovery. Un inventaire contient des informations sur l'ordinateur lui-même (Hardware), les logiciels installés sur cet ordinateur (Software) et les utilisateurs et les biens associés à cet ordinateur (UserAndAssets).
master.sai, french.sai	Permettent à Desktop Discovery de trouver la description d'un grand nombre d'applications informatiques. Pour chaque application, ce fichier contient la version, l'éditeur, le numéro de licence... Sans cette description, les logiciels sont présents sous la forme de noms de fichiers dans les fichiers .fsf .
iddsdt.cdt	Contient un type de document correspondant à l'inventaire proposé par défaut par le générateur d'inventaires (Scanner Generator) de Desktop Discovery.
datakit.cdt	Contient un type de document correspondant à un inventaire de démonstration.
Dossiers scac36 et scac41	

Kit de données

modelfeat.scr	Ce fichier de script permet de créer des caractéristiques dans une application Asset Management. Ces caractéristiques sont associées à des champs standards de ServiceCenter dans le scénario de réplication ServiceCenter-AssetCenter.
modelfeat.txt	
Dossier ac	
Sous-dossier stdfeat	En important le fichier de script (sdtfeat.scr), vous créez dans l'application Asset Management des enregistrements dans la table des caractéristiques. Ces caractéristiques apparaissent dans les types de document publiés par le connecteur Asset Management.
Dossier tc	Contient l'ensemble des fichiers permettant d'utiliser les scénarios fournis impliquant le connecteur TeleCenter.
Dossier ind	
Sous-dossier ac	Ce dossier contient des scripts d'import permettant de créer des catégories et des caractéristiques dans une application Asset Management.
Dossier doctrans	Ce dossier contient les fichiers nécessaires à la génération automatique de la documentation des scénarios.

Fichiers associés

Certains fichiers (**.mpt**, **.str**, **.bas**) sont associés à un scénario. Si le scénario doit être changé de répertoire, ces fichiers doivent être déplacés avec leur dossier respectif dans la même position relative qu'ils avaient par rapport au scénario. Exemple : les fichiers communs à plusieurs scénarios d'exemple se trouvent dans le dossier **shared**. Ce dossier doit être créé à l'endroit où vous voulez déplacer votre scénario.

 **Note :**

Plutôt que de déplacer un scénario par l'intermédiaire d'un gestionnaire de fichier, il est toujours préférable, dans la mesure du possible, de sauvegarder un scénario depuis l'éditeur de scénario vers la destination désirée. Cela garantit que tous les fichiers associés précédemment au scénario restent correctement référencés.

Informations sur Connect-It

Connect-It vous permet d'accéder à l'ensemble des informations relatives au logiciel et à son environnement de travail.

Pour accéder aux informations :

- Choisissez **Aide/ A propos**.
- Cliquez sur **Plus**.

Une fenêtre s'affiche récapitulant les informations :

- logicielles
- système
- relatives aux moteurs de base de données

3 Éditeur de scénarios

CHAPITRE

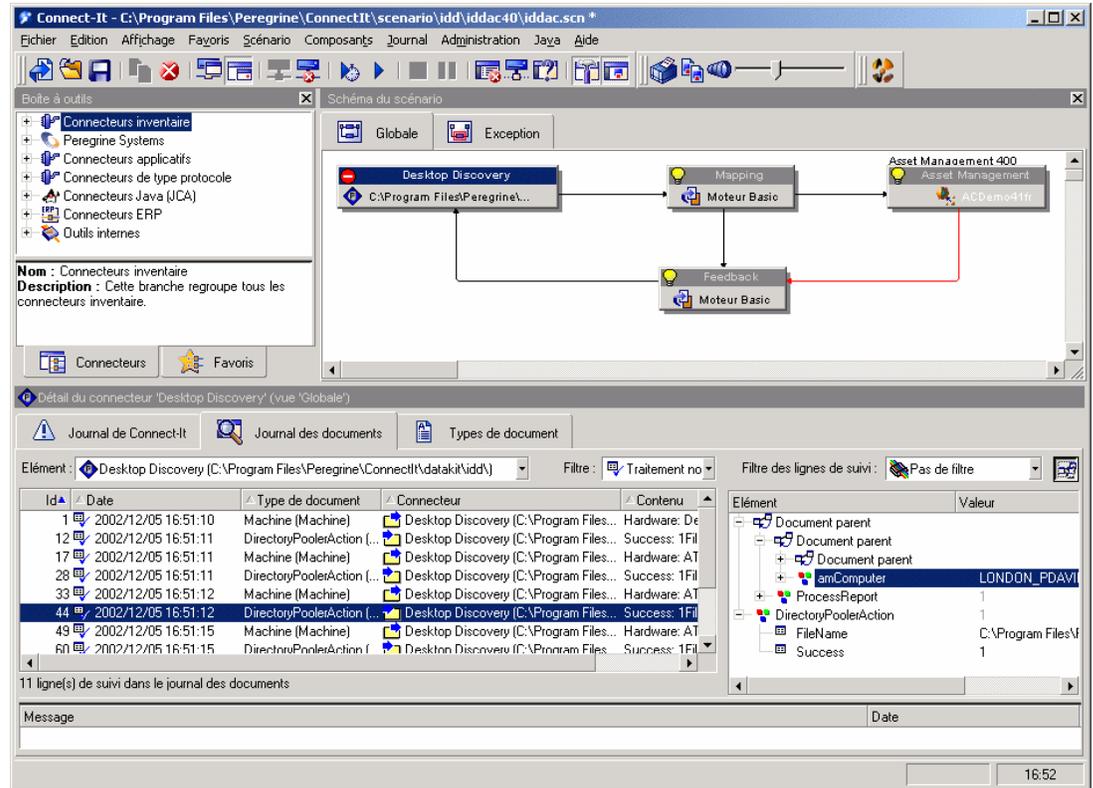
L'éditeur de scénarios est l'interface graphique qui vous permet de créer des scénarios d'intégration. Ce chapitre vous présente l'ergonomie des principaux éléments de cet éditeur ainsi que la fenêtre d'édition des options de Connect-It.

Fenêtre principale

La fenêtre principale de l'éditeur de scénarios se compose :

- d'une barre de menus
Pour une présentation complète de ces menus, consultez dans ce chapitre, la section [Menus](#) [page 40].
- de barres d'outils
- de volets

Figure 3.1. Editeur de scénarios - fenêtre principale



Barres d'outils

Les barres d'outils de l'éditeur de scénarios vous permettent d'effectuer les opérations les plus courantes sans avoir à passer par la barre de menus. Les barres d'outils disponibles sont les suivantes :

- Barres d'outils (générique)
- Barre du schéma (du scénario)
- Barre des vues
- Barre des favoris
- Barre du Studio

Tableau 3.1. Editeur de scénarios - icônes de la barre d'outils générique

Icônes	Fonctions
	Lance un assistant qui vous permet de créer un scénario minimal avec un connecteur source et un connecteur destination.
	Ouvre un scénario existant.
	Enregistrer le scénario courant.
	Permet de copier les éléments textuels de Connect-It : <ul style="list-style-type: none"> • Le nom des connecteurs dans la boîte à outils • Un ligne de de message dans les journaux • Etc.
	Supprime le composant du scénario sélectionné.
	Lance l'assistant de configuration du composant sélectionné.
	Permet à l'assistant de configuration des composants d'afficher des pages de configuration avancée. Exemple : la page Utiliser les types de document pivots.
	Ouvre tous les connecteurs du scénario.
	Ferme tous les connecteurs du scénario.
	Lance le scénario en mode programmé.
	Teste le scénario (mode non programmé)
	Arrête le scénario courant.
	Permet de mettre le scénario en mode pause
	Purge les lignes des onglets Journal de Connect-It et Journal des documents.
	Passe en mode hors connexion/avec connexion
	Permet de tester le scénario sans que les connecteurs destination interagissent avec l'application externe. Pour les connecteurs de type Base de données supportant les transactions, des messages d'erreur sont obtenus et correspondent à ceux qui seraient normalement obtenus si cette option n'était pas sélectionnée.
	Affiche ou masque le volet Boîte à outils.

Icônes	Fonctions
	Affiche ou masque le volet Schéma du scénario .

Tableau 3.2. Editeur de scénarios - icônes de la barre d'outils des favoris

Icônes	Fonctions
	Permet de reconfigurer un favori sélectionné dans l'onglet Favori de la boîte à outils.
	Permet de supprimer un favori sélectionné dans l'onglet Favori de la boîte à outils.
	Permet d'importer le fichier Favorite.fav dans le dossier de votre choix.
	Permet d'exporter le fichier Favorite.fav dans le dossier de votre choix. Il suffit ensuite qu'un autre utilisateur importe ce fichier dans le dossier bin32 de son dossier d'installation de Connect-It pour utiliser ces favoris.

Tableau 3.3. Editeur de scénarios - icônes de la barre d'outils du schéma du scénario

Icônes	Fonctions
	Imprime le schéma du scénario. (Une fenêtre d'aperçu avant impression vous permet de paramétrer cette impression).
	Copie le schéma du scénario dans le presse-papiers.
	La tirette placée à côté de cette icône permet d'agrandir le schéma du scénario.

Tableau 3.4. Editeur de scénarios - icônes de la barre des vues

Icône	Fonction
	Avant la création d'un mapping, permet de sélectionner la vue dans laquelle les connecteurs source et destination du mapping apparaîtront une fois le mapping créé.

Tableau 3.5. Editeur de scénarios - icônes de la barre d'outils du Studio

Icône	Fonction
	Permet de synchroniser le scénario courant avec un projet du Studio. Pour plus une présentation de cette fonction, consultez le manuel Studio

Pour afficher ou masquer une barre d'outils

- 1 Sélectionnez le menu **Affichage**.
- 2 Sélectionnez ou désélectionnez une des barres d'outils suivantes :
 - Barres d'outils (générique)
 - Barre du schéma (du scénario)
 - Barre du Studio
 - Barre des vues

Pour déplacer une barre d'outils

- 1 Placez le pointeur de la souris sur la poignée de manipulation (☰) de la barre d'outils.
- 2 Cliquez sur cette poignée et attendez qu'un rectangle noir reprenant la forme de la barre d'outil apparaisse.
- 3 Déplacez le rectangle noir à l'endroit de votre choix en maintenant votre doigt appuyé sur le bouton de la souris.
- 4 Relâchez le bouton de la souris après que le rectangle noir symbolisant la barre d'outils est placé à l'endroit désiré.

Vous pouvez déplacer une barre d'outils de son emplacement initial pour :

- l'encadrer dans les quatre coins de la fenêtre de l'éditeur de scénarios.

- la placer en mode flottant dans n'importe quel endroit de votre écran.

Pour encastrer une barre d'outils flottante

- 1 Placez votre pointeur sur un des bords de la barre d'outils flottante.
 - 2 Attendez que le pointeur se transforme ()
 - 3 Double-cliquez.
- La barre d'outils s'encastre automatiquement à son emplacement initial.

Volets

La fenêtre principale de l'éditeur de scénarios se divise en trois volets redimensionnables :

- Une boîte à outils vous permettant de choisir les composants ou les favoris de votre scénario
- Un volet de schéma du scénario où sont représentés graphiquement les composants et les liens de votre scénario
- Un volet composé de différents onglets :
 - Onglet du journal de Connect-It
 - Onglet du journal des documents
 - Onglets relatifs au composant sélectionné dans le schéma du scénario : types de document produits ou consommés, mappings, mappings pivots

Pour afficher ou masquer un volet

Vous pouvez afficher ou masquer les volets suivants :

- Le volet de la boîte à outils
Pour afficher ou masquer ce volet, effectuez une des actions suivantes :
 - Sélectionnez ou désélectionnez **Affichage/ Boîte à outils**.
 - Cliquez sur .
- Le volet de schéma du scénario
Pour afficher ou masquer ce volet, effectuez une des actions suivantes :
 - Sélectionnez ou désélectionnez **Affichage/ Schéma du scénario**.
 - Cliquez sur .

Pour déplacer un volet

- 1 Placez le pointeur de la souris sur le bord supérieur du volet.
- 2 Cliquez sur ce bord gris foncé et attendez qu'un rectangle noir reprenant la forme du volet apparaisse.
- 3 Déplacez le rectangle noir à l'endroit de votre choix en maintenant votre doigt appuyé sur le bouton de la souris.
- 4 Relâchez le bouton de la souris après que le rectangle noir symbolisant le volet est placé à l'endroit désiré.

Vues d'un schéma de scénario

Le volet **Schéma du scénario** comprend deux onglets : **Globale** et **Exception**. Le premier onglet affiche tous les composants du scénario courant : connecteurs source et destination, liens et boîtes de mapping. Le second onglet n'affiche par défaut aucune information mais il est destiné à la gestion des erreurs du scénario.

Vous pouvez créer différentes vues dans le volet **Schéma du scénario** permettant de n'afficher que les composants utilisés par un ou plusieurs mappings donnés.

Exemple : dans un scénario utilisant plusieurs connecteurs source, vous sélectionnez les mappings incluant un seul connecteur source afin de créer une vue n'affichant que ce seul connecteur source.

Utiliser les vues d'un schéma de scénario

Les vues créées pour un scénario apparaissent chaque fois que vous ouvrez ce scénario dans l'éditeur de scénarios.

Pour créer une vue d'un schéma de scénario

- 1 Ouvrez votre scénario.
- 2 Choisissez **Scénario \ Gestion des vues**.
- 3 Cliquez sur  dans la fenêtre qui s'affiche.
- 4 Nommez votre vue. Ce nom doit être unique pour un scénario.
- 5 Sélectionnez les mappings pour lesquels vous souhaitez obtenir une vue du schéma du scénario.
- 6 Cliquez sur le bouton **Créer**.

Un onglet portant le nom de votre vue apparaît dans le volet **Schéma du scénario** après la fermeture de l'éditeur de vues.

Pour modifier une vue d'un schéma de scénario

- 1 Ouvrez votre scénario.
- 2 Choisissez **Scénario\ Gestion des vues**.
- 3 Sélectionnez votre vue.
- 4 Renommez votre vue ou sélectionnez, désélectionnez les mappings de votre choix.
- 5 Cliquez sur **Modifier**.
- 6 Cliquez sur **Fermer**.

Pour supprimer la vue d'un schéma de scénario

- 1 Ouvrez votre scénario.
 - 2 Choisissez **Scénario\ Gestion des vues**.
 - 3 Sélectionnez votre vue.
 - 4 Cliquez sur .
- L'onglet correspondant à cette vue disparaît du volet **Schéma du scénario**.

Imprimer les vues d'un schéma de scénario

Vous pouvez imprimer chacune des vues d'un schéma de scénario.

Pour imprimer une vue d'un schéma de scénario

- 1 Sélectionnez une vue.
Exemple : **Globale**.
- 2 Effectuez une des actions suivantes :
 - Cliquez sur .
 - Choisissez **Fichier/ Imprimer le schéma du scénario**.
- 3 Attendez qu'une fenêtre d'aperçu avant impression s'affiche.
- 4 Cliquez sur **Imprimer**.

Pour configurer l'impression d'une vue

- 1 Lancez l'impression d'une vue.
- 2 Cliquez sur **Configurer**.

- 3 Attendez que le panneau de Configuration de l'impression de Windows s'affiche.
- 4 Changez les paramètres de configuration.

Pour changer la mise en page

- 1 Lancez l'impression d'une vue.
- 2 Cliquez sur **Mise en page**.
- 3 Attendez que la boîte de dialogue **Mise en page** s'affiche.
- 4 Modifiez les paramètres de mise en page :
 - Marge gauche
 - Marge haute
 - Marge droite
 - Marge basse
 - Marge en-tête
 - Marge pied de page
 - Zoom

L'unité utilisée est le centimètre.

- Numéro de page (#)
- Date courante (**date**)
- Heure courante (**H**)
- Chemin complet du fichier **.scn**

Les boutons **En-tête** et **Pied de page** affichent de nouvelles boîtes de dialogue vous permettant de définir les informations imprimées dans le pied de page et l'en-tête de votre impression :

Le bouton **A** vous permet de choisir la police de caractères des informations imprimées.

Vues d'un schéma d'un scénario - règles de fonctionnement

Les vues de schéma d'un scénario obéissent aux règles suivantes :

- Il est impossible de glisser-déplacer un composant à l'intérieur d'une vue autre que **Globale**.
- La suppression d'un mapping sélectionné dans une vue ne supprime pas l'élément < LAYER > du fichier SCN correspondant.

- L'édition des des mappings, des connecteurs et des liens n'est possible que dans la vue **Globale**.
- Le clic sur un composant dans une vue filtre les lignes de suivi associées dans le journal des documents.
- Les vues n'ont pas d'impact sur les autres fonctionnalités de Connect-It.

Menus

Cette section vous présente :

- La barre de menus de l'éditeur de scénarios
- Les menus contextuels du schéma du scénario

Barre de menus de l'éditeur de scénarios

Menu Fichier

Le tableau suivant détaille la liste des commandes du menu **Fichier**.

Tableau 3.6. Fonctions des commandes du menu Fichier

Commande	Fonction
Nouveau	Lance l'assistant 'Configurer le scénario'. Cet assistant vous permet de choisir et configurer un connecteur source et un connecteur destination. Il les relie et place entre ces deux connecteurs une boîte de mapping.
Ouvrir	Ouvre un fichier de scénario (fichiers SCN) déjà créé.
Enregistrer...	Enregistre le scénario courant
Enregistrer sous...	Permet d'enregistrer le scénario courant sous un nom différent de celui avec lequel il a été ouvert
Historique du scénario	Affiche une fenêtre vous permettant de saisir des informations sur le scénario courant.
Imprimer le schéma du scénario	Affiche une fenêtre vous permettant de saisir des paramètres d'impression du schéma du scénario.

Commande	Fonction
Copier le schéma du scénario	Copie l'image du schéma du scénario dans le presse-papiers. Vous pouvez ensuite coller cette image dans une application graphique comme Jasc Paint Shop Pro, Adobe Photoshop.
Créer une documentation de scénario	Permet de créer une documentation du scénario courant au format XML, DBK ou HTML. Pour plus d'informations sur cette fonction, consultez le chapitre Documentations des scénarios [page 171], section Création d'une documentation de scénario [page 178].
Voir la documentation de scénario HTML	Permet d'afficher une documentation du scénario courant dans un navigateur internet. Pour plus d'informations sur cette fonction, consultez le chapitre Documentations des scénarios [page 171], section Visualisation de la documentation d'un scénario au format HTML [page 174].
Propriétés des documentations de scénarios	Pour plus d'informations sur cette fonction, consultez le chapitre Documentations des scénarios [page 171], section Editer les propriétés des documentations de scénario [page 176]. Lance la fenêtre des propriétés des documentations de scénario.
Quitter	Permet de quitter l'éditeur de scénarios
Liste numérotée	Présente les derniers scénarios édités. En cliquant sur l'un des éléments de la liste, vous ouvrez directement un scénario.

Menu Edition

Le tableau suivant détaille la liste des commandes du menu **Edition**.

Tableau 3.7. Fonctions des commandes du menu Edition

Commande	Fonction
Couper	Aucune

Commande	Fonction
Copier	Permettent de copier dans le presse-papiers de votre système d'exploitation des lignes du journal de Connect-It ou du journal des documents.
Coller	Aucune
Effacer	Permet d'effacer du scénario courant un composant préalablement sélectionné.
Options...	Affiche la fenêtre des options de l'éditeur de scénarios.

Menu Affichage

Le tableau suivant détaille la liste des commandes du menu **Affichage**

Tableau 3.8. Fonctions des commandes du menu Affichage

Commande	Fonction
Boîtes à outils	Affiche ou masque le volet contenant la liste des connecteurs.
Schéma du scénario	Affiche ou masque le volet du schéma du scénario.
Barre d'outils	Affiche ou masque la barre d'outils générale.
Barre du schéma	Affiche ou masque la barre d'outils du volet du schéma.
Barre des favoris	Affiche ou masque la barre d'outils des favoris.
Barre du Studio	Affiche ou masque la barre d'outils qui vous permet de synchroniser votre scénario d'intégration avec un projet Studio. Pour plus d'informations sur cette fonction, consultez la documentation Peregrine Studio pour Connect-It 3.1 .

Menu Scénario

Le tableau suivant détaille la liste des commandes du menu **Scénario**.

Tableau 3.9. Fonctions des commandes du menu Scénario

Commande	Fonction
Ouvrir tous les connecteurs	Ouvre tous les connecteurs et boîtes de mapping du scénario courant
Fermer tous les connecteurs	Ferme tous les connecteurs et boîtes de mapping du scénario courant
Montrer les connecteurs internes	Montre les connecteurs internes créés automatiquement par Connect-It quand les types de document pivots sont utilisés.
Synchroniser les caches	Synchronise les fichiers de caches de tous les connecteurs ouverts du scénario courant
Effacer les caches	Efface les fichiers de caches de tous les connecteurs ouverts du scénario courant
Gestion des vues	Affiche l'éditeur de vues. Pour une présentation de la gestion des vues, consultez dans ce chapitre, la section Fenêtre principale [page 31] / Vues d'un schéma de scénario [page 37] .
Programmateurs	Affiche la fenêtre d'édition des programmeurs
Programmation	Affiche la fenêtre d'édition de la programmation
Tables de chaînes	Affiche la fenêtre d'édition des fichiers de tables de chaînes de caractères
Tables de correspondance	Affiche la fenêtre d'édition des fichiers de tables de correspondance
Fonctions globales	Affiche la fenêtre d'édition des fonctions et variables globales
Formats utilisateur	Permet de créer des formats de type date et numérique pouvant être utilisés dans les scripts de mappings.
Travailler hors connexion	Permet de travailler hors connexion. Cette option peut être activée en cliquant sur  dans la barre d'outils. Pour plus d'informations sur le travail hors connexion, consultez le chapitre Test et débogage d'un scénario d'intégration [page 163] , section Travailler hors connexion [page 167] .

Commande	Fonction
Mode test du scénario	Permet de tester le scénario courant sans que les connecteurs destination ne transmettent les données vers les applications externes. Cette option peut être activée en cliquant sur  dans la barre d'outils.
Démarrer tous les programmeurs	Démarre les programmeurs associés au type de documents produits par les connecteurs source du scénario courant
Arrêter	Arrête le scénario.
Pause	Effectue une pause dans l'exécution du scénario.

Menu Composant

Le tableau suivant détaille la liste des commandes du menu **Composant**.

Tableau 3.10. Fonctions des commandes du menu Composants

Commande	Fonction
Configurer	Lance l'assistant de configuration du composant sélectionné.
Configuration avancée	Permet d'afficher les pages de configuration avancée de l'assistant de configuration du composant (F2). Cette option peut être activée en cliquant sur  dans la barre d'outils.
Ouvrir	Ouvre le composant sélectionné
Fermer	Ferme le composant sélectionné
Cache/Synchroniser le cache	Synchronise le fichier de cache du connecteur sélectionné
Cache/Effacer le cache	Efface le fichier de cache du connecteur sélectionné
Editer un type de document	Lance l'assistant Sélectionner un type de document vous permettant de choisir un type de document produit ou consommé par le composant sélectionné
Editer un mapping	Lance l'assistant Sélectionner un mapping vous permettant de choisir un des mappings de la boîte de mapping sélectionnée

Commande	Fonction
Produire	Permet au composant sélectionné de produire les documents pour lequel un type de document a été créé
Démarrer/ Synchroniser Studio	Lance Studio et synchronise le scénario courant avec le projet qui lui est lié dans Studio. Pour plus d'informations sur cette fonction, consultez la documentation Peregrine Studio pour Connect-It 3.1.

Menu Journal

Le tableau suivant détaille la liste des commandes du menu **Journal**.

Tableau 3.11. Fonctions des commandes du menu Journal

Commande	Fonction
Configurer le journal des documents...	Affiche la fenêtre vous permettant de configurer le journal des documents
Rafraîchir le journal des documents	Permet de rafraîchir le contenu du journal des documents
Recharger le journal des documents	Permet d'afficher dans l'onglet Journal des documents , l'ensemble des lignes de suivi contenues dans le fichier de sauvegarde du journal des documents.
Effacer les lignes de suivi affichées	Efface de l'onglet Journal des document l'ensemble des lignes de suivi affichées.
Supprimer les lignes de suivi stockées	Efface le contenu du fichier de sauvegarde du journal des documents

Menu Administration

Le tableau suivant détaille la liste des commandes du menu **Administration**.

Tableau 3.12. Fonctions des commandes du menu Administration

Commande	Fonction
Editer le certificat d'autorisation...	Affiche la fenêtre d'édition de votre certificat d'autorisation.

Menu Java

Le tableau suivant détaille la liste des commandes du menu **Java**.

Tableau 3.13. Fonctions des commandes du menu Java

Commande	Fonction
Configurer la JVM	Affiche une fenêtre vous permettant d'indiquer le chemin de vos classes Java.
Déployer un connecteur	Affiche la fenêtre de déploiement des connecteurs Java que vous créez.

Menu Aide

Le tableau suivant détaille la liste des commandes du menu **Aide**.

Tableau 3.14. Fonctions des commandes du menu Aide

Commande	Fonction
Aide en ligne	Affiche l'aide en ligne de Connect-It
Astuce du jour	Affiche la boîte des astuces du jour.
A propos...	Affiche la fenêtre A propos contenant les informations générales sur votre version de Connect-It

Menus contextuels dans la fenêtre principale de l'éditeur de scénario

Dans la fenêtre principale de l'éditeur de scénarios des menus contextuels sont disponibles dans les situations suivantes :

- Pointeur placé dans le volet du schéma du scénario
- Pointeur placé dans l'onglet du journal de Connect-It
- Pointeur placé dans l'onglet du journal des documents

Pointeur placé dans le volet du schéma du scénario

Le tableau suivant détaille la liste des commandes du menu contextuel s'affichant que votre pointeur est placé dans le volet du schéma du scénario.

Commande	Fonction
Configurer le connecteur	Lance l'assistant Configuration du connecteur vous permettant de configurer le composant sélectionné
Ouvrir le connecteur	Ouvre le composant sélectionné
Fermer le connecteur	Ferme le composant sélectionné
Cache/Synchroniser le cache	Synchronise le fichier de cache du connecteur sélectionné
Cache/Effacer le cache	Efface le fichier de cache du connecteur sélectionné
Editer un type de document	Lance l'assistant Sélectionner un type de document vous permettant de choisir un type de document produit ou consommé par le composant sélectionné
Editer un mapping...	Lance l'assistant Sélectionner un mapping vous permettant de choisir un des mappings de la boîte de mapping sélectionnée
Produire	Permet au composant sélectionné de produire les documents pour lequel un type de document a été créé.
Effacer	Permet de supprimer le composant sélectionné.
Afficher les lignes de suivi	Affiche dans le journal des documents les lignes de suivi relatives au composant sélectionné
Montrer la boîte à outils	Affiche ou cache la boîte à outils
Montrer le schéma du scénario	Affiche ou cache le schéma du scénario

Pointeur placé dans l'onglet du journal de Connect-It

Commande	Fonction
Déplier un niveau	Déplie un niveau sélectionné
Déplier tous les niveaux	Déplie tous les niveaux
Replier tous les niveaux	Replie tous les niveaux
Montrer la boîte à outils	Affiche ou cache la boîte à outils
Montrer le schéma du scénario	Affiche ou cache le schéma du scénario

Pointeur placé dans l'onglet du journal des documents

Deux menus contextuels sont disponibles dans le journal des documents :

- Pointeur placé sur une ligne de suivi

- Pointeur placé sur un élément du détail d'un document source

Pointeur placé sur une ligne de suivi

Commande	Fonction
Copier ce document XML	Copie le document XML correspondant à la ligne de suivi dans le presse-papiers de votre système d'exploitation
Copier cette DTD XML	Copie la DTD XML du document correspondant à la ligne de suivi dans le presse-papiers de votre système d'exploitation
Ouvrir ce document XML	Ouvre le document avec l'application associée aux fichiers XML, spécifiée dans votre système d'exploitation (Exemple : Internet Explorer)
Montrer la boîte à outils	Affiche ou cache la boîte à outils
Montrer le schéma du scénario	Affiche ou cache le schéma du scénario

Pointeur placé sur un élément du détail d'un document source

Commande	Fonction
Copier ce document XML	Copie le document XML dans le presse-papiers de votre système d'exploitation
Copier cette DTD XML	Copie la DTD XML du document dans le presse-papiers de votre système d'exploitation
Ouvrir ce document XML	Ouvre le document avec l'application associée aux fichiers XML, spécifiée dans votre système d'exploitation (Exemple : Internet Explorer)
Copier le chemin	Copie dans le presse-papiers de votre système d'exploitation le chemin de l'élément sélectionné
Montrer la boîte à outils	Affiche ou cache la boîte à outils
Montrer le schéma du scénario	Affiche ou cache le schéma du scénario

Favoris

Les favoris sont des connecteurs préconfigurés. Ils sont visibles dans le volet **Favoris** de la boîte à outils de l'éditeur de scénarios.

Un favori utilisé (par glisser-déplacer) dans un scénario aboutit à la création d'une instance de connecteur associé à ce favori. Vous pouvez ensuite supprimer l'association entre cette instance de connecteur et ce favori.

Quand un favori est associé à un ou plusieurs connecteurs :

- la reconfiguration du favori entraîne automatiquement la reconfiguration des connecteurs qui lui sont associés dans l'ensemble des scénarios.

Avertissement :

Le nom et la description d'un favori lui sont propres. Quand vous modifiez ce nom et cette description dans l'assistant de configuration, ces changements ne sont pas répercutés dans la configuration des connecteurs associés.

Pour que cette mise à jour soit effective, il faut néanmoins ouvrir de nouveau ces scénarios à l'aide de l'éditeur de scénarios.

- la reconfiguration d'un connecteur associé peut entraîner la reconfiguration du favori et des autres connecteurs associés.

A la fin de la reconfiguration du connecteur une boîte de dialogue vous propose les choix suivants :

- Mise à jour du favori associé
- Suppression de l'association avec le favori avant la mise à jour du connecteur
- Annulation de la reconfiguration du connecteur et du favori

Editer les favoris

Un favori vous permet de conserver une configuration de connecteur et de la réutiliser dans d'autres scénarios.

Pour créer un favori associé à un connecteur

- 1 Démarrez l'éditeur de scénarios.
- 2 Ouvrez un scénario existant ou créez un nouveau scénario.
- 3 Sélectionnez le connecteur pour lequel vous voulez créer un favori.
- 4 Cliquez droit.
- 5 Choisissez **Favoris/ Créer un favori avec association** dans le menu contextuel.

Le favori apparaît dans le sous-onglet **Favoris** du volet **Boîte à outils**.

 **Note :**

Quand un connecteur est associé à un favori, le nom du favori apparaît au dessus du connecteur dans le volet du schéma du scénario.

Pour créer un favori non associé à un connecteur

- 1 Démarrez l'éditeur de scénarios.
- 2 Ouvrez un scénario existant ou créez un nouveau scénario.
- 3 Sélectionnez le connecteur pour lequel vous voulez créer un favori.
- 4 Cliquez-droit.
- 5 Choisissez **Favoris/ Créer un favori sans association** dans le menu contextuel.

Le favori apparaît dans l'onglet **Favoris** du volet **Boîte à outils**.

Pour utiliser un favori dans un scénario

- 1 Ouvrez ou créez le scénario dans lequel vous voulez utiliser un favori.
- 2 Placez le curseur de la souris dans l'onglet **Favori** du volet **Boîte à outils**.
- 3 Sélectionnez le favori.
- 4 Effectuez une des actions suivantes :
 - Glissez-déplacez ce favori dans le volet du schéma du scénario.
 - Double-cliquez.

Une instance de connecteur automatiquement associé à ce favori est créé dans le schéma du scénario.

Pour supprimer l'association d'un connecteur à un favori

- 1 Sélectionnez un connecteur associé à un favori dans le scénario.

Si un connecteur est associé à un favori, le nom de ce favori apparaît dans le schéma du scénario.
- 2 Cliquez droit.
- 3 Choisissez **Favoris/ Supprimer l'association avec le favori**.

Dès que l'association avec le favori est supprimé, le nom du favori qui apparaissait au dessus du connecteur dans le schéma du scénario disparaît.

Pour reconfigurer un connecteur à l'aide d'un favori et créer une association

- 1 Sélectionnez le favori.
- 2 Glissez-le sur un connecteur similaire au favori en maintenant votre doigt appuyé sur le bouton principal de la souris.
Exemple : un favori XML sur un connecteur XML.
- 3 Relâchez votre doigt.
- 4 Une boîte de dialogue s'affiche vous demandant de mettre à jour le connecteur avec la configuration du favori et de créer une association.
- 5 Cliquez sur **OK**.

Pour reconfigurer un connecteur à l'aide d'un favori sans créer d'association

- 1 Sélectionnez le favori.
- 2 Glissez-le sur un connecteur similaire au favori :
 - en maintenant votre doigt appuyé sur le bouton principal de la souris
 - en appuyant sur la touche **Maj.** (Shift) de votre clavier.
- 3 Relâchez votre doigt.
- 4 Une boîte de dialogue s'affiche vous demandant de mettre à jour le connecteur avec la configuration du favori sans créer d'association.
- 5 Cliquez sur **OK**.

Pour reconfigurer un favori à partir du sous-onglet Favoris

- 1 Démarrez l'éditeur de scénarios.
- 2 Sélectionnez l'onglet **Favoris** du volet **Boîte à outils**.
- 3 Sélectionnez le favori que vous voulez reconfigurer.
- 4 Effectuez une des actions suivantes :
 - Cliquez sur .
 - Cliquez-droit et sélectionnez **Reconfigurer le favori** dans le menu contextuel.
 - Choisissez **Favoris/ Reconfigurer le favori**.

- 5 Modifiez la configuration du favori à l'aide de l'assistant de configuration qui est lancé.

Au nouveau chargement des scénarios utilisant ce favori, les connecteurs associés sont automatiquement reconfigurés.

Pour reconfigurer un favori à partir d'un connecteur associé

- 1 Démarrez l'éditeur de scénarios.
- 2 Ouvrez un scénario utilisant un connecteur associé au favori que vous voulez reconfigurer.
- 3 Sélectionnez le connecteur associé.
- 4 Effectuez une des actions suivantes :
 - Cliquez-droit et choisissez **Configurer le connecteur** dans le menu contextuel.
 - Choisissez **Composants/ Configurer (F2)**.
- 5 Modifiez la configuration du favori à l'aide de l'assistant de configuration qui est lancé.
- 6 Cliquez sur **Oui** dans la fenêtre de dialogue qui apparaît.
Si vous cliquez sur **Non** dans cette boîte de dialogue, l'association avec le favori est supprimée avant la mise à jour du connecteur.
Au nouveau chargement des scénarios utilisant ce favori, les connecteurs associés sont automatiquement reconfigurés.

Pour supprimer un favori

- 1 Démarrez l'éditeur de scénarios.
- 2 Sélectionnez l'onglet **Favoris** du volet **Boîte à outils**.
- 3 Sélectionnez le favori que vous voulez supprimer.
- 4 Effectuez une des actions suivantes :
 - Cliquez sur .
 - Cliquez-droit et sélectionnez **Supprimer le favori** dans le menu contextuel.
 - Choisissez **Favoris/ Supprimer le favori**.
- 5 Attendez que le favori disparaisse de l'onglet **Favoris**.

 **Note :**

Quand un favori est supprimé, les connecteurs associés perdent cette association. Cette perte est symbolisée par la disparition du nom du favori au dessus du connecteur dans le schéma du scénario.

Gérer les fichiers de favoris

Les paramètres des favoris (connecteurs préconfigurés) sont contenus dans un fichier **Favorite.fav**. Ce fichier est enregistré dans le dossier bin32 de votre dossier d'installation de Connect-It. **Exemple : C:\Program Files\Peregrine\ConnectIt\bin32\Favorite.fav**

 **Note :**

La dernière version du fichier **favorite.fav** est enregistré dans un fichier **favorite.bak**. Si un problème survient lors d'une mise à jour, utilisez ce fichier en le renommant.

Ce fichier peut être exporté ou importé entre deux éditeurs de scénarios.

Pour importer un fichier de scénarios

- 1 Démarrez un éditeur de scénarios.
 - 2 Placez le curseur de la souris dans la zone de l'onglet **Favoris**.
 - 3 Effectuez une des actions suivantes :
 - Cliquez sur .
 - Cliquez droit et choisissez **Importer des favoris** dans le menu contextuel.
 - 4 Indiquez le chemin du fichier **Favorite.fav** contenant les favoris que voulez importer dans l'éditeur des scénarios.
-

 **Note :**

Vous pouvez effectuer cette opération manuellement dans un explorateur Windows en copiant et collant le fichier **Favorite.fav** dans le dossier **bin32** du dossier d'installation de Connect-It.

Pour exporter un fichier de scénarios

- 1 Démarrez un éditeur de scénarios.
- 2 Effectuez une des actions suivantes :
 - Placez le curseur de la souris dans la zone de l'onglet **Favoris**, cliquez droit et choisissez **Exporter des favoris** dans le menu contextuel.
 - Cliquez sur .
- 3 Indiquez le chemin du dossier dans lequel vous voulez enregistrer le fichier **Favorite.fav** contenant les favoris de l'éditeur de scénarios que vous utilisez.

Note :

Vous pouvez effectuer cette opération manuellement dans un explorateur Windows en copiant et collant le fichier **Favorite.fav** dans le dossier de votre choix.

Mise à jour entre favori et connecteurs associés

Quand vous ouvrez un scénario, la synchronisation entre un favori et ses connecteurs associés dépend de leur date de mise à jour.

Le favori ou le connecteur ayant la date de mise à jour la plus récente impose sa configuration aux autres.

Journaux

L'éditeur de scénario vous permet de consulter deux journaux :

- Le journal de Connect-It
- Le journal des documents

Journal de Connect-It

Le journal permet à l'utilisateur d'être informé par l'intermédiaire de messages de toutes les actions effectuées dans Connect-It.

Exemple : un message vous signale qu'un connecteur est ouvert.

Pour accéder au journal de Connect-It, sélectionnez l'onglet **Journal de Connect-It** de l'éditeur des scénarios.

Dans le journal, chaque action est symbolisée par l'icône . Le message d'une action peut être composé de plusieurs sous-messages détaillant cette action. Ces sous-messages peuvent être composés eux-mêmes d'autres sous-messages. Chaque message est accompagné de la date à laquelle l'action a été déclenchée. En cliquant droit dans le journal, un menu contextuel apparaît vous permettant de déplier un niveau ou replier et déplier tous les niveaux.

Figure 3.2. Journal de Connect-It



Pour un exemple de messages obtenus dans le journal de Connect-It, consultez le manuel **Connecteurs**, chapitre **Directives des connecteurs**, section **Directives de consommation \ Réconciliation \ Messages d'erreurs concernant le choix des clés de réconciliation**.

Typologie des icônes utilisées dans le journal de Connect-It

Ligne de suivi	Signification	Type de message	Exemple
	Information sur une action	Décrit une action dans Connect-It	L'ouverture d'un connecteur.
	Détails d'une action	Donne un détail sur une action.	Signale qu'un index particulier est utilisé dans un processus de réconciliation.
	Avertissement	Signale tout problème pouvant générer un mauvais traitement des données.	Un champ choisi comme clé de réconciliation autorise les doublons.

Ligne de suivi	Signification	Type de message	Exemple
	Problème majeur	Donne la raison de l'échec d'une action.	La configuration d'un connecteur n'est pas valide. La communication avec l'application externe est impossible.

Pour purger le journal de tous ces messages, sélectionnez l'option **Effacer les lignes de suivi affichées** dans le menu **Journal** ou cliquer sur .

Purger le journal de Connect-It a pour effet de purger également le journal des documents.

Journal des documents

Le journal des documents vous permet de contrôler et consulter le détail de vos documents après qu'ils ont été :

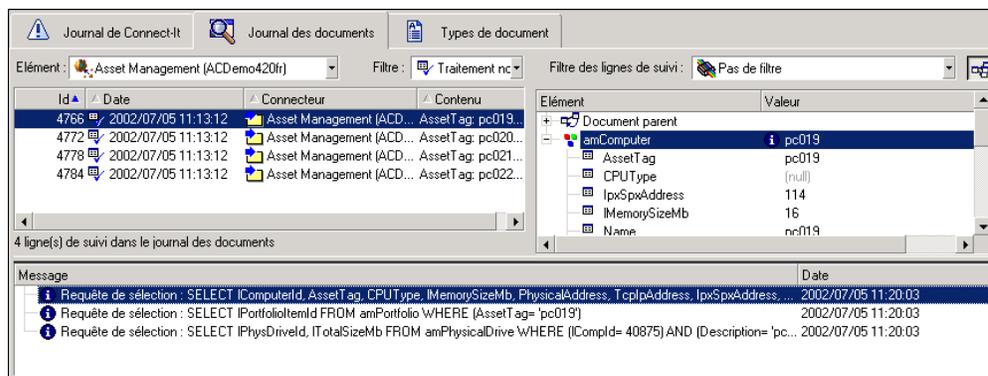
- produits ou consommés par un composant,
- acheminés par un lien.

Le journal des documents apparaît lorsque vous sélectionnez l'onglet **Journal des documents** dans l'éditeur de scénarios.

Cet onglet est composé de trois volets :

- Un volet où chaque ligne de suivi correspond à un document.
- Un volet de détail du document sélectionné
- Un volet de messages détaillant les lignes de suivi laissées par les éléments dans le document.

Figure 3.3. Editeur de scénarios - onglet du journal des documents



Configuration du journal des documents

Configurer le journal des documents consiste à :

- filtrer les lignes de suivi par type d'erreur de traitement du document
- fixer le nombre maximum de lignes de suivi affichées dans le journal des documents
- sauvegarder ou non les documents dont les lignes de suivi apparaissent dans le journal

Figure 3.4. Fenêtre de configuration du journal des documents



Pour configurer le journal des documents, sélectionnez **Configurer le journal des documents** du menu **Journal**.

Pour filtrer les lignes de suivi, sélectionnez une option du champ **Filtre**.

Pour fixer le nombre maximum de lignes de suivi affichées dans le journal des documents, renseignez le champ **Lignes de suivi maximum en mémoire** avec le nombre de lignes de votre choix.

 **Note :**

Le nombre de lignes de suivi affichées doit être adapté à la mémoire disponible de l'ordinateur sous peine de ralentir considérablement les traitements.

Pour sauvegarder vos documents, leur ligne de suivi et la ligne de suivi éventuellement laissée par un élément du scénario :

- Cochez la case **Utiliser des fichiers**
- Indiquez un nom générique dans le champ **Nom générique** du cadre **Enregistrer les lignes de suivi**(Exemple : acsc).

Le nom générique sert à créer trois fichiers qui contiennent :

- Les documents produits par votre scénario : fichiers **.dat** (Exemple : **acsc.dat**)
- Les messages présents dans le journal des documents (avertissement sur champ, échec de traitement...) : fichiers **.msg** (Exemple : **acsc.msg**)
- Un fichier d'indexation qui permet de retrouver rapidement les données contenues dans les fichiers **.dat** et **.msg** : fichiers **.idx** (Exemple : **acsc.idx**)

 **Note :**

Vous pouvez saisir dans le champ **Nom générique** des chemins complets ou relatifs pour vos fichiers. Par exemple :

-
-

Purger la mémoire et le stockage du disque

Deux commandes de purge sont disponibles dans le menu **Journal** :

- **Effacer les lignes de suivi affichées**

Cette commande permet de purger les volets des journaux de l'éditeur de scénarios : journal de Connect-It, journal des documents. En aucun cas, cette commande ne vide les fichiers contenant les lignes de suivi.

- **Supprimer les lignes de suivi stockées**

Si vous avez choisi d'enregistrer les lignes de suivi, cette commande permet de supprimer les trois fichiers (**.dat**, **.msg**, **.idx**) créés au moment du lancement du scénario.

Recharger le journal des documents

La commande **Recharger le journal des documents** du menu **Journal** permet d'afficher dans l'onglet du **Journal des documents** toutes les lignes de suivi contenues dans vos fichiers **.dat**, **.msg** et **.idx**. Cette commande est particulièrement utile si vous avez limité à un nombre restreint l'affichage de vos lignes de suivi dans le journal des documents. Dans ce cas, cette commande vous permet de voir l'ensemble des problèmes de traitement de vos documents.

Les lignes de suivi

Le journal des documents affiche des lignes de suivi. Chacune d'elles correspond à un document traité par un élément du scénario et comprend :

- un numéro d'identification
- une icône symbolisant la manière dont le document a été traité
- une heure et date de traitement
- le type du document traité
- le nom de l'élément qui a produit ou consommé le document
- le contenu du document

Le numéro d'identification correspond à l'ordre de traitement des documents. Le document portant le numéro d'identification **1** a été traité le premier. Cet identifiant unique permet de retrouver les documents quand ceux-ci sont gardés en mémoire.

Pour trier les colonnes du journal des documents par ordre croissant ou décroissant, cliquez sur le titre de chaque colonne.

Un triangle bleu orienté vers le haut signifie que le tri est croissant, orienté vers le bas que le tri est décroissant. Quand la colonne n'est pas triée, le triangle apparaît en grisé.

Quand l'affichage de l'élément d'une ligne est tronqué, il suffit de placer votre pointeur de souris sur cet élément pour que son contenu s'affiche dans son intégralité sous la forme d'une bulle d'aide.

Typologie des icônes utilisées dans le journal des documents

Chaque icône correspond à l'information la plus importante. Exemple : si pour un même document, certains champs reçoivent des avertissements et que d'autres champs sont rejetés, seul l'icône  apparaît dans le journal des documents. Le rejet intégral du document est considéré comme l'information la plus importante.

Dans le schéma du scénario, ces icônes peuvent apparaître en remplacement de l'ampoule symbolisant qu'un composant est ouvert ou fermé. Ces icônes signalent que le composant n'a pas traité normalement au moins un document (avertissement sur un champ, rejet d'un champ, rejet total du document).

Figure 3.5. Signal de problème de traitement



Quand une icône apparaît sur un composant, si vous cliquez droit sur ce composant et que vous choisissez **Afficher la ligne de suivi** dans le menu contextuel qui s'affiche, l'icône disparaît signifiant que l'utilisateur a pris connaissance du problème signalé.

Filtrage des lignes de suivi

Vous pouvez filtrer les lignes de suivi apparaissant dans le journal des documents en fonction :

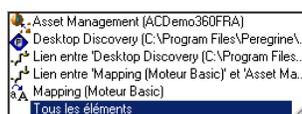
- de l'élément qui a traité le document,
- du traitement reçu par le document.

Si aucun élément n'est sélectionné dans le volet du schéma du scénario, l'ensemble des lignes de suivi disponibles en mémoire apparaît dans le journal des documents.

Pour ne faire apparaître que les lignes de suivi des documents traités par un élément donné :

- Sélectionnez cet élément dans le schéma du scénario ou
- Sélectionnez l'entrée de votre choix dans le champ **Élément**.

Figure 3.6. Filtre sur les lignes de suivi du journal des documents

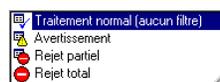


Pour faire apparaître les lignes de suivi en fonction de leur type de traitement, sélectionnez l'entrée de votre choix dans le champ **Filtre**.

La sélection d'un type de traitement fait également apparaître les types de traitement considérés plus importants.

Exemple : la sélection de l'entrée **Rejet Partiel** fait également apparaître les documents rejetés dans leur totalité.

Figure 3.7. Filtrage des lignes de suivi



Détail d'un document

Le détail d'un document vous permet de savoir quelles sont les valeurs prises par les champs de ce document ainsi que la façon dont il a été traité par un composant ou un lien.

Chaque fois qu'une ligne de suivi est sélectionnée dans le journal des documents, le détail du document concerné apparaît dans le volet situé à droite du journal des documents.

Le document est représenté sous forme arborescente.

Chaque noeud dans cette arborescence présente :

- le nom de l'élément représenté par ce noeud
- la valeur prise par ce noeud dans le document
- une ou plusieurs lignes de suivi éventuelles laissées par le composant ou le lien qui a traité le document

Toutes les lignes de suivi présentes dans le détail des documents sont accompagnées par des messages qui vous expliquent, par exemple, pourquoi le document a été rejeté dans son intégralité.

Différences entre le détail d'un document et le détail d'un type de document

Le détail d'un type de document apparaît au moment de l'édition des types de document consommés ou produits. Comme pour un document, un type de document est représenté sous forme arborescente. Il existe cependant de nombreuses différences.

Dans le détail d'un document, chaque noeud reçoit une valeur. Pour une structure ou une collection, cette valeur est une chaîne de description qui résume le contenu des champs contenus dans cette structure ou cette collection. Ainsi, par exemple, le noeud racine d'un document correspondant à un bien dans AssetCenter est décrit par son code interne qui apparaît en grisé. Pour une collection, le nombre de ses membres est indiqué, etc.

Dans le détail d'un type de document, seul le nom du noeud apparaît.

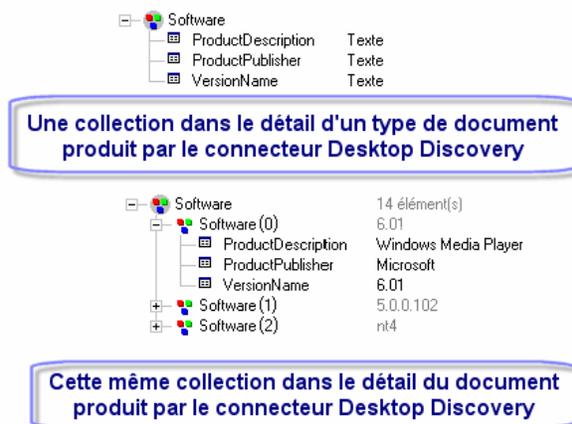
Figure 3.8. Une chaîne de description dans le document



Dans le détail d'un document, chaque membre d'une collection est représenté sous la forme d'une structure fille de cette collection. Chacun de ces noeuds porte le nom de la collection suivi d'un numéro entre crochets. Dans une collection à 3 membres, le premier membre de la collection porte le numéro 0, le second le numéro 1 et le troisième membre le numéro 2.

Dans le détail d'un type de document, seule la collection apparaît ainsi que la nature des valeurs attendues pour chaque champ (texte, nombre entier, date).

Figure 3.9. Une collection dans le détail d'un type de document et le détail d'un document



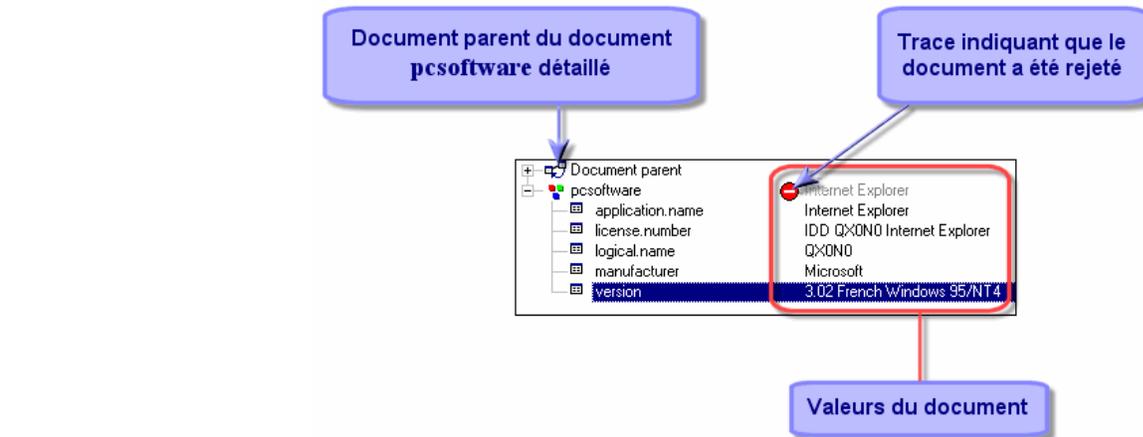
Document parent

Le détail d'un document peut contenir un document parent. Ce document parent contient les données qui ont permis la création du document. Ainsi, le détail d'un document produit par la boîte de mapping comprend également le détail du document source qui a permis sa production.

Exemple : Dans le scénario **iddac.scn**, le détail des documents produits par la boîte de mapping comprend le détail des documents produits par le connecteur Desktop Discovery. Ces documents sont du type de document source apparaissant dans le mapping.

En cas de rejet du champ d'un document par la boîte de mapping, le détail du document parent vous permet de contrôler la valeur de ce champ avant son traitement par la boîte de mapping.

Figure 3.10. Détail du document pcsoftware consommé par le connecteur ServiceCenter



Manipuler le document XML correspondant au détail d'un document

Le détail d'un document produit dans Connect-It est un document XML. Vous avez la possibilité de :

- copier ce document XML
- copier la DTD de ce document
- ouvrir ce document XML

Pour copier le document XML correspondant au détail d'un document

- 1 Positionnez votre pointeur dans la fenêtre de détail du document.
- 2 Cliquez droit.
- 3 Choisissez **Copier ce document XML** dans le menu contextuel qui s'affiche.

Pour copier la DTD du détail d'un document

- 1 Positionnez votre pointeur dans la fenêtre de détail du document.
- 2 Cliquez droit.
- 3 Choisissez **Copier cette DTD** dans le menu contextuel qui s'affiche.

Pour ouvrir le document XML correspondant au détail d'un document

- 1 Positionnez votre pointeur dans la fenêtre de détail du document.
- 2 Cliquez droit.
- 3 Choisissez **Ouvrir ce document XML** dans le menu contextuel qui s'affiche.

Note :

Le document XML est ouvert par l'application que vous avez associée aux fichiers **.xml** sur le poste sur lequel est installé Connect-It, par exemple : Internet Explorer, Netscape, etc.

Rechercher un élément dans le détail d'un document

Une fonction de recherche vous permet de retrouver les éléments dans le détail d'un document du journal des documents.

Pour rechercher un élément dans le détail d'un document

- 1 Sélectionnez un élément du détail d'un document.
- 2 Cliquez droit.
- 3 Choisissez **Rechercher** dans le menu contextuel.
- 4 Attendez qu'une boîte de dialogue s'affiche.
- 5 Saisissez le nom de l'élément que vous désirez chercher dans le champ **Rechercher** et sélectionnez les options de votre choix :
 - Respect de la casse
 - Recherche vers le haut ou le bas de la fenêtre.
- 6 Cliquez sur **Rechercher**.

Pour rechercher l'élément suivant ou précédent :

- 1 Effectuez une des actions suivantes :
 - Cliquez droit et choisissez **Rechercher suivant (F3)** dans le menu contextuel.
 - Cliquez droit et choisissez **Rechercher Précédent (Maj + F3)** dans le menu contextuel.

 **Note :**

La recherche se limite aux éléments affichés dans le détail du document. Pour que la recherche s'applique aux sous-noeuds d'un noeud donné, vous devez au préalable le déplier.

Typologie des lignes de suivi utilisées dans le détail d'un document

Une ligne de suivi correspond à une information de traitement laissée par un composant ou un lien sur l'un des champs du document.

Les lignes de suivi sont symbolisées par des icônes dans le volet de détail d'un document. Chaque ligne de suivi est commentée par un message apparaissant dans un autre volet.

Les lignes de suivi se séparent en deux catégories :

- Les lignes de suivi informatives (information essentielle sur le champ, information détaillée sur le champ)
- Les lignes de suivi relatives à des problèmes de traitement (avertissement sur un champ, échec de traitement)

Seule la ligne de suivi d'échec de traitement  signale le rejet total ou partiel du document.

Le tableau suivant représente les lignes de suivi, leur signification et le type de message les commentant.

Ligne de suivi	Signification	Type de message	Exemple
	Information détaillée	Informations techniques et exhaustives accompagnant le traitement d'une donnée.	Le connecteur IND télécharge des données qu'il convertit en documents. L'information détaillée indique la taille des fichiers en Ko, la vitesse de téléchargement...

Ligne de suivi	Signification	Type de message	Exemple
●	Information essentielle	Informations essentielles à l'utilisation des données.	Le nom du dossier et des fichiers créés par un connecteur après avoir consommé des documents.
▲	Avertissement sur un champ	Causes pour lesquelles un champ peut ne pas avoir été traité correctement.	Le connecteur ServiceCenter accepte un champ mais une partie de ce champ est tronquée pour respecter la structure de la base ServiceCenter.
●	Echec de traitement	Cause de l'échec du traitement.	Un champ est rejeté car sa valeur n'est pas conforme à la valeur attendue. Il peut entraîner le rejet total ou partiel du document.

Lignes de suivi grisées apparaissant à côté d'un noeud

Dans le volet de détail d'un document, les lignes de suivi peuvent apparaître en grisé à côté d'un noeud parent.

La présence de cette ligne de suivi grisée signifie qu'au moins un des noeuds fils présents sous ce noeud comporte au moins une ligne de suivi.

Quand plusieurs lignes de suivi sont présentes à l'intérieur d'un noeud, seule la ligne de suivi considérée comme la plus importante apparaît en grisé à coté du noeud parent.

Figure 3.11. Ligne de suivi grisée signalant une ligne de suivi sous le noeud



Filtrage du document par type de ligne de suivi

Filtrer le détail d'un document par type de ligne de suivi vous permet de ne voir apparaître que les noeuds portant la ligne de suivi de votre choix.

Si seuls les noeuds rejetés vous intéressent, sélectionnez l'entrée  dans le champ **Filtre des lignes de suivi** situé au dessus de votre volet de détail des documents.

Chaque fois que vous sélectionnez une entrée dans cette liste déroulante, vous obtenez l'affichage des noeuds portant les lignes de suivi situées sous cette entrée.

La hiérarchie établie dans cette liste correspond à l'importance de l'information : de l'information détaillée sur un champ à l'échec de traitement.

Figure 3.12. Filtrage des lignes de suivi dans le détail des documents



Messages d'explication des lignes de suivi

Chaque ligne de suivi présente dans le détail d'un document est commentée par un message.

Il suffit de sélectionner une ligne de suivi dans le volet de détail d'un document pour que le message commentant cette ligne de suivi apparaisse dans le volet des messages. Ce volet est situé sous le journal des documents.

Exemple : un message vous explique qu'un champ a été rejeté par le connecteur Asset Management car sa valeur viole l'intégrité de la base de données à laquelle il est connecté.

Afficher tous les messages des noeuds fils du noeud sélectionné

Pour faire apparaître l'ensemble des messages contenus dans les noeuds fils d'un noeud parent de votre document, cliquez sur .

Exemple

L'option  n'est pas sélectionnée, seul le message du noeud sélectionné apparaît.



Impossible de tester les erreurs d'insertion pour les événements traités de façon asynchrone

L'option  est sélectionnée, tous les messages des noeuds fils apparaissent.



Impossible de tester les erreurs d'insertion pour les événements traités de façon asynchrone
 Le champ 'Mois' de valeur '192.168.0.264' est trop long dans '192.168.0.264'. Il faut au maximum 2 caractère(s).
 Impossible de convertir '192.168.0.264' (type texte) en type date et heure (donnée non valide).
 Impossible de convertir '192.168.0.264,00C04F6ED7E6' (type texte) en type nombre entier 8 bits (donnée non valide).

Options

Des options dans l'éditeur de scénarios sont disponibles pour les catégories suivantes :

- **Accessibilité**
- **Affichage**
- **Confirmations**
- **Connecteur**
- **Documents**
- **Journaux**

Pour accéder à la fenêtre des options de l'éditeur de scénarios, choisissez **Options** dans le menu **Edition**.

Dans cette fenêtre, modifiez les options en double cliquant sur la cellule de votre choix dans la colonne **Valeur**. Dans la zone texte qui apparaît entrez la valeur de votre choix.

 **Note :**

Quand vous modifiez une des valeurs par défaut des options, l'intitulé de cette option s'affiche en rouge. Ceci vous permet de repérer rapidement les options dont vous avez changé la valeur.

Accessibilité

Cette catégorie regroupe des options de 'confort' d'utilisation de l'éditeur de scénario :

- Enlever les images de fond
- Eviter le clignotement
- Eviter les couleurs
- Mettre les polices en gras

Attention : toutes ces options ne sont pas utilisées dans l'éditeur de scénarios mais par d'autres applications Peregrine Systems.

Affichage

Cette catégorie regroupe les options d'affichage de l'éditeur de scénarios.

Afficher uniquement les connecteurs autorisés par le certificat d'autorisation

Cette option permet d'afficher dans la boîte d'outils de l'éditeur de scénarios les connecteurs que votre certificat d'autorisation vous donne le droit d'utiliser.

Changer la couleur des descriptions des assistants

Permet de changer la couleur des descriptions des assistants. Exemple : la couleur des descriptions des assistants de configuration des assistants.

Désactiver la touche 'Echap' dans les champs

Sous Windows, l'utilisation de la touche **Echap** vous permet de sortir d'une fenêtre d'édition. Exemple : la fenêtre d'édition des types de document produits par un connecteur source.

Par sécurité, dans l'éditeur de scénarios, la touche **Echap** ne s'applique qu'à l'édition de l'élément courant. Exemple : l'édition d'un mapping collection à collection dans la fenêtre d'édition d'un mapping.

Pour retrouver le comportement habituel de Windows, sélectionnez cette option.

Graphisme Windows

Cette option vous permet d'obtenir une présentation à plat ou standard de l'éditeur de scénarios.

Mode verbeux

Cette option permet le lancement de l'éditeur en mode verbeux. Si l'application a été lancée dans une invite de commandes avec l'option **-verbose**, la valeur de cette option n'est pas prise en compte.

Nombre de documents récupérés dans un bloc de la prévisualisation

Permet de fixer le nombre de documents afficher dans la fenêtre de prévisualisation des documents. Par défaut, le nombre de documents est 20.

Nombre de documents à produire lors d'un test de production des documents

Permet de fixer le nombre de documents à produire lors d'un test de production des documents. Dans ce mode de test, les connecteurs source du scénario produisent des documents mais les connecteurs destination n'interagissent pas avec les applications externes. Pour limiter ce nombre, il faut absolument cocher la case placée sur cette ligne.

Cette option est automatiquement prise en compte lorsque vous tester un type de document produit. En revanche, quand le scénario est exécuté en tant que service sous Windows, cette option est automatiquement désactivée.

Nombre maximum de lignes de suivi à afficher dans le journal de Connect-It

Cette option vous permet de limiter le nombre de lignes de suivi apparaissant dans le journal de Connect-It.

Nombre maximum de noeuds fils devant s'afficher simultanément

Cette option vous permet de fixer le nombre de noeuds fils apparaissant sous un noeud parent dans les types de document et les documents. Exemple : Vous préférez que seules les trois premières collections d'un type de document apparaissent sous le noeud racine.

Nombre maximum des noeuds fils d'un noeud devant être dépliés automatiquement

Cette option vous permet de fixer le nombre de noeuds fils apparaissant sous un noeud parent dans les types de documents et les documents. Cette option est utile quand certains noeuds de vos types de document contiennent trop de noeuds fils.

Nombre maximum de niveaux (récuratif) à mapper automatiquement

Cette option vous permet de définir le nombre de niveaux de noeuds (structure ou collection) qui pourra être automatiquement mappé lors d'un mapping table à table ou noeud à noeud.

Afficher les astuces au démarrage

Permet d'afficher une fenêtre d'astuce à chaque démarrage de Connect-It. Cette option peut-être désélectionnée directement dans la fenêtre des astuces (Case à cocher **Montrer au démarrage**).

Onglet

Vous pouvez modifier l'aspect des onglets apparaissant dans Connect-It en modifiant les options suivantes :

- **Afficher l'image**

Cette option permet d'afficher ou non l'icône accompagnant le nom d'un onglet. Exemple : l'icône  pour l'onglet **Journal de documents**.

- **Afficher l'info-bulle**

Cette option permet d'afficher ou non l'info-bulle qui apparaît chaque fois que le pointeur s'arrête sur le haut de l'onglet.

- **Afficher le texte**
Cette option permet ou non d'afficher le titre des onglets.
- **Style de présentation des onglets**
Permet de changer l'aspect des onglets dans la fenêtre des types de document produit.

Voir les bulles d'aide

Cette option permet de voir s'afficher les bulles d'aide qui apparaissent quand vous laissez votre pointeur sur un objet de Connect-It.

Confirmations

Cette catégorie représente les confirmations qui sont demandées à l'utilisateur après certaines opérations. Les confirmations s'opèrent par le biais d'une boîte de dialogue.

Afficher un avertissement chaque fois qu'un mapping collection à collection est nécessaire

Cette option permet d'afficher un avertissement chaque fois qu'un mapping collection à collection est nécessaire. Exemple : les champs d'une collection source sont mappés aux champs d'une collection destination sans que ces deux collections soient mappées l'une à l'autre. Dans ce cas, cet avertissement apparaît, rappelant à l'utilisateur qu'il doit obligatoirement mapper ces collections.

Afficher un message chaque fois que le mode Pause est activé

Cette option vous permet d'afficher un message chaque fois que le mode Pause est activé.

Afficher un message chaque fois que les paramètres de la JVM sont modifiés

Cette option vous permet d'afficher un message chaque fois que les paramètres de la JVM sont modifiés

Ne pas autoriser la validation d'un script comprenant des erreurs de compilation

Cette option ne vous autorise pas à valider un script comprenant des erreurs de syntaxe.

Ouvrir tous les connecteurs du scénario courant au démarrage d'un connecteur

Cette option permet d'ouvrir tous les connecteurs du scénario courant au démarrage d'un connecteur. Cette option est particulièrement utile car elle permet d'activer rapidement les mappings d'un scénario sans avoir à ouvrir les connecteurs les uns après les autres.

Utiliser une boîte de dialogue pour sélectionner le type de document ou le mapping à éditer

Cette option permet l'affichage d'une boîte de dialogue dans laquelle vous sélectionnez un type de document ou un mapping après avoir double-cliqué sur un connecteur ou une boîte de mapping ouverts.

Connecteurs

Des options sont disponibles pour les connecteurs suivants :

- Connecteur LDAP
- Connecteur InfraTools Network Discovery
- Connecteur ServiceCenter
- Connecteur Texte délimité et Connecteur XML

D'autres options sont disponibles pour l'ensemble des connecteurs :

- **Afficher les requêtes dans les lignes de suivi**
- **Base de données**
Afficher un message lorsque le type de donnée d'un champ n'est pas supporté.
- **Calculer le nombre de documents à traiter**
- **Compenser les dates par rapport au décalage avec le serveur**
- **Valider par défaut l'option d'utilisation d'un cache de l'assistant de configuration**

Connecteur LDAP

L'option **Montrer les attributs définis à la racine du DSE** permet notamment de déterminer si le serveur LDAP expose un attribut permettant de récupérer la date courante du serveur.

Connecteur InfraTools Network Discovery

L'option **Traiter uniquement les événements dont la date est inférieure à celle de démarrage** permet d'obliger votre scénario, une fois lancé, à traiter tous les événements survenus avant son lancement. Si ces événements sont en nombre important, votre scénario risque d'être bloqué. Par défaut, la valeur de cette option est **non**.

Connecteur ServiceCenter

L'option **Forcer le numéro de page de code de ServiceCenter** permet d'indiquer le numéro de page de code de votre choix. Par défaut, la valeur de cette option est 0. Cette valeur signifie que la page de code de ServiceCenter est celle de l'ordinateur sur lequel est installé Connect-It.

L'option **Contrôler le type de données du filtre** permet de contrôler si le type des valeurs des champs de ServiceCenter correspond bien au type spécifié par l'auteur de la requête. Exemple : le filtre `sysmodtime > 3` dans lequel un champ de type **date** est comparé à un entier n'est pas valide. Un message d'erreur s'affiche et le connecteur ne produit aucun document.

L'option **Informations de progression durant la récupération des types de document publiés par le connecteur** permet d'afficher dans le journal de Connect-It le nom des types de documents publiés par le connecteur que vous venez d'ouvrir.

Base de données

L'option **Afficher un message lorsque le type de donnée d'un champ n'est pas supporté** permet d'afficher un message dans le journal des documents quand un champ n'est pas supporté par un connecteur.

XML et Texte délimité

L'option **Afficher l'URL en cours de traitement dans le journal de Connect-It** permet d'afficher dans le journal de Connect-It l'URL en cours de traitement par le connecteur XML ou le connecteur Texte délimité.

Afficher les requêtes dans les lignes de suivi

Cette option permet de voir s'afficher le texte de vos requêtes dans le journal des documents (volet des messages). Cela est utile en phase de test de votre scénario (débogage).

Calculer le nombre de documents à traiter

Cette option permet d'afficher dans la barre d'état de l'éditeur le nombre de documents traités sur l'ensemble des documents à traiter. Exemple : si quatre documents ont été traités sur un ensemble de deux cents, la mention **4/200 documents traités** apparaît dans la barre d'état.

 **Avertissement :**

Cette option peut pénaliser les performances de votre machine si le nombre de documents à traiter est trop important.

Compenser les dates par rapport au décalage avec le serveur

Si cette option est activée, chaque fois que vous spécifiez un décalage avec le serveur au moment de la configuration du connecteur, les dates lues ou écrites sont compensées de la valeur de ce décalage.

Note : la compensation du décalage était automatique jusqu'à la version 2.7.1 de Connect-It.

Pour plus d'informations sur les options de décalages avec le serveur, consultez le manuel **Connecteurs**, chapitre **Configuration des connecteurs**, section **Déterminer le décalage avec le serveur**.

Valider par défaut l'option d'utilisation d'un cache de l'assistant de configuration. Non

Cette option permet d'utiliser les fichiers de cache pour les connecteurs sans avoir à sélectionner manuellement cette option dans l'assistant de configuration du connecteur.

Pour plus d'informations sur les options de cache, consultez le manuel **Connecteurs**, chapitre **Configuration des connecteurs**, section **Configurer le cache**.

Documents

Attention : les documents de cette catégorie ne sont pas les documents traités par les connecteurs mais les fichiers SCN contenant vos scénarios.

Cette catégorie regroupe les options concernant le chargement de vos fichiers SCN :

- **Nombre maximum de documents récemment ouverts à mémoriser dans le menu Fichier**

Cette option concerne la liste des fichiers SCN apparaissant dans la liste du menu **Fichier**.

- **Au démarrage, charger automatiquement le dernier document utilisé**
- **Demander confirmation avant de sauvegarder**

Cette option permet d'afficher une boîte de dialogue avant toute sauvegarde effectuée dans Connect-It (sauvegarde d'un mapping, d'un scénario, d'un programmeur, etc.)

Journaux

Pour les journaux, l'option **Taille des fichiers .log (Mo)** permet d'indiquer la taille de vos fichiers **.log**. Chaque fois que cette taille est atteinte, les nouvelles données écrasent les données les moins récentes.

4 | Mise en place d'un scénario d'intégration

CHAPITRE

La mise en place d'un scénario d'intégration consiste à :

- Choisir des composants
- Lier ces composants

Mettre en place un scénario avec un assistant

L'assistant **Mise en place d'un scénario** vous permet :

- de choisir le connecteur source et le connecteur destination de votre scénario
- d'effectuer la configuration de ces deux connecteurs

Une fois achevée la configuration de votre connecteur destination, les deux connecteurs sont reliés à une boîte de mapping. Les connecteurs et la boîte de mapping apparaissent dans le volet Schéma du scénario de l'éditeur de scénarios.

Pour lancer l'assistant Mise en place d'un scénario

- 1 Effectuez l'une des actions suivantes :
 - Cliquez sur  dans la barre d'outils de l'éditeur de scénarios
 - Choisissez **Nouveau** dans le menu **Fichier**

Mettre en place manuellement un scénario

Pour mettre en place un scénario manuellement, vous devez :

- 1 placer les composants de votre scénario dans le schéma du scénario
- 2 lier ces composants

Placer des composants dans le schéma du scénario

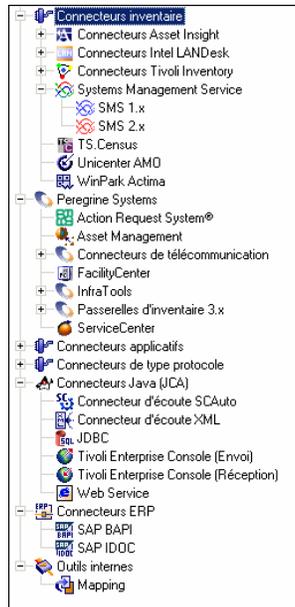
Le fait de placer un connecteur dans le schéma du scénario lance automatiquement l'assistant **Configuration du connecteur**.

Pour placer un composant dans le schéma du scénario

- 1 Glissez-le de la boîte à outils vers le schéma du scénario ou sélectionnez-le et double-cliquez.
- 2 Configurez-le en saisissant des valeurs dans les pages de l'assistant **Configuration du connecteur** qui s'est affiché.

Pour une présentation de la configuration des connecteurs, consultez le manuel **Connecteurs**, chapitre **Configuration des connecteurs**.

Figure 4.1. Boîte à outils de l'éditeur de scénarios



Pour reconfigurer un composant déjà placé

- 1 Sélectionnez-le dans le schéma du scénario
- 2 Effectuez l'une des actions suivantes :
 - Choisissez **Composants/ Configurer**
 - Cliquez droit et choisissez **Configurer le connecteur** dans le menu contextuel qui s'affiche.

Lier des composants

La manière dont vous liez les connecteurs permet de définir lequel d'entre eux est un connecteur source ou un connecteur destination :

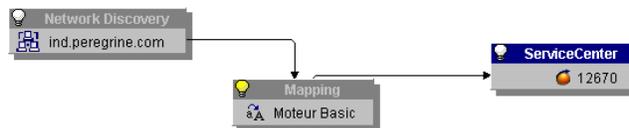
- Un lien sortant indique que le connecteur est un connecteur source
- Un lien entrant indique que le connecteur est un connecteur destination

Note :

Quand vous n'utilisez pas les types de document pivots, un connecteur source, à de rares exceptions, n'est jamais relié directement à un connecteur destination. Il est relié à une boîte de mapping qui est, elle-même, reliée au connecteur destination.

Dans l'exemple ci-dessous, le connecteur Network Discovery est le connecteur source et le connecteur ServiceCenter, le connecteur destination. Quand le connecteur Network Discovery produit des documents, ceux-ci sont consommés par le connecteur ServiceCenter après avoir été transformés par la boîte de mapping.

Figure 4.2. Liaison des composants d'un scénario



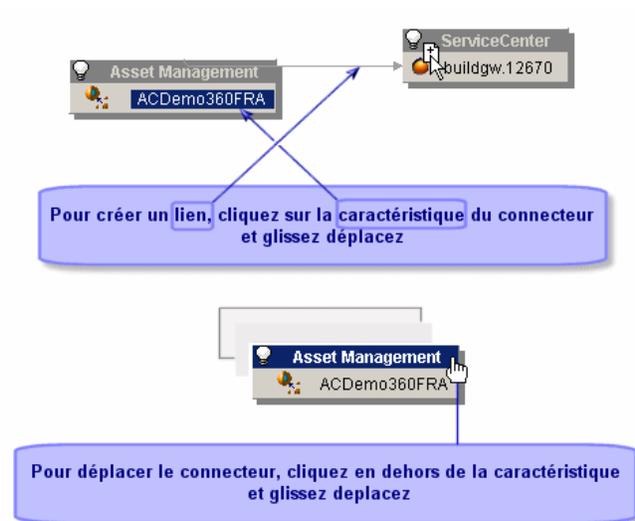
Pour lier un composant à un autre composant

- 1 Cliquez sur la partie inférieure du composant.
- 2 Attendez que la caractéristique du composant s'affiche en blanc.
- 3 Maintenez le doigt appuyé sur le bouton gauche de la souris.
- 4 Glissez le lien qui apparaît jusqu'à un autre composant.

Note :

Si vous cliquez en dehors de la caractéristique du composant, vous déplacez ce composant dans le schéma de votre scénario.

Figure 4.3. Créer un lien ou déplacer un connecteur



Pour lier deux connecteurs sans passer par une boîte de mapping

Quand vous liez deux connecteurs, Connect-It crée automatiquement une boîte de mapping entre ces deux connecteurs. Si vous ne voulez pas que cette boîte de mapping soit créée entre deux connecteurs, liez-les en maintenant la touche **Shift**.

Le lien direct entre deux connecteurs s'applique à certains connecteurs (Exemple : connecteurs Texte et XML). Il permet d'exporter directement les données du connecteur source en utilisant la structure du document produit par le connecteur source.

Supprimer un composant

Pour supprimer un connecteur, un lien ou une boîte de mapping dans le schéma du scénario :

- 1 Sélectionnez ce composant
- 2 Effectuez une des actions suivantes :
 - Choisissez **Edition/ Effacer**.

- Appuyez sur la touche **Suppr.**
- Cliquez droit et choisissez **Effacer** dans le menu contextuel.

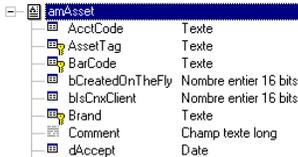
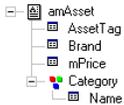
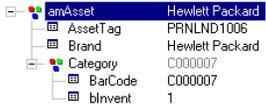
Editer les types de document produits ou consommés

Un type de document produit ou consommé est le modèle sur lequel s'appuie un connecteur pour produire ou consommer des documents durant le déroulement d'un scénario. Un type de document produit ou consommé contient une sélection des champs d'un type de document publié par le connecteur.

Exemple :

Le type de document **amAsset** consommé par le connecteur Asset Management du scénario **idd\iddac41\iddac.scn** ne contient qu'un petit nombre des champs que l'on retrouve dans le type de document **amAsset** publié par ce même connecteur Asset Management.

Le tableau suivant retrace la généalogie de tout document traité dans Connect-It.

Type de document publié	Type de document produit ou consommé	Document produit
 <pre> amAsset ├── AcciCode Texte ├── AssetTag Texte ├── BarCode Texte ├── bCreatedOnTheFly Nombre entier 16 bits ├── bisCnxClient Nombre entier 16 bits ├── Brand Texte ├── Comment Champ texte long └── dAccept Date </pre>	 <pre> amAsset ├── AssetTag ├── Brand ├── mPrice ├── Category └── Name </pre>	 <pre> amAsset Hewlett Packard ├── AssetTag PRNLND1006 ├── Brand Hewlett Packard ├── Category C000007 ├── BarCode C000007 └── blnInvent 1 </pre>

Configuration

Le connecteur publie des types de document. Un type de document publié est sélectionné.

Création d'un type de document

Créer un type de document produit ou consommé revient à sélectionner certains éléments d'un type de document publié.

Lancement du scénario

Le connecteur produit ou consomme un document où les éléments du type de document produit ou consommé prennent des valeurs.

Créer un type de document produit ou consommé par un connecteur revient à :

- choisir un type de document parmi ceux publiés par le connecteur
- sélectionner des champs parmi ceux du type de document

Exemple :

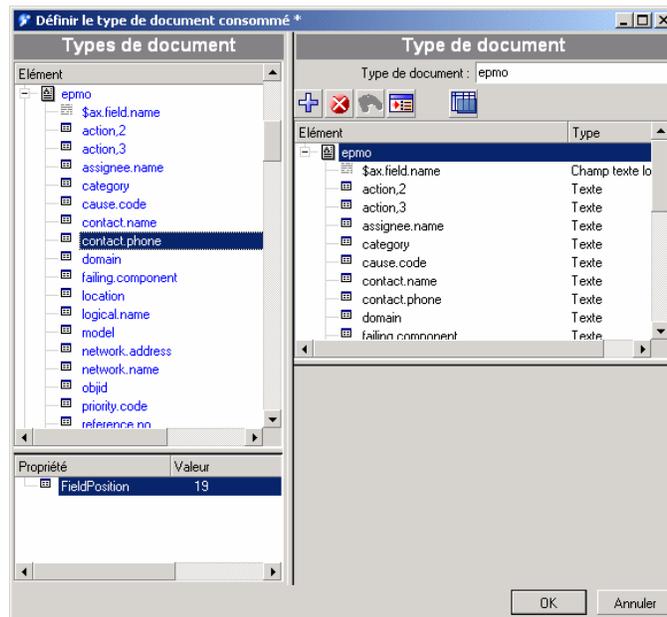
Vous choisissez le type de document **amProduct** parmi les types de document publiés par le connecteur Asset Management. Ensuite, vous sélectionnez les champs qui vous intéressent : le code-barres du produit, sa marque, son prix, sa référence catalogue, etc.

Accéder à la fenêtre d'édition des types de document produits ou consommés

La fenêtre d'édition des types de document produits ou consommés se sépare en deux zones :

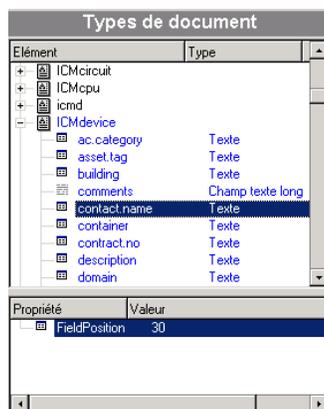
- Un plan de travail à droite sur lequel vous créez le type de document produit ou consommé.
 Sous ce plan de travail apparaissent les onglets de directives. Pour certains connecteurs aucune directive n'est disponible.
- Une zone à gauche où apparaissent les types de document publiés par le connecteur. Cette zone se scinde en deux volets :
 - Un volet où les noeuds racine (📁) des types de document publiés apparaissent. Dans ce volet, vous déployez les noeuds (noeud racine, structures, collections) de vos types de document pour en voir la composition. Chaque noeud et chaque champ (noeud terminal) est décrit par son nom et son type (texte, nombre entier, date...). Quand vous sélectionnez un noeud dans un type de document, ce noeud devient la sélection courante.
 - Un volet où figurent des propriétés de la sélection courante ainsi que la valeur de ces propriétés. Ces propriétés sont des indications techniques qui aident les utilisateurs avancés à comprendre la signification de la sélection courante ainsi que ses propriétés.

Figure 4.4. Fenêtre d'édition des type de documents consommés par le connecteur ServiceCenter



Dans la capture d'écran suivante, le champ **contact.name** du type de document ICMdevice est de type **texte**.

Figure 4.5. Détail de la zone de destination



Pour accéder à la fenêtre d'édition des types de document produits ou consommés

- 1 Sélectionnez un connecteur dans le volet du schéma du scénario.
- 2 Sélectionnez l'onglet **Types de document** ou double-cliquez sur le connecteur.
- 3 Choisissez le volet des types de document produits ou celui des types de document consommés.
- 4 Cliquez sur  pour créer un nouveau type de document ou sur  pour éditer le type de document existant que vous avez sélectionné.

Créer un type de document produit ou consommé

Créer un nouveau type de document produit ou consommé consiste à déplacer le noeud racine ou un ou plusieurs éléments (collection, structure, champ) sur le plan de travail de la fenêtre d'édition des types de document.

Pour créer un type de document produit ou consommé

Pour créer un type de document produit ou consommé :

- 1 Placer le noeud racine, un ou plusieurs éléments d'un type de document sur le plan de travail en effectuant une des actions suivantes :
 - Glissez-déplacez-le sur le plan de travail.

- Sélectionnez-le puis cliquez sur .
- Double-cliquez directement cet élément ou ce noeud racine

 **Note :**

Si vous prévoyez d'utiliser un grand nombre des champs d'un noeud non terminal (noeud racine, structure, collection), posez-le directement sur le plan de travail. Automatiquement tous les champs de ce noeud sont également posés dans le plan de travail. Vous n'avez plus qu'à effacer les éléments indésirables.

Pour effacer un élément du plan de travail

- 1 Sélectionnez-le et cliquez sur  ou appuyez sur la touche **Supprimer** ou
- 2 Glissez-déplacez-le à nouveau vers la fenêtre des types de document publiés par le connecteur

Une fois votre sélection faite, nommez votre type de document (par défaut, il porte le nom du type de document publié dont le noeud racine est posé sur le plan de travail) et cliquez sur **OK**.

 **Note :**

Lorsqu'on réalise un mapping dans la boîte de mapping, tous les éléments du type de document publié par le connecteur source et utilisés dans le mapping sont automatiquement ajoutés au type de document produit. Il est souvent utile d'éditer le type de document pour :

- y supprimer des éléments qui ne sont plus utilisés dans le mapping,
 - ajouter des éléments qui ne sont pas utilisés par le mapping, mais qui sont indispensables. A l'exécution d'un scénario, il s'agira notamment d'éléments dont les valeurs doivent être récupérées dans la source pour mettre à jour le pointeur de programmation du connecteur InfraTools Network Discovery.
-

Copier la DTD d'un type de document produit ou consommé

Chaque type de document dans Connect-It correspond à un document de type XML. Connect-It vous permet de copier la DTD liée à ce document.

Pour copier la DTD d'un type de document produit ou consommé

- 1 Créez ou éditez un type de document produit.
- 2 Positionnez votre pointeur sur le plan de travail.
- 3 Cliquez sur le bouton droit de votre souris.
- 4 Choisissez **Copier cette DTD XML** dans le menu contextuel.

Chaque fois que vous vous trouvez dans la fenêtre d'édition d'un type de document, un menu contextuel obtenu par clic droit vous permet de copier la DTD dans votre presse-papiers.

Voir les données d'un type de document produit

Connect-It vous permet de voir les données correspondant aux éléments d'un type de document produit. Ces données sont celles enregistrées dans l'application externe à laquelle est connecté votre connecteur.

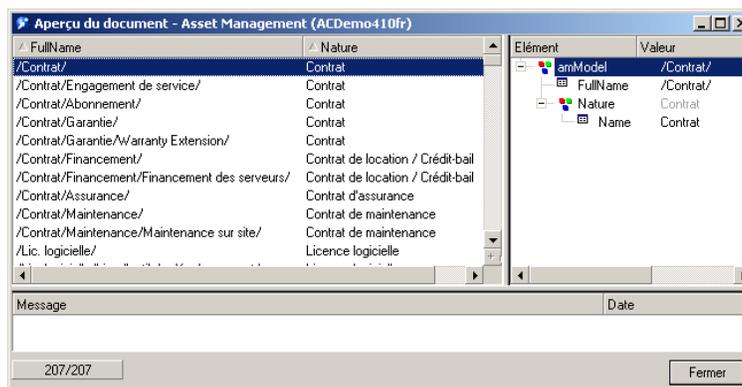
Note :

Si la visualisation des données dépasse un certain temps, une fenêtre s'ouvre automatiquement et vous propose d'interrompre l'action en cours. Si vous choisissez d'interrompre cette visualisation, vous avez accès aux dernières données recherchées. Vous avez la possibilité de reprendre l'action interrompue en cliquant sur .

Exemple :

Les données d'un type de document produit par un connecteur de type base de données correspondent aux enregistrements d'une des tables de cette base.

Figure 4.6. Fenêtre de visualisation des données d'un type de document produit



Le nombre de documents qui sont chargés dans la fenêtre de visualisation est fixé par l'option **Nombre de documents récupérés dans un bloc lors de la prévisualisation** (section **Affichage**) disponible dans le menu **Edition/Options**. Par défaut, les vingt premiers documents sont récupérés. Appuyez sur le bouton  pour recharger une nouvelle série de documents. Lorsque tous les documents sont visualisés, le bouton  est grisé et désactivé.

 **Note :**

Le bouton  n'est pas disponible pour tous les connecteurs.

Pour voir les données d'un type de document produit

- 1 Créez ou éditez un type de document produit.
- 2 Effectuez une des actions suivantes :
 - Cliquez sur .
 - Positionnez votre pointeur sur le plan de travail, cliquez sur le bouton droit de votre souris et choisissez **Voir les données** dans le menu contextuel.

Une fenêtre de visualisation des données d'un type de document se décompose en deux volets :

- Un volet gauche dans lequel apparaissent tous les enregistrements de l'application externe correspondant aux éléments du type de document

Chacune des colonnes de ce volet correspond à un des éléments du type de document produit

- Un volet droit représentant le type de document produit accompagné des valeurs de l'enregistrement sélectionné dans le volet gauche

Pour interrompre la collecte des données

- 1 Effectuez une des actions suivantes :
 - Attendez qu'une boîte de dialogue s'affiche vous demandant d'interrompre la requête courante et cliquez sur OK.
 - Appuyez sur la touche d'échappement (**Echap**).

Rechercher un élément dans le type de document

Une fonction de recherche vous permet de retrouver les éléments dans un type de document sans avoir à déplier l'ensemble des noeuds de ce type de document.

Pour rechercher un élément dans un type de document

- 1 Sélectionnez un élément du type de document dans le plan de travail ou la zone des types de document publiés.
- 2 Effectuez une des actions suivantes :
 - Cliquez droit et choisissez Rechercher (Ctrl + F) dans le menu contextuel.
 - Cliquez sur .
- 3 Saisissez le nom de l'élément que vous désirez chercher dans le champ **Rechercher** et sélectionnez les options de votre choix :
 - Respect de la casse
 - Recherche vers le haut ou le bas de la fenêtre.
- 4 Cliquez sur **Rechercher**.

Pour rechercher l'élément suivant ou précédent :

- 1 Effectuez une des actions suivantes :
 - Cliquez droit et choisissez **Rechercher suivant (F3)** dans le menu contextuel.
 - Cliquez droit et choisissez **Rechercher Précédent (Ctrl + F3)** dans le menu contextuel.

 **Note :**

La recherche se limite aux éléments affichés dans les fenêtres des types de document publiés. Pour que la recherche s'applique aux sous-noeuds d'un noeud donné, vous devez au préalable le déplier.

Menus contextuels de la fenêtre d'édition des types de document

Dans la fenêtre d'édition des types de document, deux menus contextuels sont disponibles :

- Pointeur placé dans la zone des types de document publiés
- Pointeur placé sur le plan de travail

Pointeur placé dans la zone des types de document publiés

Commande	Fonction
Rechercher	Permet d'afficher la boîte de dialogue de recherche d'une chaîne de caractères.
Rechercher suivant	Permet de relancer la recherche de la dernière chaîne de caractères indiquée vers le bas de la fenêtre.
Rechercher précédent	Permet de relancer la recherche de la dernière chaîne de caractères indiquée vers le haut de la fenêtre.
Copier cette DTD XML	Copie la DTD XML du document dans le presse-papiers de votre système d'exploitation.
Copier le chemin	Copie dans le presse-papiers de votre système d'exploitation le chemin de l'élément sélectionné.
Ajouter cet élément	Ajoute l'élément sélectionné au type de document produit ou consommé que vous créez.
Filtrer les types de document inutilisés	N'affiche que le type de document produit ou consommé édité dans le plan de travail.

Pointeur placé sur le plan de travail

Commande	Fonction
Rechercher	Permet d'afficher la boîte de dialogue de recherche d'une chaîne de caractères.
Rechercher suivant	Permet de relancer la recherche de la dernière chaîne de caractères indiquée vers le bas de la fenêtre.
Rechercher précédent	Permet de relancer la recherche de la dernière chaîne de caractères indiquée vers le haut de la fenêtre.
Copier cette DTD XML	Copie la DTD XML du document dans le presse-papiers de votre système d'exploitation
Copier le chemin	Copie dans le presse-papiers de votre système d'exploitation le chemin de l'élément sélectionné
Enlever cet élément	Enlève l'élément sélectionné du type de document produit ou consommé que vous créez
Voir les données	Permet de voir les données correspondant aux éléments d'un type de document produit
Filtrer les types de document inutilisés	N'affiche que le type de document produit ou consommé édité dans le plan de travail.

Enregistrer un scénario

Pour enregistrer un scénario

1 Effectuez une des actions suivantes :

- Choisissez **Fichier/ Enregistrer**.
- Cliquez sur .

Pour enregistrer un scénario sous un nom différent, choisissez :

1 Choisissez **Fichier/ Enregistrer sous**.

Copie de sauvegarde d'un scénario

Lorsque vous enregistrez un scénario, une copie de sauvegarde est automatiquement créée dans le dossier du scénario. Cette copie porte l'extension **.bak**.

Pour désactiver cette option :

- 1 Choisissez **Edition/ Options**.
- 2 Dépliez le noeud **Confirmation**.
- 3 Attribuez **Non** à l'option **Conserver la copie de sauvegarde du scénario**.
- 4 Cliquez sur **OK**.

Déplacer un scénario

Lorsque vous déplacez un scénario vers un autre dossier, n'oubliez pas

- Soit de déplacer les fichiers associés du scénario.
 - Soit de changer les chemins relatifs contenus dans les scripts du scénario.
- Vérifiez notamment les chemins des fichiers **.bas**, **.str** et **.mpt**.

Ouvrir un scénario existant

L'éditeur de scénarios vous permet d'ouvrir les fichiers portant l'extension **.scn**.

Pour ouvrir un scénario existant

- 1 Effectuez une des actions suivantes :
 - Lancez l'éditeur de scénarios et choisissez **Fichier/ Ouvrir**.
 - Sélectionnez le fichier **.scn** dans un explorateur et double-cliquez sur ce fichier.
L'éditeur de scénarios se lance et ouvre le scénario.
 - Sélectionnez le fichier **.scn** dans un explorateur et glissez-le dans la fenêtre principale de l'éditeur de scénarios.

 **Note :**

Si un scénario est ouvert, une boîte de dialogue s'affiche demandant si vous voulez enregistrer ce scénario avant d'ouvrir un nouveau scénario.

5 | Mappings des types de document

CHAPITRE

Mapper des types de document permet à un connecteur destination de consommer les documents produits par un connecteur source. Dans l'éditeur de scénarios, une boîte de mapping placée entre deux connecteurs permet d'éditer des mappings entre un type de document produit source et un type de document consommé destination.

Créer un mapping consiste à associer les éléments des types de document source à ceux d'un type de document destination.

Pour simplifier la lecture de ce chapitre, nous utilisons les notions d'**élément source** et d'**élément destination** ainsi que celles de champ, structure et collection source et destination.

Quand aucun élément du type de document source ne peut être associé directement à un élément du type de document destination, vous devez rédiger des scripts de mappings.

Pour une présentation des scripts de mapping, consultez le chapitre [Scripts de mapping](#) [page 121].

Boîte de mapping

La boîte de mapping vous permet d'éditer des mappings entre des types de document produits par un connecteur source et les types de document consommés par un connecteur destination.

Configuration de la boîte de mapping

Configurer une boîte de mapping dans Connect-It se fait par l'intermédiaire d'un assistant.

Pour lancer l'assistant de configuration d'une boîte de mapping, les méthodes suivantes sont disponibles :

- **La boîte de mapping n'est pas présente dans le schéma du scénario**
 - 1 Double-cliquez sur la boîte de mapping dans la boîte à outils.
 - 2 Glissez déplacez la boîte de mapping de la boîte à outils dans le schéma du scénario.
- **La boîte de mapping est déjà présente dans le schéma du scénario**
 - 1 Sélectionnez la boîte de mapping et choisissez **Composants/Configurer**.
 - 2 Sélectionnez la boîte de mapping et appuyez sur **F2**.
 - 3 Sélectionnez la boîte de mapping, cliquez droit et choisissez **Configurer** dans le menu contextuel.
 - 4 Cliquez sur .

Deux pages sont disponibles dans l'assistant de configuration de la boîte de mapping. Elles sont documentées dans les sections suivantes.

 **Note :**

La seconde page n'apparaît pas si vous n'avez pas cliqué sur , dans la barre d'outils de l'éditeur de scénarios.

Nommer et décrire la boîte de mapping

La première page de l'assistant de configuration vous permet de spécifier :

- Le nom de la boîte de mapping
- La description de la boîte de mapping

Nom

Nommez votre boîte de mapping. Par défaut, la valeur de champ est **Mapping**. Ce nom doit être unique au sein d'un même scénario.

Description

Rédigez un texte qui décrit votre connecteur.

Configuration avancée

Cette page vous permet d'indiquer le comportement de l'opérateur [..] dans les scripts de mapping. Pour les versions postérieures à la version 3.2 de Connect-It, l'opérateur {..} peut être utilisé à la place de l'opérateur [..].

Exemple d'utilisation des opérateurs [..] et {..}

Les scripts du tableau suivant testent l'existence et la valeur d'un champ **FamilyName**. Si ce champ a une valeur, la valeur de retour est **Doe** dans tous les cas.

En fonction de la sélection ou de la non sélection de l'option **Vérifier l'existence de l'élément sur lequel s'applique l'opérateur [..]** et du type d'opérateur utilisé dans le script, la valeur envoyée dans le document consommé est différente.

IMPORTANT :

La notion de valeur vide diffère en fonction du type du champ concerné :

- Pour un champ de type texte, la valeur vide est égale à "".
- Pour un champ de type numérique, la valeur vide est égale à 0.
- Etc.

Option désélectionnée (ancien comportement)

	FamilyName = ""	FamilyName n'existe pas	FamilyName = " valeur "
RetVal = "Doe" If [FamilyName] = "" Then RetVal = [FamilyName] End If	Valeur vide	Valeur vide	Doe

RetVal = "Doe" If {FamilyName} = "" Then RetVal = [Fami lyName] End If	Valeur vide	Valeur vide	Doe
---	-------------	-------------	------------

RetVal = "Doe" If {FamilyName} = "" Then RetVal = {Fami lyName} End If	Valeur vide	Valeur vide	Doe
---	-------------	-------------	------------

Option sélectionnée (nouveau comportement)

RetVal = "Doe" If [FamilyName] = "" Then RetVal = [Fami lyName] End If	Valeur vide	Le noeud concerné par le script est rejeté par l'éditeur de scénarios. Le message "" apparaît dans le journal des documents.	Doe
---	-------------	--	------------

RetVal = "Doe" If {FamilyName} = "" Then RetVal = [Fami lyName] End If	Valeur vide	Le noeud concerné par le script est rejeté par l'éditeur de scénarios. Le message "" apparaît dans le journal des documents.	Doe
---	-------------	--	------------

RetVal = "Doe" If {FamilyName} = "" Then RetVal = {Fami lyName} End If	Valeur vide	Valeur NULL	Doe
---	-------------	-------------	------------

Editer des mappings

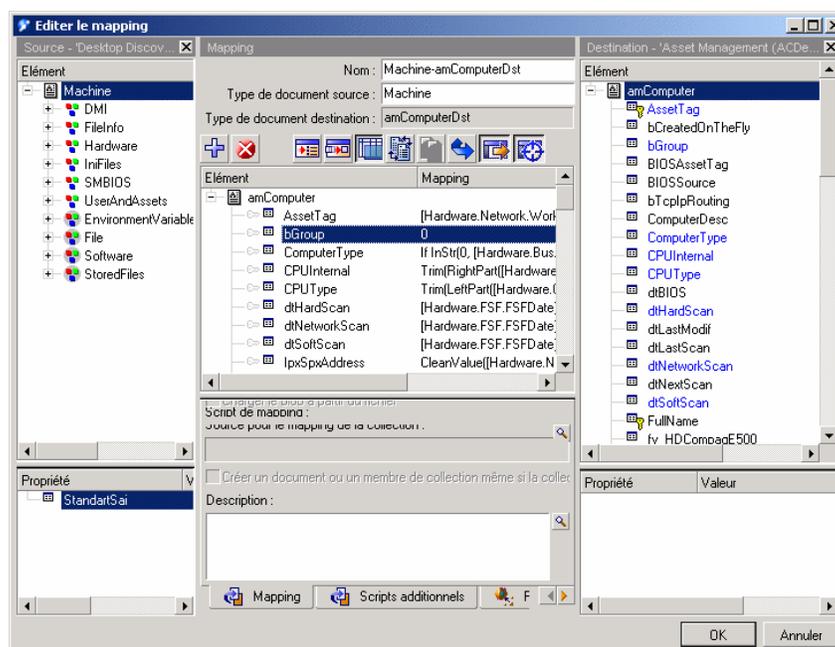
Cette section vous présente les différentes procédures vous permettant d'éditer les mappings de vos scénarios d'intégration.

Fenêtre d'édition des mappings

Une fenêtre d'édition d'un mapping se sépare en trois zones distinctes :

- Une zone faisant apparaître les types de document publiés par le composant source sélectionné
Si vous avez choisi dans la boîte **Sélectionner la source et la destination** un type de document produit existant, le noeud racine de ce type de document apparaît déplié dans cette zone.
- Une zone faisant apparaître les types de document publiés par le composant destination sélectionné.
- Une zone de mapping composée :
 - d'un plan de travail où l'on mappe des éléments source à des éléments destination
 - d'un volet faisant apparaître les onglets de directives du composant destination et un onglet permettant de décrire chaque noeud du mapping.

Figure 5.1. Fenêtre d'édition d'un mapping

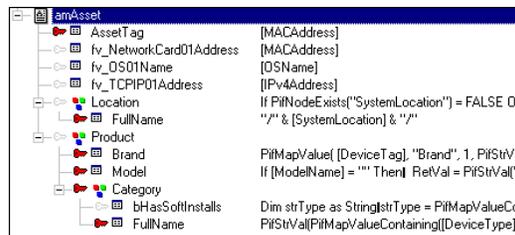


La barre d'outils de la boîte de mapping vous sert à effectuer les actions les plus courantes du mapping.

Icône	Fonction
	Poser un élément destination sur le plan de travail.
	Supprime la sélection courante du plan de travail.
	Associe un élément destination à un élément source.
	Filtre les types de document source et destination non utilisés dans le mapping.
	Duplique un élément destination dans le plan de travail.
	Trouve l'élément destination mappé.
	Masque ou affiche les types de document publiés par le connecteur source.
	Masque ou affiche les types de document publiés par le connecteur destination.
	Permet d'afficher une boîte de recherche dans la zone où un élément est sélectionné.
	Permet d'afficher une boîte de recherche et remplacement dans le plan de travail du mapping.

Le plan de travail de la fenêtre de mapping présente le détail du mapping entre éléments destination et éléments source. Quand un script est utilisé, il apparaît sous forme résumée à côté de l'élément destination auquel il est associé.

Figure 5.2. Résumé des scripts de mapping sur le plan de travail



La fenêtre d'édition d'un mapping, en exposant tous les types de document publiés par un connecteur source et un connecteur destination, vous permet de créer simultanément le type de document produit par le connecteur source et le type de document consommé par le connecteur destination. Le fait

d'associer pour la première fois un élément source à un élément destination impose le type de document produit et le type de document consommé par vos composants source et destination.

 **Note :**

Dans un même mapping, il est impossible d'associer les éléments d'un type de document source à ceux de plusieurs types de document destination et inversement. Si au cours de l'édition de votre mapping, vous changez l'un des types de document, vous commencez un nouveau mapping et vous perdez toutes vos associations précédentes. Pour associer un élément source à ceux de plusieurs types de document destination, effectuez plusieurs mappings s'appuyant sur le même type de document source.

En revanche, dans des mappings différents, utiliser le même document source permet de minimiser le nombre de requêtes à effectuer sur le connecteur source.

Vous pouvez également sélectionner un type de document destination et glisser-déplacer l'un de ses noeuds sur votre plan de travail sans l'associer à un type de document source : les champs situés sous ce noeud apparaissent sur le plan de travail.

 **Note :**

Vous pouvez visualiser les données de la source, et parfois celles de la destination, en utilisant le menu contextuel **Voir les données** sur un élément de la source ou de la destination.

Pour masquer ou afficher la zone source

- 1 Effectuez l'une des actions suivantes :
 - Cliquez sur .
 - Cliquez-droit dans la fenêtre d'édition des mappings et sélectionnez ou désélectionnez **Montrer Source**.

Pour masquer ou afficher la zone source

- 1 Effectuez l'une des actions suivantes :
 - Cliquez sur .

- Cliquez-droit dans la fenêtre d'édition des mappings et sélectionnez ou désélectionnez **Montrer Destination**.

Pour déplacer une zone de la fenêtre d'édition des mappings

- 1 Placez le pointeur de la souris sur le bord supérieur de la zone.
- 2 Cliquez sur ce bord gris foncé et attendez qu'un rectangle noir reprenant la forme de la zone apparaisse.
- 3 Déplacez le rectangle noir à l'endroit de votre choix en maintenant votre doigt appuyé sur le bouton de la souris.
- 4 Relâchez le bouton de la souris après que le rectangle noir symbolisant le volet est placé à l'endroit désiré.

Pour créer un mapping

- 1 Sélectionnez une boîte de mapping dans le schéma du scénario.
- 2 Effectuez l'une des actions suivantes :
 - Sélectionnez l'onglet **Mappings**.
 - Double-cliquez.
Double-cliquer permet l'affichage d'une fenêtre de dialogue contenant les mêmes données que l'onglet **Mappings**.

- 3 Cliquez sur .

Quand vous créez un mapping, une boîte **Sélectionner la source et la destination** apparaît dans laquelle vous choisissez :

- le composant source
- le type de document source
- le composant destination

Figure 5.3. Boîte de sélection de vos composants et du type de document source



 **Note :**

Cette boîte n'apparaît pas lorsque :

- un seul composant est lié en entrée et un seul en sortie de votre boîte de mapping,
 - aucun type de document produit n'a été créé pour le composant source.
-

Chemin des éléments d'un type de document

Pour localiser un élément de vos types de document, notamment dans le champ **Source pour le mapping de la collection**, vous devez indiquer le chemin de cet élément dans le type de document. Les relations de parenté des éléments de vos types de document sont signalées à l'aide d'un point. Le schéma suivant représente un ensemble d'éléments apparentés.

Pour indiquer le chemin qui mène à la collection X, vous devez écrire :

Structure A.Structure B.Collection X

Cette syntaxe signifie que la collection X est fille de la structure B, elle-même fille de la structure parente A.

Exemple :

Dans le mapping **Asset information** du scénario Desktop Discovery vers Asset Management, la collection **SubAssets** du type de document destination **amAsset** est mappée à la collection **Printer**. Comme cette collection est fille de la structure **Peripherals**, elle-même fille de la structure **Hardware** du type de document **Machine**, le champ **Collection source** indique le chemin suivant :

```
Hardware.Peripherals.Printer
```

Au cas où le nom d'un élément comprend lui-même un point. Vous devez faire référence à cet élément en l'entourant de simples guillemets :

```
'Element.A'
```

Exemple :

Exemple : de nombreux éléments des types de document publiés par le connecteur ServiceCenter comportent des points. Pour faire référence au champ **sw.vendor** de la collection **sw.vendor** dans le type de document **ICMDevice** publié par le connecteur ServiceCenter, vous devez indiquer le chemin suivant :

```
'sw.vendor' . 'sw.vendor'
```

Pour les champs ne comportant aucun point, les simples guillemets sont facultatifs. Vous pouvez indifféremment écrire :

```
ElementA ou 'ElementA'
```

Chemin des éléments d'un document

La notation du chemin des éléments d'un document est la même que celle employée pour les types de document. La seule exception concerne les membres d'une collection.

Dans un document, les membres d'une collection sont numérotés à partir de 0 : le premier membre porte le numéro 0, le second membre, le numéro 1, etc.

Pour indiquer un membre particulier dans un chemin, faites référence à son rang en utilisant la convention suivante :

Nom de la collection (rang du membre désiré)

Exemple : le chemin concernant le champ **VersionName** du premier membre de la collection **Software** est indiqué de la manière suivante :

`Software(0).VersionName`. Pour le deuxième membre de la collection, le chemin indiqué est `Software(1).VersionName`, etc.

Chemin d'une collection d'attributs

Pour indiquer le membre particulier d'une collection d'attributs (collection ne comportant qu'un champ), n'oubliez pas d'indiquer le nom de cet attribut dans votre chemin. Exemple : Pour indiquer les membres d'une collection **Address** comportant un attribut **address**, vous devez écrire : `Address(1).Address`, `Address(2).Address`, `Address(3).Address`, etc.

Note :

Quand le mapping collection à collection est effectué (champ **Source pour le mapping de la collection** renseigné), il n'est plus nécessaire de préciser les numéros des membres de la collection.

Pour modifier un mapping existant

- 1 Sélectionnez une boîte de mapping dans le schéma du scénario.
- 2 Effectuez l'une des actions suivantes :
 - Sélectionnez l'onglet **Mappings**.
 - Double-cliquez.
Double-cliquer permet l'affichage d'une fenêtre de dialogue contenant les mêmes données que l'onglet **Mappings**.
- 3 Effectuez l'une des actions suivantes :
 - Cliquez sur .
 - Double-cliquez sur le mapping que vous désirez éditer.

Pour supprimer un mapping

- 1 Sélectionnez une boîte de mapping dans le schéma du scénario.
- 2 Effectuez l'une des actions suivantes :
 - Sélectionnez l'onglet **Mappings**.
 - Double-cliquez.
Double-cliquer permet l'affichage d'une fenêtre de dialogue contenant les mêmes données que l'onglet **Mappings**.

- 3 Cliquez sur .

Pour copier/coller des parties de mapping

Vous pouvez copier dans le presse papier un mapping ou une partie d'un mapping d'un type de document source vers un type de document destination. Ensuite, vous pouvez coller ou fusionner le contenu du presse papier dans un autre mapping où le type de document destination du mapping copié est disponible. (Exemple : Dans un scénario utilisant le connecteur Asset Management, vous copiez une partie du mapping où le type de document destination est **amAsset** et vous le collez dans un autre mapping où le type de document destination est toujours **amAsset**).

Cette fonctionnalité est particulièrement intéressante lorsque vous avez à créer un nombre important de mappings soit identiques, soit grandement similaires.

- 1 Ouvrez un mapping d'un scénario existant.
- 2 Dans le plan de travail du mapping, placez le pointeur sur un des éléments suivants du type de document destination :
 - le noeud racine
 - une structure
 - une collection
 - un champ
- 3 Cliquez droit.
- 4 Sélectionnez **Copier le mapping** dans le menu contextuel qui s'affiche.
- 5 Ouvrez un autre mapping où le type de document destination du mapping copié est présent (soit comme noeud racine, soit comme structure, soit comme collection).
- 6 Dans le plan de travail, placez-vous à l'endroit où vous voulez ajouter ou fusionner votre mapping (un noeud racine, une structure, une collection).
- 7 Cliquez droit et sélectionnez **Coller le mapping** dans le menu contextuel qui s'affiche.
- 8 En fonction de vos besoins :
 - cliquez **Oui** pour fusionner le mapping : l'élément sélectionné est modifié par le mapping contenu dans le presse papier (Il y a fusion des deux mappings en un seul).

- Cliquez **Non** pour ajouter le mapping : le mapping du presse papier est ajouté au mapping existant sous la forme d'une nouvelle structure ou d'une nouvelle collection.

Pour décrire un mapping

L'éditeur de scénarios vous permet de décrire chaque noeud de votre mapping.

Exemple : description d'un script de mapping utilisé pour tester la présence d'un élément dans une structure ou une collection.

- 1 Editez un mapping.
- 2 Sélectionnez un noeud de votre mapping.
- 3 Sélectionnez l'onglet **Mapping** sous le plan de travail.
- 4 Saisissez votre description dans la zone de texte **Description**.
- 5 Enregistrez votre mapping.

La description est enregistrée dans le fichier **.scn**.

Pour prévisualiser les données source et destination

Dans la fenêtre d'édition des mappings, vous pouvez prévisualiser les données des applications source et destination. Note : la prévisualisation des données destination n'est possible qu'avec certains connecteurs.

- 1 Editez un mapping.
- 2 Positionnez votre pointeur dans la fenêtre des types de documents source ou destination.
- 3 Cliquez droit.
- 4 Choisissez **Voir les données** dans le menu contextuel.

Attendez qu'une fenêtre de prévisualisation des données s'affiche.

Editer la prévisualisation des données

Vous pouvez éditer le type de document source de votre mapping dans une fenêtre d'édition afin de prévisualiser les données de votre choix.

- 1 Editez un mapping.
- 2 Positionnez votre pointeur dans la fenêtre des types de documents source.
- 3 Choisissez **Editer un type de document pour la prévisualisation des données**.

- 4 Attendez qu'une fenêtre d'édition du type de document source du mapping s'affiche.
- 5 Ajoutez ou supprimez des éléments sur le plan de travail.
- 6 Cliquez sur .
- 7 Visualisez vos données source.
- 8 Cliquez sur **Fermer** pour revenir dans la fenêtre d'édition de votre mapping

Important : n'utilisez pas cette fonction pour éditer le type de document source.

Toutes les modifications apportées au type de document source sont perdues après la fermeture de la fenêtre de prévisualisation.

Pour classer les mappings

Les mappings que vous éditez sont classés par type de document produit par le connecteur source. Lorsque plusieurs mappings concernent le même type de document, il est possible de les classer pour déterminer dans quel ordre ils s'exécutent au moment de la production des documents. Dans certains cas, par exemple, un premier mapping permet de créer les enregistrements dans l'application destination alors que les mappings suivants permettent de mettre à jour ces enregistrements.

- 1 Sélectionnez un mapping.
- 2 Cliquez sur ▲ ou sur ▼ pour le remonter ou le descendre dans la liste des mappings.

Ordre de production des documents et d'exécution des mappings

Les règles suivantes s'appliquent :

- Mode test
C'est l'ordre des documents produits par le connecteur source visible dans l'onglet **Types de document produits** qui s'applique.
- Mode programmé
C'est l'ordre des documents tel qu'il est défini dans la fenêtre d'édition de la programmation qui s'applique. Dans cette fenêtre, un programmeur peut être associé à plusieurs types de document. C'est l'ordre dans lequel ces types de documents sont associés à un programmeur qui s'applique.

Dans ce cas, l'ordre indiqué dans l'onglet **Types de document produits** du connecteur n'a aucune incidence.

Pour chercher un élément dans la fenêtre d'édition d'un mapping

- 1 Sélectionnez un élément dans la zone des types de document source ou destination ou dans le plan de travail.
- 2 Effectuez une des actions suivantes :
 - Cliquez droit et choisissez **Rechercher** (Ctrl + F) dans le menu contextuel.
 - Cliquez sur .
- 3 Saisissez le nom de l'élément que vous désirez chercher dans le champ **Rechercher** et sélectionnez les options de votre choix :
 - Respect de la casse
 - Recherche vers le haut ou le bas de la fenêtre.
- 4 Cliquez sur **Rechercher**.

Pour rechercher l'élément suivant ou précédent :

- 1 Effectuez une des actions suivantes :
 - Cliquez droit et choisissez **Rechercher suivant** (F3) dans le menu contextuel.
 - Cliquez droit et choisissez **Rechercher Précédent** (Maj + F3) dans le menu contextuel.

Recherche - règles de fonctionnement

La recherche dans l'arborescence d'un type de document obéit aux règles suivantes :

- La recherche se limite aux éléments affichés dans les fenêtres des types de document publiés. Pour que la recherche s'applique aux sous-noeuds d'un noeud donné, vous devez au préalable le déplier.
- La recherche dans le plan de travail parcourt les types de document destination ainsi que les scripts de mapping.

Pour rechercher et remplacer un élément dans les scripts d'un mapping

- 1 Sélectionnez un élément dans le plan de travail.
- 2 Effectuez une des actions suivantes :
 - Cliquez droit et choisissez **Remplacer** (Ctrl + H) dans le menu contextuel.
 - Cliquez sur .
- 3 Saisissez le nom de l'élément que vous désirez chercher dans le champ **Rechercher** et sélectionnez les options de votre choix :
- 4 Saisissez le nom de remplacement de cet élément dans le champ **Remplacer par**.
- 5 Sélectionnez les options de votre choix :
 - Respect de la casse
 - Recherche vers le haut ou le bas de la fenêtre.
- 6 Cliquez sur **Rechercher**, **Remplacer** ou **Remplacer tout**.

Pour désactiver un mapping

Désactiver un mapping vous permet d'empêcher que le connecteur ne produise les documents nécessaires à ce mapping. Exemple : un mapping permet de supprimer des enregistrements dans une application destination correspondant à des biens n'existant plus dans une application source. Si vous voulez que la suppression de ces enregistrements soit ponctuelle, il vous suffit de désactiver ce mapping.

Pour désactiver un mapping, décocher la case correspondant à ce mapping dans l'onglet **Mappings**.

Types de mapping

Cette section détaille les différents types de mapping de l'éditeur de scénarios.

Le mapping champ à champ direct

Le mapping champ à champ direct consiste à mapper un champ source à un champ destination. Au moment du traitement des données, la valeur prise par le champ source renseigne le champ destination du document produit par la boîte de mapping. Cette opération s'effectue sans l'intervention du moteur Basic.

Exemple :

Le champ **Name** d'un type de document source est associé au champ **Name** d'un type de document destination.

Deux méthodes sont disponibles pour le mapping champ à champ direct :

- 1 Méthode Glisser-déplacer
 - Sélectionnez un champ dans la fenêtre **Source** ou **Destination**.
 - Glissez-déplacez-le jusqu'à un champ de la fenêtre opposée.
- 2 Méthode Icône
 - Sélectionnez les deux champs que vous voulez associer dans la zone source et la zone destination.
 - Cliquez sur .
- 3 Méthode Glisser-déplacer en deux temps
 - Sélectionnez un champ destination.
 - Glissez-déplacez-le sur le plan de travail.
 - Sélectionnez un champ source.
 - Glissez-déplacez-le dans le plan de travail sur la même ligne que votre champ destination.
- 4 Méthode Glisser-déplacer et script de mapping
 - Sélectionnez un champ destination.
 - Glissez-déplacez-le sur le plan de travail.
 - Ecrivez directement le nom du champ source dans le champ **Script de mapping**.

Chaque fois que vous faites un mapping champ à champ direct, le nom de votre élément source apparaît automatiquement dans le champ **Script de mapping**. Si vous saisissez manuellement le nom de l'élément source, son nom doit impérativement être mis entre crochets.

Figure 5.4. Élément source dans le script du mapping

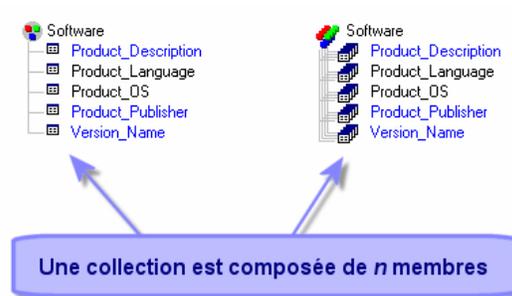
```
Script de mapping:
[Software.VersionName]
```

Le mapping collection à collection

Une collection est composée d'éléments (champs, structures, collections) dont la répétition est indéterminée.

Chaque répétition des éléments d'une collection constitue un membre de cette collection.

Exemple : dans le type de document **Machine** publié par le connecteur Desktop Discovery, les logiciels installés sur un ordinateur sont représentés par une collection. Chaque membre de cette collection contient la description d'un logiciel (nom, éditeur, version...).

Figure 5.5. Collection

Quand vous mappez une collection source à une collection destination, Connect-It calcule au moment du traitement des données le nombre de membres présents dans la collection source et crée autant de membres dans la collection destination.

Exemple :

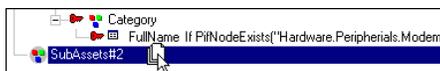
Dans le scénario Desktop Discovery vers AssetCenter, des éléments de la collection **Software** (logiciel) du type de document **Machine** sont associés aux éléments de la collection **Softinstall** (logiciels installés) du type de document **amAsset**.

Deux méthodes sont disponibles pour le mapping collection à collection :

- Méthode glisser déplacer
- Méthode clavier

Méthode glisser déplacer

- 1 Glissez-déplacez une collection destination sur votre plan de travail
- 2 Sélectionnez une collection source que vous glissez-déplacez dans le plan de travail sur la ligne de la collection destination en maintenant votre doigt appuyé sur le bouton gauche de votre souris et en maintenant enfoncée la touche **Ctrl** du clavier.
- 3 Relâchez votre doigt une fois qu'un document triple (symbole de la répétition des membres de la collection) apparaît sous votre pointeur.



Le champ **Source pour le mapping de la collection** de votre script de mapping est automatiquement renseigné avec le chemin de ce collection.

 **Note :**

Une boîte d'avertissement apparaît à chaque fois qu'un mapping d'un élément d'une collection est effectué, pour rappeler qu'il faut dans la majorité des cas, renseigner le champ **Source pour le mapping de la collection**. L'option **Afficher un avertissement chaque fois qu'un mapping collection à collection est nécessaire** (section **Confirmations**) du menu **Edition/ Options** permet de désactiver l'apparition de la boîte d'avertissement.

Méthode Clavier

- 1 Glissez-déplacez une collection destination sur votre plan de travail.

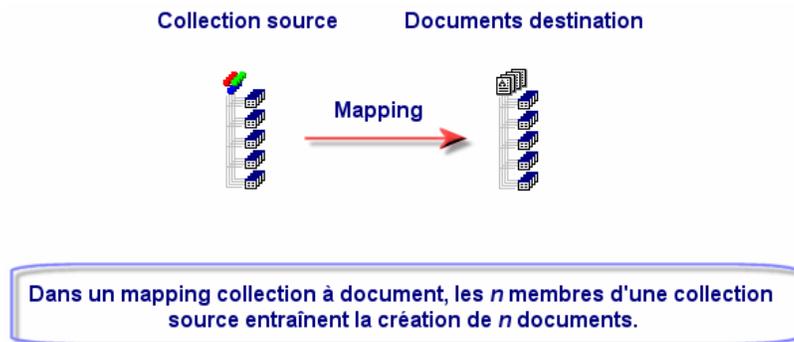
- 2 Ecrivez directement le chemin de votre collection source dans le champ **Source pour le mapping de la collection** de votre script de mapping. Comme vous ne faites pas appel à la valeur de cette collection, n'entourez pas son nom par des crochets.

Le mapping collection à document

Connect-It vous donne également la possibilité d'associer une collection source au noeud racine () d'un type de document destination. Au moment du traitement des données, Connect-It crée autant de documents destination qu'il existe de membres dans la collection source.

Un mapping collection à document s'effectue comme un mapping collection à collection. Dans ce cas, le noeud racine du type de document remplace la collection destination.

Figure 5.6. Mapping collection à document



Exemple :

Dans le scénario **scac\sc4ac41\scac.scn**, la collection **Software** (qui décrit les logiciels installés sur un ordinateur), est associée au type de document **pcsoftware**. Au moment du traitement des données, les n membres de la collection software donnent naissance à n documents de type **pcsoftware** qui, à leur tour, transmettent n événements entrants dans ServiceCenter.

Le mapping champ à collection

Dans un mapping champ à collection, on associe un ou plusieurs champs source n'appartenant pas à une collection, à un champ destination appartenant à une collection. La structure qui contient ce champ est considérée comme une collection composée d'un seul membre.

Exemple :

Le champ **CartesInstallées** regroupe toutes les cartes installées sur un ordinateur : carte mère, carte graphique, carte son. Dans le type de document source, ces informations sont représentées par des champs différenciés (champs **CarteMère**, **CarteGraphique**, **CarteSon**)

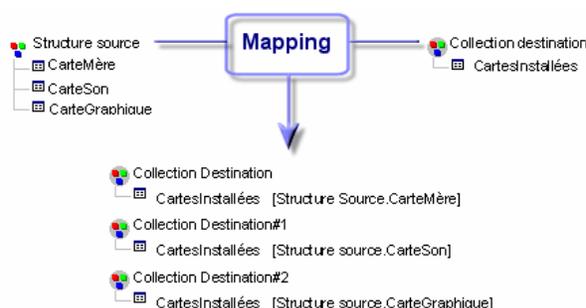
Après avoir glissé-déplacé votre collection sur le plan de travail, vous la dupliquez autant de fois que nécessaire et vous mappez le champ **CartesInstallées** de ces duplications avec les champs source.

Pour dupliquer une collection présente dans le plan de travail :

- 1 Sélectionnez cette collection.
- 2 Cliquez sur .

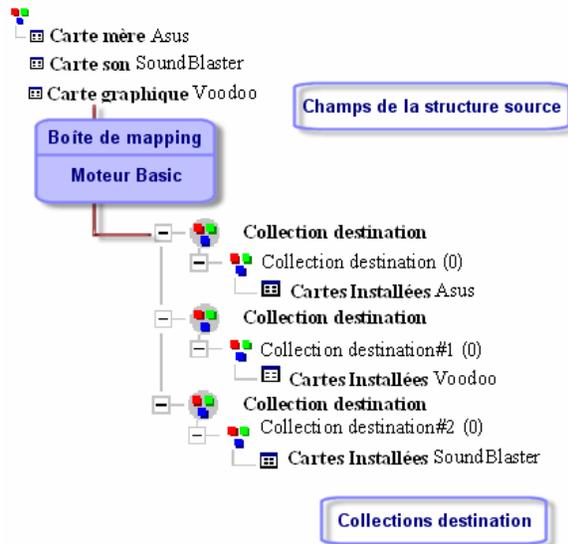
Le premier duplicata de la collection est indiquée par le suffixe #1, le second par le suffixe #2, etc.

Figure 5.7. Mapping champ à collection



Dans ce mapping champ à collection, les champs d'une structure source sont mappés aux membres d'une collection destination dupliquée.

Au moment du traitement des données, la boîte de mapping produit autant de collections à un membre qu'il existe de duplicatas de cette collection destination dans votre mapping. Chaque membre porte la valeur d'un des champs source.



Mapping des champs de type Blob

Les champs de type **Blob** (binary large object) sont des champs correspondant à des données binaires : fichier son, fichier vidéo, fichier image.

Un champ Blob ne peut être mappé qu'à un autre champ de type Blob.

Deux cas sont possibles :

- 1 Vous mappez le champ Blob destination au champ Blob source.
Dans ce cas, les fichiers binaires sont transmis de l'application source vers l'application destination via les connecteurs du scénario.
- 2 Vous mappez le champ Blob destination à un champ indiquant le nom du fichier binaire se trouvant dans un fichier partagé par Connect-It et l'application destination.

Dans ce cas, vous devez utiliser l'option **Charger le Blob à partir du fichier** disponible sous la zone de saisie des script de mapping.

Charger le Blob à partir du fichier

Cette option vous permet d'indiquer que :

- Le fichier Blob se trouve dans un dossier partagé par Connect-It et l'application destination.

Exemple : **[nom du dossier partagé sur réseau]/blob/pictures.**

- l'application destination charge le fichier blob à partir de ce dossier partagé.

Pour que l'application destination charge le fichier à partir du dossier partagé :

- 1 Dans le mapping, associez le nom du fichier **Blob** de la source à un fichier de type Blob de l'application destination.

Le nom du fichier de l'application source peut être le champ nom lié au fichier Blob dans l'application source (icon.name) ou directement le chemin complet du fichier dans le dossier partagé : **[nom du dossier partagé sur réseau]/blob/pictures/monimage.png.**

- 2 Sélectionnez l'option **Charger le Blob à partir du fichier.**

Pendant le lancement du scénario, l'application destination récupère le fichier **Blob** dans le dossier partagé.

Créer un document ou un membre de collection même si la collection source ne contient aucun membre

Si vous sélectionnez cette option, une collection ne contenant aucun membre produit par le connecteur source permet de créer :

- un document consommé par le connecteur destination si vous effectuez un mapping collection à document.

Consultez dans cette section, la sous-section [Le mapping collection à document](#) [page 116].

- un membre de collection dans le document consommé par le connecteur destination si vous effectuez un mapping collection à collection.

Consultez dans cette section, la sous-section [Le mapping collection à collection](#) [page 114].

Forcer la création d'un membre de collection ou celle d'un document permet de signaler cette absence de valeur dans l'application destination.

6 | Scripts de mapping

CHAPITRE

Quand le mapping champ à champ direct est impossible, un script de mapping est nécessaire (champ **Script de mapping**). Ce script indique comment sont manipulés des éléments source ou des données pour qu'ils soient associés à un élément destination.

Dans les mappings d'un scénario d'intégration, des scripts vous permettent :

- d'associer une valeur fixe à un élément destination
- d'associer une valeur calculée à un champ du type de document destination
- de subordonner le traitement des champs d'une structure ou d'une collection à la vérification d'une condition

Exemple :

Un script Basic permet la concaténation de deux éléments source. La valeur de retour de cette concaténation est associée à un élément destination au moment du traitement des données.

Référence de programmation

Afin de vous aider dans la rédaction de vos scripts, une référence de programmation en ligne est disponible dans Connect-It.

Cette référence de programmation est accessible par la touche **F1** quand votre pointeur se trouve à l'intérieur :

- du champ **Script de mapping** de la fenêtre d'édition du mapping
- de la zone de saisie des différents éditeurs

Fonctions Basic

Cette section vous présente certaines instructions Basic utilisées dans les mappings.

Fonctions If, Then, Else, Else If, End If

Syntaxe

```
If condition Then
  statements
Else If condition Then
  statements
Else
  statements
End If
```

Exemple

```
Dim strVal As String
(...)
If strVal = "" Then
  RetVal = "Empty"
ElseIf strVal = "Default" Then
  RetVal = "Default"
Else
  RetVal = "Unknown"
End If
```

Ce script retourne la valeur :

- "Empty" si le champ texte d'un document produit ne contient aucune information.

- "Default" si le champ texte d'un document produit contient l'information "default".
- "Unknown" si le champ texte d'un document produit contient tout autre information.

Fonction For Loop

Cette instruction permet de créer une boucle.

Syntaxe

```
For counters = start To end [Step increment]
  statements
Next
```

Exemple

```
For i=0 To 10 Step 2
  PifLogInfoMsg(i)
Next
```

Ce script retourne la valeur (i) dans le journal des documents.

La visualisation dans le journal des documents est la suivante :

```
0
2
4
6
8
10
```

Fonction While Loop

Cette instruction permet de créer une boucle.

Syntaxe

```
While loop
While condition
  statements
WEnd
```

Exemple

```
Dim i As Integer
i = 0
While i < 10
    i = i + 2
    PifLogInfoMsg(i)
WEnd
```

Ce script retourne la valeur (i) dans le journal des documents tant que (i) est inférieur à dix.

La visualisation dans le journal des documents est la suivante :

```
0
2
4
6
8
10
```

Fonction Return

Dans un script, si les conditions définies avant cette instruction ne sont pas respectées, le reste du script est ignoré.

Syntaxe

```
(...)  
Return  
(...)
```

Exemple

```
If [MacAdress] = "" And [IPAdress] = "" Then
    PifIgnoreNodemapping
    Return
End If

If [MacAdress] <> "" Then
    RetVal = [MacAdress]
Else
    RetVal = [IPAdress]
End If
```

Ce script teste les champs d'adresse MAC et IP pour un type de document produit. Si ces champs sont vides alors le noeud de mapping est ignoré et la fin du script n'est pas exécuté.

Fonction Select

Cette instruction permet d'exécuter un bloc d'instructions en fonction de la valeur d'une variable.

Syntaxe

```
Select Case testvar
  Case var1
    Statement Block
  Case var2
    Statement Block
  Case Else
    Statement Block
End Select
```

Exemple

Dans cet exemple :

- Le champ **seStatus** du document source contient l'état d'un ticket.
- L'état d'un ticket est :
 - 0 = ticket ouvert
 - 1 = ticket fermé

```
Select Case [seStatus]
  Case 0
    RetVal = "Opened"
  Case 1
    RetVal = "Closed"
  Case Else
    RetVal = "Unknown status"
End Select
```

Ce script associe la chaîne de caractères décrivant l'état du ticket à la valeur numérique du champ source. Si l'état n'est pas connu, la valeur "Unknown Status" sera retournée.

Les fonctions PIF

Les fonctions de type **PIF** ont été spécialement développées pour les scripts de mappings de Connect-It.

Toutes ces fonctions spécifiques sont documentées dans la référence de programmation en ligne disponible dans le sous-dossier **doc** du dossier d'installation de Connect-It.

Fonction PifIgnoreDocumentMapping

Cette instruction permet d'ignorer un document source et de ne pas l'envoyer vers le connecteur de destination.

Elle s'applique à n'importe quel élément complexe d'un document (noeud racine, collection, structure, champ).

Syntaxe

```
(...)  
PifIgnoreDocumentMapping("message")  
(...)
```

Vous pouvez ajouter un message d'erreur ("message"). Celui-ci est visible dans le Journal des documents pour le type de document produit de la boîte de mapping.

Si une instruction de type **retval** est spécifiée, cela suppose que l'instruction **PifIgnore** se fait sur un champ portant une clé de réconciliation.

Exemple

```
If [MacAdress] = "" Then  
  PifIgnoreDocumentMapping("Missing MacAdress")  
End If  
  
RetVal = [MacAdress]
```

Le champ adresse MAC sert de clé de réconciliation. Si ce champ ne contient aucune valeur, le document est ignoré. L'erreur générée (🔴 Missing MacAdress) est visible dans le journal des documents.

Fonction PifRejectDocumentMapping

Cette instruction permet de rejeter un document source et de ne pas l'envoyer vers le connecteur de destination.

Elle s'applique à n'importe quel élément complexe d'un document (noeud racine, collection, structure, champ).

Syntaxe

```
(...)  
PifRejectDocumentMapping("message")  
(...)
```

Vous pouvez ajouter un message d'erreur ("message"). Celui-ci est visible dans le Journal des documents pour le type de document produit de la boîte de mapping.

Si une instruction de type **retval** est spécifiée, cela suppose que le l'instruction **PifIgnore** se fait sur un champ portant une clé de réconciliation.

Exemple

```
If [MacAdress] = "" Then
  PifRejectDocumentMapping("Missing MACAdress")
End If

RetVal = [MacAdress]
```

Le champ adresse MAC sert de clé de réconciliation. Si ce champ ne contient aucune valeur, le document est rejeté. L'erreur générée (⚠ Missing MacAdress) est visible dans le journal des documents.

Fonction PifIgnoreNodeMapping

Cette instruction permet d'ignorer un quelconque élément complexe d'un type de document (collection, structure, champ).

Le comportement de l'instruction **PifIgnoreNodeMapping** est différent selon qu'elle porte sur une collection ou non.

Si cette instruction porte sur une collection, seul l'élément courant de la collection est ignoré. Si vous souhaitez ignorer l'ensemble de la collection, utilisez l'instruction **PifIgnoreCollectionMapping**.

Syntaxe

```
(...)  
PifIgnoreNodeMapping("Message")  
(...)
```

Vous pouvez ajouter un message d'erreur ⓘ ("message"). Celui-ci est visible dans le Journal des documents pour le type de document produit de la boîte de mapping.

Exemple

```
If [MacAdress] = "" Then
  ' Silently ignoring the element, no string given as parameter of PifIgn
```

```

oreNodeMapping
  PifIgnoreNodeMapping
End If
RetVal = [MacAdress]

```

Ce script permet d'éviter une mise à jour avec une chaîne vide si le champ ou la structure contenant l'adresse MAC est vide. Si le champ est renseigné, alors la mise à jour est effectuée.

```

If Left([Software.Name], 7) = "Windows" Then
  ' If soft name start with "Windows", ignore it
  PifIgnoreNodeMapping
ElseIf Left([Software.Name], 5) = "SunOS" Then
  ' If soft name start with "SunOS", ignore it
  PifIgnoreDocumentMapping
End If

```

Ce script permet de ne pas prendre en compte les données contenant "Windows" ou "SunOS"; les données ne sont pas mappées.

Fonction PifIgnoreCollectionMapping

Cette instruction permet d'ignorer une collection d'un type de document produit lors d'un mapping de collection à collection.

Pour plus d'informations sur le mapping collection à collection, consultez la section [Le mapping collection à collection](#) [page 114].

Syntaxe

```

(...)
PifIgnoreCollectionMapping
(...)

```

Exemple

```

Dim i As Integer
Dim iCount As Integer

' Number of item in the Logs collection
iCount = PifGetItemCount("Logs")

' Loop on all logs item to check if there is an error
For i=0 To iCount - 1
  ' Note that it is possible to use the loop index directly in the path to
  o a collection element ([Logs(i).LogType])
  If [Logs(i).LogType] = 1 Then
    ' LogType value is 1 if it corresponds to an error message.

```

```

        ' We have found an item with an error, no need to continue further
        Return
    End If
Next
' If here, means that there is no log item corresponding to an error
PifIgnoreCollectionMapping

```

Dans ce script, pour un bilan de traitement, tous les membres de la collection **logs** sont ignorés si il n'y a pas de message d'erreur.

Si le document ne contient aucune erreur, il n'est pas nécessaire de procéder à un tel script. Le champ **ErrorNumber** contient le nombre d'erreurs associées au document.

Le script précédent peut être remplacé par le suivant :

```

If [ErrorNumber] = 0 Then
    PifIgnoreCollectionMapping
End If

```

Variables globales

Lorsqu'une variable est déclarée dans un script de mapping, sa portée est locale.

Par exemple, si vous déclarez une variable dans un script associé à un champ, vous ne pourrez utiliser cette variable pour un autre élément du document.

Si le même nom de variable est utilisé dans un script de mapping effectué sur un autre élément, une nouvelle variable locale est créée, et la valeur de la variable du premier script ne peut être récupérée dans le deuxième.

Déclarer une variable globale

Pour que la portée d'une variable soit globale, il faut la déclarer à la racine d'un type de document, dans l'onglet **Scripts additionnels**, afin qu'elle puisse être calculée et utilisée dans différents scripts appliqués à différents éléments complexes d'un même mapping.

Note :

De façon à différencier les variables locales des variables globales, il est conseillé de préfixer ces dernières par **g_**.

Par exemple :

```
Dim g_lCounter As Long
```

La déclaration d'une variable globale doit être effectuée avant la définition des scripts Basic. La variable est prise en compte lors de l'ouverture ou la fermeture d'une session.

Par exemple :

```
Dim g_lCounter As Long
(...)

Sub OpenSession()
    rem add your code here
End Sub

Sub CloseSession()
    rem add your code here
End Sub
```

Créer un compteur de nombre de documents

Pour créer un compteur de nombre de documents, il vous faut dans un premier temps déclarer une variable globale, comme décrit dans la section [Variables globales](#) [page 129].

Par défaut, cette variable a pour valeur zéro.

Exemple

```
g_lCounter = g_lCounter + 1
PifLogInfoMsg(g_lCounter)
```

Chaque fois qu'un document est produit par un connecteur source, le compteur est incrémenté et le script retourne la valeur du compteur dans le journal des documents.

 **Note :**

Tant que le scénario est exécuté, le compteur n'est pas réinitialisé.

Pour réinitialiser la valeur du compteur à chaque nouvelle ouverture de session, il vous faut l'effectuer dans un script additionnel.

```
Dim g_lCounter As Long

Sub OpenSession()
    g_lCounter = 0
End Sub
```

En conservant à la racine du type du document le script d'incrémentation du compteur, les valeurs de la variables globales seront par exemple les suivantes :

```
Session 1
1
2
3
...
Session 2
1
2
3
...
```

Sauvegarder une variable globale dans un fichier

Si un scénario est arrêté alors qu'il tourne en mode service, la valeur courante de la variable globale est perdue.

Pour conserver la valeur courante de la variable globale, il vous faut la sauvegarder dans un fichier et la recharger depuis ce fichier lorsque cela est nécessaire.

Exemple

Dans cet exemple :

- Un compteur décompte le nombre de documents traités, sans jamais être réinitialisé.
- Ce compteur sera rechargé depuis un fichier si l'application est arrêtée.
- Sur l'élément racine, si la variable globale vaut 0, le script additionnel charge le fichier "C:/tmp/counter.txt" en vérifiant qu'il existe et qu'il contient des données.

Si la variable globale est égale à 0, cela implique que :

- Le scénario est exécuté pour la première fois.
- Le scénario a été interrompu.

```
Dim g_lCounter As Long
Sub OpenSession()
' Counter equals to 0, means that it has not been initialized.
' The application has been stopped and we reload the counter
' value
If g_lCounter = 0 Then
' If the file does not exist, it must be the first run. Check the
```

```

' existence of the file to avoid error on opening.
If FileExists("c:/tmp/counter.txt") Then

    ' Open the file in read mode
    Open "c:/tmp/counter.txt" For Input As #1

    ' Check the file contains data and read the first line of the file.
    If Not Eof(1) Then
        Line Input #1, g_lCounter
    End If
    Close #1
End If
End If
End Sub

```

Le script suivant permet de sauvegarder la valeur du compteur dans un fichier, chaque fois que le compteur est incrémenté sur l'élément racine :

```

g_lCounter = g_lCounter + 1
Open "c:/tmp/counter.txt" For Output As #1
Print #1, g_lCounter
Close #1

PifLogInfoMsg(g_lCounter)

```

Ce script peut être appliqué à n'importe quel élément d'un type de document.

Aide à la rédaction des scripts de mapping

Cette section présente quelques notions élémentaires qui vous aideront dans la rédaction de vos scripts de mapping.

Associer une valeur fixe à un élément destination

Pour associer une valeur fixe (une date, une chaîne de caractères, un nombre...) à un champ destination :

- 1 Glissez-déplacez le champ destination dans le plan de travail
- 2 Saisissez la valeur fixe dans le champ **Script de mapping**.

Exemple :

Exemple : dans le scénario Desktop Discovery vers Asset Management, chaque ordinateur inventorié donne lieu à la création d'un nouveau bien dans la table **amAsset**. Cette table exige par défaut que chaque bien ait un nom de catégorie. Dans votre définition de mapping, vous associez au champ **FullName.Category** du type de document **amAsset** la valeur fixe **"/Matériel/Ordinateur de bureau/"**.

Définir une variable globale

Vous pouvez définir des variables globales pour vos scripts de mapping.

Une variable globale est une variable qui a été définie à la racine d'un type de document et qui peut être appelée par un script associé à n'importe quel élément complexe d'un type de document (collection, structure, champ).

La déclaration d'une variable globale doit être effectuée avant la définition des scripts Basic. La variable est prise en compte lors de l'ouverture ou la fermeture d'une session.

La définition d'une variable globale se fait dans l'onglet **Scripts additionnels**.

Onglet Scripts additionnels

Cet onglet est actif uniquement pour l'élément racine d'un type de document.

Il permet de :

- Définir des variables globales
- Exécuter du code Basic à l'ouverture de chaque session
- Exécuter du code Basic à la fermeture de chaque session

Glisser-déplacer plusieurs champs

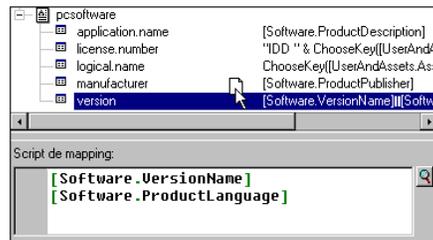
Dans vos scripts Basic, vous pouvez utiliser plusieurs champs source pour, par exemple, les concaténer à l'aide d'une fonction Basic.

Pour glisser-déplacer plusieurs champs :

- 1 Mappez un premier champ source à un champ destination
- 2 Glissez-déplacez les autres champs en maintenant appuyée une des touches **Maj.** de votre clavier. Les champs apparaissent les uns sous les autres dans

le volet **Script de mapping**. Vous n'avez plus qu'à rédiger le script utilisant ces champs.

Figure 6.1. Glisser-déplacer plusieurs champs



Retrouver vos éléments source et destination

Pour diverses raisons, vous pouvez vouloir retrouver la place d'origine d'un élément destination posé sur le plan de travail dans le type de document destination. Pour cela, vous devez :

- 1 Sélectionnez cet élément dans le plan de travail.
- 2 Cliquez sur .

Après cette action, l'élément destination d'origine devient la sélection courante dans le volet des types de document destination. Cela s'avère très utile quand le type de document destination contient un nombre important d'éléments. De même, les éléments source mappés (leur nom s'affiche en bleu) peuvent être retrouvés dans le plan de travail.

Pour cela, double-cliquez sur un élément source mappé dans la fenêtre du type de document source. Après ce double clic, les éléments destination qui font appel à cet élément source pour la définition du mapping apparaissent en vert.

Dupliquer un élément destination

Dans certains cas, vous avez besoin de dupliquer un champ destination, pour l'associer à différents éléments source. C'est notamment le cas pour une collection. Exemple : vous voulez associer les champs source X et Y au champ A d'une collection. Pour cela, il vous suffit de dupliquer la collection dans laquelle le champ A est contenu, de mapper le champ X au champ A de la

collection originale, et de mapper le champ Y au champ A de la collection dupliquée.

Pour dupliquer un élément :

- 1 Sélectionner un élément déjà présent dans le plan de travail.
- 2 Cliquez sur .

Les éléments dupliqués sont numérotés : l'élément d'origine porte le numéro 0 (qui n'apparaît pas), la première duplication le 1, la deuxième le 2, etc. Cette numérotation est imposée par Connect-It sans possibilité de modification de votre part.

Copier le chemin d'un élément

Élément dans le document

amAstProjDesc	
dIncluded	Date
dPlannedRemov	Date
dRemoved	Date
dtLastModif	Date et heure
Field1	Texte
Asset	
Project	
Document	
Comment	Champ texte long
dDocDate	Date

Chemin de l'élément dans le script de mapping

Script de mapping :
[Document.dDocDate]

Dans un script de mapping, vous devez indiquer le chemin complet des éléments que vous utilisez. Exemple : Dans le schéma suivant, le chemin complet du champ **dDocDate** est [**Document.dDocDate**].

Pour obtenir le chemin de l'élément d'un type de document :

- 1 Sélectionnez cet élément
- 2 Cliquez droit et choisissez **Copier le chemin** dans le menu contextuel qui s'affiche (**Ctrl +C**)
- 3 Placez votre pointeur dans le champ **Script de mapping**
- 4 Cliquez droit et choisissez Coller dans le menu contextuel qui s'affiche (**Ctrl+V**)

Utiliser le menu contextuel du script de mapping

En cliquant droit dans le script de mapping, un menu contextuel s'affiche.

Ce menu contextuel vous propose les fonctions suivantes (entre parenthèses les raccourcis clavier) :

- Annuler frappe
- Répéter frappe
- Couper (**Ctrl+X**)
- Copier (**Ctrl+C**)
- Coller (**Ctrl+V**)
- Sélectionner tout (**Ctrl+A**)
- Filtrer les types de document inutilisés
- Montrer source
- Montrer destination

Menus contextuels de la fenêtre d'édition des mappings

Dans la fenêtre d'édition des mappings, les commandes de quatre menus contextuels vous aident à rédiger vos scripts de mappings.

Pointeur placé sur un élément d'un type de document publié par le connecteur source

Commande	Fonction
Copier cette DTD XML	Copie la DTD du type de document produit par le connecteur source.
Copier le chemin	Copie le chemin de l'élément sélectionné dans le presse-papiers.
Montrer les noeuds associés à la sélection	Fait apparaître en vert les noeuds du mappings faisant intervenir l'élément sélectionné (mapping direct ou scripté).
Voir les données	Affiche une fenêtre permettant de voir les données du type de document source.
Editer un type de document à prévisualiser	Affiche une fenêtre vous permettant de créer un type de document produit par le connecteur source et dont vous pouvez voir les données.
Filtrer les types de document inutilisés	Filtre les types de documents non utilisés dans le mapping courant.
Montrer source	Affiche ou masque le volet du type de document source.

Commande	Fonction
Montrer destination	Affiche ou masque le volet du type de document destination.

Pointeur placé sur un des éléments d'un type de document publié par le connecteur destination

Commande	Fonction
Copier cette DTD XML	Copie la DTD du type de document consommé par le connecteur destination.
Copier le chemin	Copie le chemin de l'élément sélectionné dans le presse-papiers.
Ajouter cet élément	Place l'élément sur le plan de travail.
Voir les données	Affiche une fenêtre permettant de voir les données du type de document destination.
Filtrer les types de document inutilisés	Filtre les types de documents non utilisés dans le mapping courant.
Montrer source	Affiche ou masque le volet du type de document source.
Montrer destination	Affiche ou masque le volet du type de document destination.

Pointeur placé sur le plan de travail

Commande	Fonction
Copier cette DTD XML	Copie la DTD du type de document consommé par le connecteur destination
Copier le chemin	Copie dans le presse-papiers le chemin de l'élément sélectionné
Editer de mapping	Permet d'éditer le script du mapping à l'aide de l'éditeur de texte
Décrire le mapping	Affiche une fenêtre vous permettant de décrire le noeud de mapping sélectionné.
Clé de réconciliation	Indique que l'élément sélectionné est utilisé comme clé de réconciliation dans le cas d'un connecteur destination de type base de données
Enlever cet élément	Enlève l'élément sélectionné du plan de travail
Copier le mapping	Permet de copier le mapping en totalité ou en partie dans le presses papiers.

Commande	Fonction
Coller le mapping	Permet de coller un mapping ou une partie de mapping du presse papiers dans le mapping courant.
Filtrer les types de document inutilisés	Filtre les types de documents non utilisés dans le mapping courant.
Montrer source	Affiche ou masque le volet du type de document source
Montrer destination	Affiche ou masque le volet du type de document destination

Pointeur placé la zone éditable Script du mapping

Commande	Fonction
Annuler Frappe	Annule la dernière frappe
Répéter Frappe	Répète la dernière frappe
Couper	Coupe le texte sélectionné
Copier	Copie le texte sélectionné dans le presse-papiers
Coller	Colle le contenu du presse-papiers dans la zone éditable
Sélectionner tout	Sélectionne l'ensemble du texte de la zone éditable
Filtrer les types de document inutilisés	Filtre les types de documents non utilisés dans le mapping courant.
Montrer source	Affiche ou masque le volet du type de document source
Montrer destination	Affiche ou masque le volet du type de document destination

Chaîne de caractères

Concaténer une chaîne de caractères

L'opérateur & permet de concaténer des chaînes de caractères.

Si l'une des opérands n'est pas de type chaîne de caractères, une conversion en chaîne de caractères est automatiquement effectuée.

Exemple

```
RetVal = "Current date: " & Date()
```

Ce script affiche une date selon le format suivant :

```
Current Date : 2002/12/26
```

Créer une chaîne de caractères à partir du code d'un caractère

La fonction **Chr()** permet de générer un caractère selon la page de code utilisée par Connect-It.

Cette fonction est utile pour générer notamment des caractères de type :

- guillemets simples : Chr(39)
- guillemets doubles : Chr(34)
- retour chariot : Chr(13)
- saut de ligne : Chr(10)

Exemple

```
Dim lVal As Long
lVal = 5
RetVal = "Value: " & Chr(39) & lVal & Chr(39)
```

Ce script renvoie la valeur :

```
Value: '5'
```

Chaîne de caractères UNIX / Windows

Le retour à la ligne n'est pas géré de la même façon selon que l'on travaille sous UNIX ou Windows.

Sous système Windows le retour à la ligne est géré par les caractères Chr(13) et Chr(10). Sous UNIX, seul le caractère Chr(10) est utilisé.

Exemple

```
RetVal = Replace([WindowsText], Chr(13) & Chr(10), Chr(10))
```

Ce script remplace tous les retour à la ligne Windows par ceux d'UNIX dans une chaîne de caractères récupérée dans l'élément **WindowsText** du document source.

Edition des fichiers associés

Les scripts de mapping nécessitent parfois l'utilisation de fichiers associés.

Ces fichiers associés contiennent :

- des tables de chaînes de caractères
Pour une présentation des tables de chaînes de caractères, consultez la section de ce chapitre.
- des tables de correspondance
Pour une présentation des tables de correspondance, consultez la section de ce chapitre.
- des fonctions et des variables globales
Pour une présentation des fonctions et des variables globales, consultez la section de ce chapitre.

Ces fichiers sont enregistrés indépendamment du fichier **.scn** dans des fichiers dont l'extension de nom correspond à leur contenu.

Extension des noms de fichier

.str (pour string)	Tables de chaînes de caractères
.mpt (pour mactable)	Tables de correspondance
.bas (pour Basic)	Fonctions et variables globales

Note :

Un scénario (un fichier SCN) ne disposant pas de ses fichiers **.str**, **.mpt** et **.bas** associés ne fonctionne pas. Veillez que, chaque fois que vous déplacez un scénario, ses fichiers attachés soient également déplacés. Prenez également soin de conserver la place de ces fichiers vis-à-vis de votre fichier SCN. Le mauvais emplacement des fichiers associés empêcherait au scénario de récupérer leur contenu au moment du traitement des données.

Editer un fichier associé

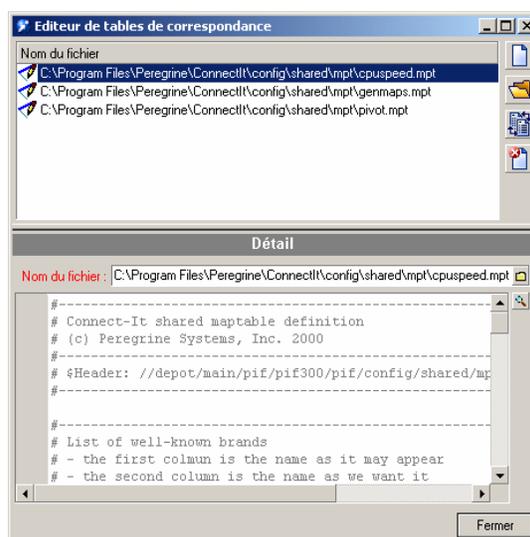
L'édition de ces éléments se fait par l'intermédiaire d'éditeurs propres à Connect-It.

Pour accéder à ces éditeurs, utilisez le menu **Scénario**.

Chaque fois que vous accédez à un éditeur par le menu **Scénario**, une fenêtre modale identique s'affiche. Elle se compose de deux volets :

- Un volet contenant les fichiers (**.str**, **.mpt** ou **.bas**) associés au scénario courant.
- Un volet d'édition des fichiers (tables de chaînes de caractères, tables de correspondance, fonctions et variables globales).

Figure 6.2. L'éditeur des tables de correspondance



Ces trois éditeurs disposent d'un éditeur de texte également commun au script de mapping.

Pour créer un nouveau fichier associé

- 1 Cliquez sur  dans la barre d'outils
- 2 Indiquez le nom de votre fichier dans le champ **Nom du fichier**

En cliquant sur , vous avez la possibilité de parcourir votre disque dur pour indiquer directement le dossier où vous voulez créer ce fichier.

3 Cliquez sur **Créer**

 **Avertissement :**

Si le fichier existe déjà, il sera supprimé.

Pour ouvrir un fichier associé existant

Pour ouvrir un fichier existant, vous disposez de deux méthodes.

Pour ouvrir un fichier figurant dans le volet des fichiers associés

Pour ouvrir un fichier figurant dans le volet des fichiers associés au scénario courant, cliquez sur ce fichier.

Pour ouvrir un fichier ne figurant pas dans le volet des fichiers associés

Pour ouvrir un fichier ne figurant pas dans le volet des fichiers associés au scénario courant :

- 1 Cliquez sur  dans la barre d'outils.
- 2 Indiquez le chemin complet du fichier dans le champ **Nom du fichier**.
Vous avez la possibilité de parcourir votre disque dur pour trouver ce fichier.
- 3 Cliquez sur **Ouvrir**.

Pour dupliquer un fichier associé

- 1 Sélectionnez ce fichier dans la fenêtre des fichiers associés.
- 2 Cliquez sur  dans la barre d'outils.
- 3 Saisissez le nom de votre fichier dupliqué dans le champ **Nom du fichier**.
En cliquant sur , vous avez la possibilité de parcourir votre disque dur pour indiquer directement le dossier où vous voulez créer ce fichier.

Pour supprimer un fichier associé

Pour supprimer un fichier associé au scénario courant :

- 1 Sélectionnez ce fichier dans la fenêtre des fichiers associés.
- 2 Cliquez sur  dans la barre d'outils.

Validation des scripts

La cohérence des scripts est par défaut contrôlée en permanence par Connect-It. La validation teste à la fois la syntaxe du Basic et la présence dans le document source des éléments référencés dans le script du mapping (seuls les éléments visibles dans la structure du document source sont considérés comme présents pour la validation).

 **Note :**

L'option **Ne pas autoriser la validation d'un script qui comporte des erreurs de compilation** (section **Confirmations**) permet d'activer ou de désactiver le comportement précédemment décrit.

Editer le texte d'un fichier associé

Vous accédez à l'éditeur de texte en cliquant sur l'icône  présent dans chaque éditeur et à côté du champ **Script du mapping** de la fenêtre d'édition du mapping.

Cet éditeur de texte vous permet de rédiger plus confortablement le contenu de vos scripts Basic et de vos fichiers **.str**, **.mpt** et **.bas**. Il dispose des fonctions traditionnelles d'un éditeur de texte : copier/coller, annuler l'action courante, etc. Vous accédez à ces fonctions par le menu **Edition** ou la barre d'outils.

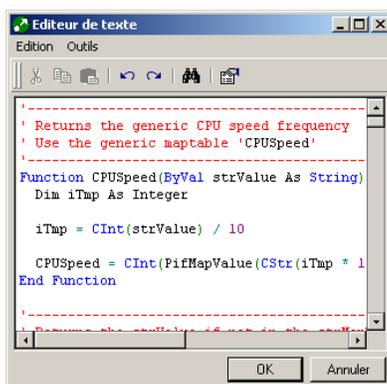


Tableau 6.1. Editeur de texte - barre d'outils

Icône	Fonction
	Coupe le texte sélectionné
	Copie le texte du presse-papiers
	Colle le texte du presse-papiers
	Annuler l'action courante
	Refaire l'action annulée
	Affiche la fenêtre de configuration de l'éditeur de texte

Configuration de l'éditeur de texte

Pour configurer l'éditeur de texte :

- Cliquez sur  ou
- Sélectionnez **Options** dans le menu **Composants**

Configurer l'éditeur de texte vous permet de modifier le modèle utilisé pour la rédaction de vos fichiers.

Avertissement :

Afin d'éditer un des modèles proposés par l'éditeur de texte, vérifiez que le fichier **codeedit.cfg** présent dans le sous-dossier **bin32** du dossier d'installation de Connect-It n'est pas en lecture seule.

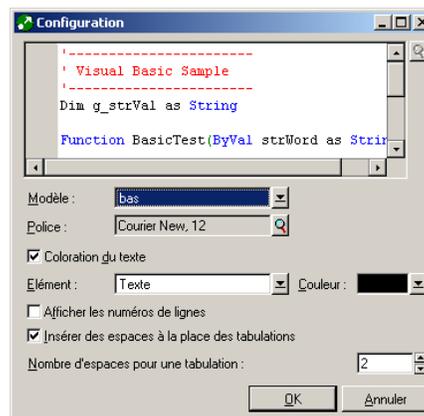
Pour chaque éditeur, un modèle est imposé. Exemple : si vous rédigez une table de correspondance, le modèle **mpt** (maptables) est imposé par l'éditeur de texte.

Nom du modèle	Rédaction
bas	des scripts de mapping des fonctions globales
str	des tables de chaînes de caractères
mpt	des tables de correspondance
sql	des requêtes SQL
scn	des scénarios de Connect-It
Default	de tous les fichiers autres que ceux énumérés

Pour un modèle donné, vous pouvez :

- modifier sa police de caractère (type, corps, style)
- choisir la coloration de ses éléments
- afficher ou non ses numéros de lignes
- insérer de un à huit espaces à la place des tabulations

Figure 6.3. Configuration de l'éditeur de texte



Modification d'un modèle

Pour modifier la police de caractères utilisée dans un modèle, cliquez sur la loupe présente à côté du champ **Police**. Une boîte de dialogue apparaît vous permettant de choisir une police parmi celles installées sur votre poste.

Vous pouvez également modifier la couleur des éléments qui apparaissent dans le texte. Ces éléments sont :

- le texte
- le fond du texte
- le texte sélectionné
- le fond du texte sélectionné
- les numéros de ligne
- les valeurs numériques
- les délimiteurs
- les commentaires
- les chaînes de caractères
- les mots clés

- les opérateurs

Pour colorer un élément, sélectionnez-le dans le champ **Élément** et choisissez sa couleur dans le champ **Couleur**.

Vous pouvez également choisir de ne pas colorer votre texte en décochant la case **Coloration du texte**.

Pour afficher les numéros de ligne et insérer des espaces à la place des tabulations vous cochez les cases correspondantes.

Fonctions et variables globales

Pour accéder à l'éditeur de tables des fonctions et variables globales utilisez le menu **Scénario**.

Vos scripts de mapping font appel à des fonctions et des variables dont vous pouvez avoir besoin à plusieurs endroits dans votre script. Ces fonctions et variables globales peuvent être sauvegardées au sein d'un même fichier **.bas**. Ces fonctions et variables sont appelées dans le script de mapping. Au moment du traitement des données, la boîte de mapping utilise ces fonctions et variables en les prenant dans les fichiers **.bas** attachés à votre scénario.

La fonction globale **CPUSpeed()** est contenue dans le fichier **gen.bas**. Cette fonction fait référence à une table de correspondance **CPUSpeed()** et permet d'indiquer la valeur générique de la vitesse d'un microprocesseur.

```
'-----
'-----
' Returns the generic CPU speed frequency
' Use the generic mactable
' CPUSpeed'
Function CPUSpeed(ByVal strValue As String) As Integer
    Dim iTmp As Integer
    iTmp = CInt(strValue) / 10
    CPUSpeed = CInt(PifMapValue(CStr(iTmp * 10), "CPUSpeed", 1, strValue))
End Function
```

Tables de chaînes de caractères

Pour accéder à l'éditeur de tables de chaînes de caractères utilisez le menu **Edition**.

Dans une table de chaînes de caractères, chaque chaîne est associée à un identifiant unique. Au moment du traitement des données, les scripts Basic remplacent les identifiants par leur chaîne de caractères respective.

Pour obtenir une chaîne de caractères dans votre mapping, vous devez utiliser la fonction **PifStrVal** suivi de l'identifiant de la chaîne entre guillemets et parenthèses.

L'exemple suivant vous montre une partie de la table des chaînes de caractères **category.str** utilisée dans le scénario Desktop Discovery vers Asset Management.

```
CAT_UNIX, "/Hardware/Unix Workstation"
CAT_SERVER, "/Hardware/Server"
CAT_WORKSTATION, "/Hardware/PC"
CAT_MAC, "/Hardware/Mac"
CAT_TERMINAL, "/Hardware/Terminal"
CAT_PORTABLE, "/Hardware/Portable"
```

Exemple :

Ainsi pour obtenir la valeur de "/Hardware/Unix Workstation"; il suffit d'utiliser la ligne de code suivante dans votre script de mapping :

```
PifStrVal ("CAT_UNIX" )
```

Dans cette table, l'identifiant CAT_UNIX correspond à la valeur "Hardware/Unix Workstation". Par la suite, si vous changez le nom de cette chaîne de caractères, votre scénario tient compte de ce changement sans que vous ayez à recompiler les scripts Basic où son identifiant est évoqué. Vous pouvez également disposer de plusieurs fichiers de tables de chaînes de caractères correspondant à une langue donnée : **frcategory.str** pour le français, **grcategory.str** pour l'allemand, etc.

Tables de correspondance

Pour accéder à l'éditeur de tables de correspondance utilisez le menu **Scénario**.

Dans une table de correspondance, la première colonne contient un clé et les colonnes suivantes des valeurs. Chaque clé correspond à une valeur dans une colonne donnée.

Pour obtenir la valeur d'une colonne donnée dans votre mapping, vous utilisez la fonction **PifMapValue()** avec en paramètre la clé, une valeur par défaut si

la clé n'est pas définie dans la table de correspondance, la mention du nom de votre table de correspondance et celle d'un numéro de colonne.

Au moment du traitement des données, chaque clé trouvée dans votre script est remplacée par sa valeur dans la colonne indiquée.

Dans l'extrait de la table de correspondance **Brand**, les clés **Apple** et **MacIntosh** retournent la valeur "Apple" si la colonne 1 est indiquée dans le script.

```
{ MapTable Brand
Compaq | Compaq
IBM -Lexmark| IBM - Lexmark
Hewlett Packard| Hewlett Packard
HP-UX| Hewlett Packard
Toshiba | Toshiba
Apple | Apple
Macintosh | Apple}
```

Exemple :

Ainsi pour obtenir la valeur de "HP-UX"; il suffit d'utiliser la ligne de code suivante dans votre script de mapping : `PifMapValue([Nom du champ dans le mapping], "Brand", 0, "DefaultValue")`

Pour une description complète de cette fonction reportez-vous à la référence de programmation en ligne, fonction **PifMapValue()**

Créer des tables de correspondance pour des langues différentes

Pour obtenir des tables de correspondance renvoyant des chaînes dans des langues différentes :

- 1 Créez des fichiers de chaînes de caractères où chaque ligne associe un identifiant à une chaîne localisée (français, anglais, polonais, etc.) sous la forme [Identifiant], ["chaîne localisée"]. Exemple : un fichier **category.str** contient, pour l'anglais, la ligne `CAT_UNIX, "UNIX Workstation")`, un autre fichier **category.str**, pour le français, contient la ligne `CAT_UNIX, "Station de travail UNIX")`, etc.
- 2 Incluez ce fichier dans votre fichier de tables de correspondance en utilisant la syntaxe suivante : `#include_str "[Nom du fichier]"`. Exemple : `#include_str "category.str"`

- 3 Référez dans la table de correspondance chaque chaîne en l'associant à un identifiant de la forme [dollar]([Identifiant]).

```
#include_str "category.str"
{MapTable Category
[dollar](IDS_CAT_UNIX) | workstation}
{ MapTable Sc2AcCat
[dollar](IDS_CAT_UNIX) | workstation}
```

Les tables de correspondance dynamiques

Une table de correspondance dynamique vous permet d'exécuter une requête SQL sur une base de données ODBC et de récupérer les valeurs retournées par la requête sous la forme d'une table de données, appelée table de correspondance. Le contenu de cette table de données est mis à jour dynamiquement à chaque fois que vous testez ou démarrez votre scénario.

Pour créer une table de correspondance dynamique, vous devez utiliser la fonction **PifCreateDynaMaptable**. Pour l'utilisation de cette fonction, reportez-vous à la référence de programmation en ligne.

Formats utilisateur

Les formats utilisateur sont des formats de type date ou numérique que l'utilisateur peut définir pour les utiliser dans ses scripts de mappings. Ces formats s'utilisent exclusivement avec les fonctions **PifUserFmtVarToStr** et **PifUserFmtStrToVar**. Pour l'utilisation de ces fonctions, reportez-vous à la référence de programmation en ligne disponible avec la touche F1 chaque fois que vous rédigez un script dans Connect-It.

Créer un format utilisateur de type date

Pour créer un format utilisateur de type date :

- 1 Sélectionnez **Formats utilisateur** dans le menu **Scénario**.
- 2 Cliquez sur **Suivant** dans la page de l'assistant qui s'affiche.

Cette page affiche les formats de type date qui ne peuvent pas être définis par l'utilisateur.

- 3 Cliquez sur  dans la page de l'assistant qui s'affiche

- 4 Cliquez dans la colonne **Nom** sous le nom du dernier format de la liste des formats existants
- 5 Nommez votre format dans la zone éditable qui est apparue
Ce nom ne doit comprendre aucun espace.
- 6 Cliquez dans la colonne **Format** correspondant au nom que vous venez de saisir
- 7 Saisissez une formule utilisant les attributs du volet **Symbole de format de type date**
- 8 Cliquez sur ✓.
Un exemple du format créé apparaît dans le champ **Résultat du format de type date**.

Créer un format utilisateur de type numérique

Pour créer un format utilisateur de type numérique :

- 1 Sélectionnez **Scénario/Formats utilisateur**.
- 2 Cliquez sur **Suivant** dans la page de l'assistant qui s'affiche.
Cette page affiche les formats de type numérique qui ne peuvent pas être définis par l'utilisateur.
- 3 Cliquez sur ✎.
- 4 Cliquez dans la colonne **Nom** sous le nom du dernier format de la liste des formats existants
- 5 Nommez votre format dans la zone éditable qui est apparue
Ce nom ne doit comprendre aucun espace.
- 6 Cliquez dans la colonne **Format** correspondant au nom que vous venez de saisir
- 7 Saisissez une formule utilisant les attributs du volet **Symbole de format numérique**.
- 8 Cliquez sur ✓.
Un exemple du format créé apparaît dans le champ **Résultat du format numérique**.

Syntaxe à utiliser pour la création des formats

La rédaction des formules de format doit respecter les règles suivantes :

- L'utilisation des symboles utilisés pour les formats de type date ou numérique est identique à celle utilisée par les systèmes d'exploitation Windows.
- Les chaînes de caractères apparaissant dans les formats doivent être entourées de guillemets simples
- Les espaces entre deux valeurs doivent être saisies comme une chaîne de caractères signalés par des guillemets simples

Tableau 6.2. Exemples de formats de type date

Formule du format	Exemple
yyyy'-mm'-dd	2002-02-07
hh':nn':ss	11:55:29
h':nn':ss	11:55:41
hh':nn	14:18
hh" 'h' "nn	15:54:53
h t' 'nn' 'tt	2 29 pm
dd'/'MM'/'yy	07/02/02
dd'.'MM'.'yy	07.02.02
dd'-'MM'-'yy	07-02-02
dd'/'MM'/'yyyy	07/02/2002
dddd' 'd' 'MMMM' 'yyyy	jeudi 7 février 2002

Tableau 6.3. Exemples de formats de type numérique

Formule du format	Exemple
n' 'n',dd-'USD'	1 0 2 0 3,41-USD
-'USD'n'.'nnn',ddd	-USD10.203,408
-n'.'nnn',dd'USD'	-10.203,41USD
-n'.'nnn',ddd'USD'	-10.203,408USD

Collection - exemples

Dans cette section, vous trouverez différents exemples de scripts portant sur un des éléments complexes d'un type de document : la collection.

Créer des membres dans une collection à partir d'une liste de valeurs

Cette section vous présente un exemple de script permettant de créer un membre dans une collection donnée à partir d'une liste de valeurs d'un document source.

Dans cet exemple :

- Le champ source **Software** contient une liste de valeurs.
- Les valeurs sont délimitées par un séparateur donné.

Le script :

- Extrait le nom des logiciels un par un.
- Crée un membre dans la collection destination **SoftInstalled**.
- Renseigne l'élément **Name** avec le nom du logiciel extrait.

```
Dim iCount As Integer
Dim iIndex As Integer
Dim strSoft As String
Dim lDummy As Long
Dim strPath As String

' Décompte du nombre de valeurs dans le champ source "Software"
' les noms des logiciels sont délimités par le séparateur virgule (','),
' par exemple "Excel, Connect-It,
' AssetCenter"
iCount = CountValues([Software], ",")

' Boucle sur tous les éléments de la liste de façon à les extraire un à un.
For iIndex = 0 To iCount - 1

    strSoft = GetListItem([Software], ",", iIndex+1)

    ' Suppression des espaces autour du nom du logiciel
    strSoft = Trim(strSoft)

    ' Création du chemin de la collection destination à partir de l'élément
    ' racine.
    ' Par exemple, pour le 3eme logiciel de la source, le chemin "SoftInstalled(3).Name" est créé
    strPath = "SoftInstalled" & iIndex & ".Name"

    ' Affectation de la valeur courante du logiciel de type chaîne de caractères à ce chemin en utilisant
    ' la fonction PifSetStringVal.
    ' La fonction PifSetStringVal retournant un code d'erreur si le chemin n'est pas valide, il est
    ' nécessaire d'affecter dans une variable la valeur de retour de la fonction. La fonction ne serait
```

```
' pas appliquée dans le cas contraire.
lDummy = PifSetStringValue(strPath, strSoft)

Next iIndex
```

Ce script de mapping peut être appliqué sur un élément quelconque du type de document destination.

Pour une meilleure lisibilité du mapping, nous vous conseillons d'effectuer ce mapping sur la collection à laquelle des membres doivent être ajoutés.

Avertissement :

L'élément indiqué par son chemin dans l'appel à la fonction **BasicPifSetStringValue** doit être présent dans le type de document destination. Dans l'exemple présent, l'élément **Name** de la collection **SoftInstalled** doit être ajouté par l'utilisateur dans le type de document consommé.

Concaténer des membres d'une collection dans un champ

Dans cet exemple :

- Le document source contient une collection de valeurs.
- Les éléments de cette collection sont mappés à un champ du type de document destination.

La source contient la collection des logiciels installés sur une machine. Les différents noms de logiciels doivent être écrits dans un champ contenant la liste des logiciels séparés par une virgule (',').

```
Dim iCollectionCount As Integer
iCollectionCount = PifGetItemCount("SoftInstalled")

Dim strList As String
Dim iItem As Integer

Pour chaque élément de la collection, récupérer le nom du logiciel (élément
"Name" de la collection "SoftInstalled") et le concatener avec la liste
courante.
For iItem = 0 to iCollectionCount - 1

    ' Ajouter le séparateur de nom si la liste n'est pas vide
    If strList = "" Then
        strList = strList & ", "
    End If

    ' Ajouter le nom du logiciel à la liste courante.
```

```

' Remarquez qu'il est possible d'utiliser directement une variable pour
indiquer le numéro
' d'un membre dans une collection. Par exemple, si la variable de iItem
vaut 3, le chemin
' [SoftInstalled(3).Name] sera automatiquement créé à partir de la vale
ur de iItem.
strList = strList & ", " [SoftInstalled(iItem).Name]
Next iItem

' Affecter la variable strList à l'élément destination
RetVal = strList

```

Mapper plusieurs champs dans une collection

Dans cet exemple :

- Le document source contient plusieurs champs distincts.
Ici, les champs **Address1** et **Address2** comportent les 2 adresses possibles d'un client.
- La valeur de ces champs doit être associée à un membre de la collection destination.
Ici, la collection **Adress**.

Par exemple :

Il faut donc :

- Créer deux membres dans la collection destination et les associer aux champs "Adress1" et "Adress2".
- Utiliser la fonctionnalité de duplication de collection :
 - 1 Ajouter la collection **Adress** dans le type de document destination.
 - 2 Dupliquer cette collection.
La collection **Adress#1** apparaît dans le type de document destination.
 - 3 Les scripts de mappings [Adress1] et [Adress2] sont à appliquer respectivement sur les champs **Adress.Adress** et **Adress#1.Adress**

Ignorer certains membres d'une collection dans un mapping collection a collection

Pour ignorer certains membres d'une collection, il vous faut utiliser les instructions **PifIgnoreCollectionMapping** et **PifIgnoreNodeMapping**.

Pour plus de renseignements sur ces instructions, consultez la section [Fonction PifIgnoreCollectionMapping](#) [page 128].

7 | Types de document pivots

CHAPITRE

Un type de document pivot est une représentation générique d'une entité (un bien, un logiciel, une personne, etc.). Un connecteur source établit la correspondance entre ses propres types de document et les types de document pivots (en production et en consommation, en fonction de ses caractéristiques). Les connecteurs qui utilisent les types de document pivots peuvent être liés sans qu'aucun mapping n'ait besoin d'être défini.

Les types de document pivots sont disponibles pour les connecteurs suivants :

- Connecteur Asset Management (versions du logiciel 4.xx et 4.1)
- Connecteur ServiceCenter (versions du logiciel 3.xx et 4.xx)
- Connecteur InfraTools Desktop Discovery
- Connecteur InfraTools Management
- Connecteur Unicenter AMO
- Connecteur PDA Inventory
- Connecteurs Tivoli Inventory
- Connecteur Winpark Actima
- Connecteurs Passerelles 3.xx

Principe de fonctionnement des types de document pivot

Les types de document pivots sont des types de document génériques. Un connecteur source définit le mapping de ses types de documents publiés vers les types de documents pivot. Un connecteur destination définit le mapping des types de documents pivot vers ses types de documents consommés. Il est alors possible d'échanger des données d'un connecteur source quelconque vers un connecteur destination quelconque sans avoir à définir de mapping spécifique. Les mappings de/vers les types de documents pivot suffisent à définir les transformations à effectuer pour échanger des données entre la source et la destination.

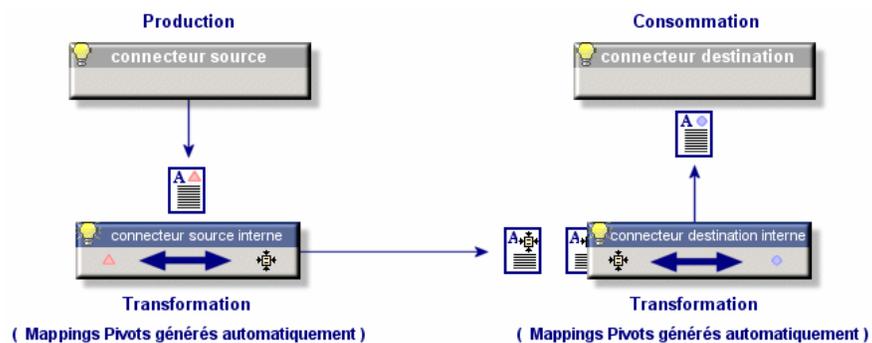
Mappings créés automatiquement

Quand un connecteur source produit pour la première fois des documents basés sur le type de document pivot, un processus automatique crée :

- un mapping entre le type de document produit par le connecteur source et le type de document pivot
- un mapping entre le type de document pivot et un type de document consommé par le connecteur destination

Ces deux mappings se basent sur des mappings pivots qui mappent chaque type de document publié par un connecteur au type de document pivot.

Figure 7.1. Utilisation d'un type de document pivot



Créer un scénario d'intégration en utilisant les types de document pivots

Cette section vous indique comment créer un scénario d'intégration utilisant les types de document pivot. Ce scénario utilise un connecteur source et un connecteur destination.

Pour créer ce scénario :

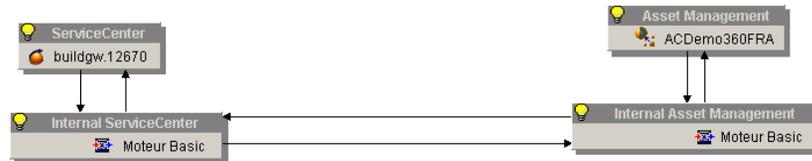
- 1 Lancez l'éditeur de scénarios
- 2 Glissez vos connecteurs source et destination dans le schéma du scénario
- 3 Configurez ces deux connecteurs en utilisant l'assistant **Configuration du connecteur**

L'option **Utiliser des types de document pivots** doit être sélectionnée dans la page **Utilisation des types de document pivots** de l'assistant **Configuration du connecteur**.

- 4 Reliez le connecteur source au connecteur destination en maintenant appuyée la touche **Maj.** de votre clavier
Si vous ne maintenez pas la touche **Maj.**, une boîte de mapping est créée automatiquement entre les deux connecteurs.
- 5 Sélectionnez le connecteur source
- 6 Créez un ou plusieurs types de document produits en vous basant sur les types de documents pivots (Exemple : **PrgnAsset**) disponibles dans la liste des types de documents publiés par le connecteur source.
- 7 Testez votre scénario en démarrant votre connecteur source
En démarrant votre connecteur source pour la première fois, les mappings entre les types de document publiés par les connecteurs et le type de document pivot sont générés automatiquement.
- 8 Passez en mode d'exploitation de votre scénario en associant une programmation et un service à votre scénario.

Afficher les connecteurs internes

Quand un connecteur utilise des types de document pivots, des connecteurs internes sont créés pour chaque connecteur du scénario. Pour afficher les connecteurs internes, choisissez **Montrer les connecteurs internes** dans le menu **Scénario**.

Figure 7.2. Affichage des connecteurs internes

L'affichage des connecteurs internes vous permet d'éditer les mappings créés automatiquement quand les types de documents pivots sont utilisés.

Onglets Mappings et Mappings pivots

La sélection d'un connecteur pivot dans le schéma du scénario fait apparaître deux onglets :

- l'onglet Mapping
- l'onglet Mapping pivot

L'onglet Mappings

Les mappings de cet onglet correspondent aux mappings créés automatiquement entre les types de document produits ou consommés par les connecteurs et le type de document pivot.

Ces mappings peuvent être consultés mais ne doivent pas être édités puisqu'ils sont régénérés à chaque fois que le scénario est lancé. Pour modifier ces mappings, vous devez modifier les mappings pivots sur lesquels ils se basent.

L'onglet Mappings pivots

Les mappings de cet onglet correspondent aux mappings entre les types de document publiés par les connecteurs et le type de document pivot. Ces mappings servent de modèle à tous les mappings de l'onglet **Mappings** créés automatiquement.

Liste des types de document pivots

Les types de documents pivots disponibles sont :

- prgnAsset
- prgnCompany
- prgnContract
- prgnCostCenter
- prgnDepartment
- prgnFeatures
- prgnLicense
- prgnLocation
- prgnModel
- prgnNetworkDevice
- prgnPC
- prgnPerson
- prgnPrinter
- prgnSoftInst
- prgnTelephony

Dans la liste de types de documents publiés par les connecteurs, ces types de documents sont symbolisés par une icône particulière (✳).

Modifier les mappings Type de document publié - type de document pivot

Pour modifier les mappings des types de documents publiés par vos connecteurs à un type de document pivot :

- 1 Créez un scénario utilisant les types de documents pivots.
- 2 Affichez les connecteurs pivots en choisissant **Montrer les connecteurs internes** dans le menu **Scénario**.
- 3 Sélectionnez un connecteur pivot.
- 4 Sélectionnez un mapping dans l'onglet **Mappings Pivots**.
- 5 Modifiez votre mapping.

Le détail des mappings entre les types de document publiés par un connecteur et le type de document pivot sont disponibles dans la sous-section **Mappings pivots** de chaque connecteur pour lesquels le type de document pivot est disponible.

 **Attention :**

Le fichier .piv de votre connecteur doit être disponible en lecture et écriture pour permettre la modification des mappings.

Changer de connecteur destination

Les types de document pivots vous permettent pour le même scénario de changer de connecteur destination et relancer la production des documents par le connecteur source.

Pour changer de connecteur destination :

- 1 Ouvrez un scénario pour lequel vous avez utilisé les types de document pivots
- 2 Placez votre nouveau connecteur destination dans le schéma du scénario
- 3 Configurez ce connecteur avec l'assistant **Configuration du connecteur**
- 4 Liez votre connecteur source à votre connecteur destination en maintenant la touche **Maj.** appuyée
- 5 Relancez votre programmation ou votre connecteur source manuellement

8 | Test et débogage d'un scénario d'intégration

CHAPITRE

Tester un scénario consiste à savoir si :

- le connecteur source produit convenablement les documents pour lesquels vous avez créé des types de document produits
- la boîte de mapping transforme convenablement ces documents
- le connecteur destination consomme convenablement ces documents une fois que la boîte de mapping les a transformés

 **Avertissement :**

Pour effectuer ces tests, il est conseillé d'utiliser des données de test (base de données ou fichiers de démonstration). Des scénarios non testés utilisés en mode d'exploitation pourraient gravement endommager vos véritables données.

Tester un type de document produit

Dans un scénario dont le connecteur produit différents types de documents, il est conseillé de tester chacun des types de documents produit.

Pour tester un type de document produit par un connecteur source :

- 1 Ouvrez le connecteur source
- 2 Sélectionnez l'onglet **Types de document**
- 3 Désélectionnez tous les types de document produits par le connecteur exception faite du type de document que vous voulez tester
Si vous voulez tester plusieurs types de document, il vous suffit de ne pas les désélectionner avant de passer à l'étape suivante.
- 4 Lancez la production du type de document en cliquant sur ► ou en sélectionnant **Produire** dans le menu **Composant**
- 5 Consultez le contenu du journal des documents pour prendre connaissance des problèmes survenus lors du traitement des documents produits par votre connecteur source

 **Note :**

Dans les options de l'éditeur de scénario (Edition\ Options), l'option **Nombre de documents à produire lors d'un test de production des documents** fixe le nombre de document produits lors d'un test.

Cette option est automatiquement prise en compte lorsque vous tester un type de document produit. En revanche, quand le scénario est exécuté en tant que service sous Windows, cette option est automatiquement désactivée.

Utiliser le journal des documents

Utiliser le journal des documents vous permet de prendre connaissance des problèmes survenus lors du traitement des documents produits par un connecteur source.

Pour traiter les problèmes de traitement, il est conseillé de configurer le journal des documents pour que seuls apparaissent les documents présentant des problèmes de traitement.

Pour plus d'informations sur l'utilisation du journal des documents, consultez le chapitre [Editeur de scénarios](#) [page 31], section [Journaux](#) [page 54], sous-section [Journal des documents](#) [page 56].

Utiliser les fichiers de cache

Les fichiers de cache utilisés par les connecteurs contiennent la description des types de documents qu'ils publient.

Exemple :

A l'ouverture d'un connecteur de type base de données, ce connecteur récupère la description de toutes les tables de la base de données indiquée lors de sa configuration. En utilisant un fichier de cache, le connecteur n'a plus besoin de récupérer cette description mais utilise la description enregistrée dans un fichier de cache créé lors de la dernière ouverture.

Cette fonctionnalité est très utile en phase de test et de débogage d'un scénario durant laquelle vous allez ouvrir et fermer de nombreuses fois les connecteurs de votre scénario.

L'utilisation de fichiers de cache est disponible pour les connecteurs suivants :

- Connecteur Action Request System
- Connecteur Asset Management
- Connecteur ServiceCenter
- Connecteur InfraTools Network Discovery
- Connecteur InfraTools Management
- Connecteur FacilityCenter
- Connecteur TeleCenter
- Connecteur InfraTools Management
- Connecteur LDAP
- Connecteur Base de données
- Connecteur Lotus Notes
- Tous les connecteurs inventaire

Pour qu'un connecteur utilise un fichier de cache et ne se connecte pas à son application externe :

- 1 Configurer votre connecteur en sélectionnant l'option **Utiliser un fichier de cache** dans la page **Configuration du cache** de l'assistant **Configuration du connecteur**
- 2 Ouvrez une première fois ce connecteur pour que le fichier de cache contienne la description des types de documents qu'ils publient

Note : lors de cette première ouverture, l'icône signalant l'utilisation du cache n'apparaît sur la boîte du connecteur.

3 Fermez et ouvrez ce connecteur

Cette fois-ci le connecteur ne se connecte pas à son application externe mais utilise le fichier de cache pour publier ses types de document.

Note :

Chaque fois que vous ouvrez un connecteur utilisant le contenu de son fichier de cache, une icône symbolisant une base de données se surimpose à l'icône de la lampe jaune.

Figure 8.1. Connecteur - Icône signalant l'utilisation d'un cache



Effacer le contenu des fichiers de cache

Pour effacer le contenu des fichiers de cache :

- 1 Ouvrez votre connecteur
- 2 Sélectionnez **Effacer les caches** dans le menu **Scénario**

Vous pouvez effacer le contenu du fichier de cache d'un seul connecteur en sélectionnant **Cache/Effacer le cache** dans le menu **Composant**.

Lors de la prochaine ouverture de vos connecteurs, ceux-ci se connectent à leur application externe pour synchroniser le contenu de leur fichier de cache avec le contenu de leur application externe.

Avertissement :

La synchronisation des fichiers de cache est possible si l'option **Travailler hors connexion** a pour valeur **Non** dans les options de l'éditeur de scénarios.

Synchroniser le contenu des fichiers de cache avec le contenu des applications externes

Pour synchroniser le contenu des fichiers de cache avec le contenu des applications externes :

- 1 Ouvrez les connecteurs de votre scénario
- 2 Sélectionnez **Synchroniser les caches** dans le menu **Scénario**
Vous pouvez synchroniser le contenu du fichier de cache d'un seul connecteur en sélection **Cache/Synchroniser le cache** dans le menu **Composant**.
- 3 Attendez que Connect-It récupère la description des types de documents publiés par le connecteur
Une fois terminée la phase de synchronisation, les types de documents publiés par le connecteur correspondent au contenu de l'application externe.

 **Avertissement :**

La synchronisation des fichiers de cache est possible si l'option **Travailler hors connexion** a pour valeur **Non** dans les options de l'éditeur de scénarios.

Travailler hors connexion

Travailler hors connexion vous permet de travailler sans que les connecteurs du scénario ne soient connectés à leur application externe.

Pour travailler hors connexion, cliquez sur  après avoir ouvert vos connecteurs une première fois.

Le mode hors connexion crée automatiquement un fichier de cache pour les connecteurs du scénario.

Quand vous fermez et ouvrez à nouveau vos connecteurs, alors que le mode hors connexion est toujours activé, ceux-ci utilisent le contenu de leur fichier de cache respectif pour publier leurs types de document.

Note :

Chaque fois que vous ouvrez un connecteur alors que vous travaillez hors connexion, une icône symbolisant une base de données se surimpose à l'icône de la lampe éteinte.

Figure 8.2. Connecteur - Icône signalant le travail hors connexion



IMPORTANT :

En mode hors connexion, aucune donnée ne peut être récupérée de la source. La fonction de prévisualisation des données source dans la fenêtre d'édition des types de document produit est inactive. Ce mode n'est à utiliser que pour définir un mapping.

Ouvrir tous les connecteurs du scénario courant au démarrage d'un connecteur

Cette option permet d'ouvrir automatiquement tous les connecteurs au démarrage d'un seul connecteur. Cette option est très utilisée pour les scénarios utilisant de nombreux connecteurs.

Mode test

- Si le connecteur destination supporte les transactions, les données sont alors insérées, mais un rollback (retour à l'état initial, avant l'insertion des données) est effectué sur la transaction. On peut ainsi tester la consommation des documents sans modifier les données dans l'application destination.
- Si le connecteur destination ne supporte pas les transactions, les données ne sont pas envoyées vers l'application destination, ce qui permet de tester la production et le mapping, mais pas la consommation.

Effectuer une pause

Si de nombreux documents sont traités dans un scénario, une pause peut être effectuée pour consulter le journal de Connect-It et vérifier la présence d'éventuelles erreurs sans attendre la fin des traitements de toutes les données de la source.

Conseils lors de la définition des mappings

Il est parfois difficile de repérer la localisation (mapping/élément concerné) d'une erreur syntaxique dans un script Basic. Il est donc conseillé d'écrire et de tester les scripts un par un, plutôt que de les écrire tous en bloc et de les tester tous ensemble.

9 | Documentations des scénarios

CHAPITRE

L'éditeur de scénarios permet de voir ou créer une documentation contenant de multiples informations sur le scénario courant.

Les informations de la documentation d'un scénario sont toujours à jour puisqu'elles sont extraites dynamiquement du fichier SCN du scénario.

Contenu d'une documentation de scénario

Une documentation de scénario contient les sections suivantes :

- Informations générales

Cette section fournit les informations suivantes sur le scénario :

- Nom
Le nom du scénario est précédé de son chemin complet sur votre ordinateur.
- Date de dernière modification
- Version et numéro de Build de Connect-It avec lequel le scénario a été modifié pour la dernière fois.

- Toutes les informations saisies par l'utilisateur dans le dernier historique du scénario. (Menu **Fichier/Historique du scénario**).
- Informations sur les schémas du scénario
Cette section fournit les informations suivantes sur le scénario :
 - Informations sur les vues
Les mappings associés à chaque vue sont listés.
 - Schémas associés aux vues
Un image au format **.bmp** reprend l'aperçu des vues du scénario.
- Configuration des connecteurs
Cette section fournit les paramètres de configuration des connecteurs du scénario. Pour plus d'informations sur la configuration des connecteurs, consultez le manuel **Connecteurs**, chapitre **Configuration des connecteurs**.
- Liste des clauses 'where' et 'order by' par type de document
Cette section reprend la liste des clauses WHERE et ORDER BY par type de document. Ces clauses font partie des directives de production des connecteurs source du scénario.
- Types de document pivot
Cette section présente les éléments de chaque type de document pivot du scénario.
Pour plus d'informations sur les types de document pivots, consultez le chapitre [Types de document pivots](#) [page 157].
Pour chaque type de document pivot, les informations suivantes sont disponibles :
 - Nom du type de document pivot
 - Nom de chaque élément
 - Nature de l'élément :
 - champ (ATTRIBUTE)
 - structure (STRUCT)
 - collection (ARRAY)
 - Le type de donnée pour les champs : String, short, double, timestamp, etc.

 **Note :**

Si le scénario n'utilise pas les types de document pivots, cette section n'apparaît pas.

- **Types de document définis par l'utilisateur**

Cette section présente les éléments de chaque type de document défini par l'utilisateur.

Pour chaque type de document, les informations suivantes sont disponibles :

- Nom du type de document
- Nom de chaque élément
- Nature de l'élément :
 - champ (ATTRIBUTE)
 - structure (STRUCT)
 - collection (ARRAY)
- Le type de donnée de chaque champ : string, short, double, timestamp, etc.

Pour plus d'informations sur la création des types de documents, consultez le chapitre [Mise en place d'un scénario d'intégration](#) [page 79], section [Editer les types de document produits ou consommés](#) [page 84] .

- **Mappings pivots**

Cette section détaille les mappings pivots du scénario.

 **Note :**

Si aucun mapping pivot n'a été défini dans le scénario, cette section n'apparaît pas.

- **Mappings définis par l'utilisateur**

Cette section détaille les mappings entre les types de document définis par l'utilisateur.

Pour chaque mapping, les informations suivantes sont disponibles :

- Le nom du mapping
 - Après le nom du mapping les noms des types de document source et destination publiés par les connecteurs sont indiqués entre parenthèses.
- Le nom du type de document source

- Le nom du type de document destination
- Pour chaque noeud du mapping :
 - l'élément destination
 - le script de mapping
 - l'élément source
 - la description éventuelle du mapping

Pour plus d'informations sur la création des mappings , consultez le chapitre [Mappings des types de document](#) [page 97].

- Programmeurs et fichiers associés

Cette section présente :

- Les programmeurs du scénario
 - Pour chaque programmeur sont détaillés :
 - Son nom
 - Sa périodicité
 - Les connecteurs et les types de documents associés
- Les fichiers de tables de correspondance
- Les fichiers de script Basic
- Les fichiers de chaînes de caractères

- Historique des modifications du scénario

Cette section fournit pour chaque scénario des informations sur chacune de ses modifications. Ces informations correspondent à celle que l'auteur du scénario saisit dans la fenêtre **Historique du scénario**. (Menu **Fichier/Historique du scénario**).

Visualisation de la documentation d'un scénario au format HTM

L'éditeur de scénarios vous permet d'afficher dans votre navigateur internet la documentation du scénario courant au format HTM.

Pour voir la documentation d'un scénario au format HTM

- 1 Ouvrez ou créez un scénario avec l'éditeur de scénarios.
- 2 Sélectionnez **Fichier/Voir la documentation du scénario**.
- 3 Attendez que votre navigateur internet se lance et affiche la documentation du scénario.

 **Note :**

Pour utiliser la fonction **Voir la documentation du scénario**, vérifiez que votre système d'exploitation associe l'extension de nom de fichier HTM à un navigateur internet. Exemples : Internet Explorer de Microsoft ou Netscape Navigator.

Propriétés des documentations de scénario

La création des documentations de scénario utilise les éléments suivants :

- un processeur XSL
Le processeur fourni avec Connect-It est le processeur Xalan.
- Une feuille de style CSS utilisée pour la visualisation au format HTML
- Des feuilles de style XSL permettant la conversion des fichiers SCN au format DBK
 - fichiers SCN au format DBK
 - fichiers DBK au format HTML

La conversion d'un fichier SCN au format DBK passe par une conversion intermédiaire du fichier SCN au format XML.

La conversion d'un fichier SCN au format HTML passe par les conversions intermédiaires au format XML puis au format HTM.

Format de la documentation de scénario	Ordre de conversion
---	----------------------------

DBK	1 SCN
	2 XML
	3 DBK

Format de la documentation de scénario	Ordre de conversion
HTM	1 SCN
	2 XML
	3 DBK
	4 HTM

Editer les propriétés des documentations de scénario

L'édition des documentation de scénario s'effectue à l'aide d'un assistant que vous lancez en sélectionnant **Fichier/Propriétés des documentations**.

Configurer le processeur XSL

Cette page de l'assistant vous permet de configurer le processeur XSL utilisé pour convertir les fichiers SCN au format DBK et HTM.

Le processeur xalan fourni avec Connect-It est le processeur par défaut.

Le chemin de ce fichier est le suivant : **[Dossier d'installation de Connect-It]\datakit\doctrans\xalan\xalan.bat**.

Vous pouvez utiliser d'autres processeurs XSL. Exemple : le processeur MSXSL téléchargeable gratuitement sur différents sites Microsoft.

Fichier exécutable du processeur XSL

Indiquez dans ce champ le chemin complet d'un processeur XSL. Par défaut, le chemin de ce fichier est : **[Dossier d'installation de Connect-It]/datakit/doctrans/xalan/xalan.bat**.

Paramètres du processeur XSL

Les paramètres par défaut de ce champ sont :

- %IN
Cette variable correspond au chemin complet du fichier SCN à convertir.
- %XSL
Cette variable correspond au chemin complet du fichier XSL utilisé pour la conversion au format DBK ou HTM.
- %OUT

Cette variable correspond au chemin complet du fichier DBK ou HTM de sortie.

Permettre l'utilisation d'une feuille de style CSS lors d'une documentation de scénario HTML

Sélectionnez cette option pour que la visualisation de la documentation de scénario au format HTM utilise une feuille de style CSS.

Le chemin de la feuille de style CSS utilisé par défaut est **[Dossier d'installation de Connect-It]\datakit\doctrans\css\default.css**

Feuilles de style XSL prédéfinies

Cette page vous indique les feuilles de style utilisées pour la conversion :

- du fichier XML au format DBK
- du fichier DBK au format HTM

Vous pouvez changer la valeur des colonnes **Description** et **Alias ou Feuille de style XSL** en cliquant directement sur ces valeurs.

L'alias d'une feuille de style est la valeur saisie dans la colonne **Extension**. En indiquant un alias avant le chemin complet d'une autre feuille de style séparés par un point virgule, vous indiquez à votre processeur XSL une première feuille de style à utiliser.

Pour l'extension HTM, la valeur **DBK;C:\Program**

Files\Peregrine\ConnectIt\datakit\doctrans\xs\dbk2htm.xsl indique que le processeur XSL convertit le format XML du scénario au format DBK puis au format HTM.

Feuilles de style XSL définies par l'utilisateur

Cette page vous permet d'indiquer de nouvelles feuilles de style XSL vous permettant de convertir un fichier d'un format à un autre.

Pour indiquer une feuille de style XSL définie par l'utilisateur

- 1 Cliquez sur .
- 2 Saisissez un nom d'extension de fichier dans la colonne **Extension**.
- 3 Cliquez sur la même ligne dans la colonne **Description** et saisissez une description.

- 4 Cliquez sur la même ligne dans la colonne **Alias ou Feuille de style XSL** et saisissez le chemin de votre feuille de style XSL.

En indiquant les alias d'autres feuilles de style XSL, vous pouvez indiquer autant de conversions intermédiaires entre un format de départ et un format de sortie final.

Création d'une documentation de scénario

L'éditeur de scénarios permet de créer des documentation aux formats suivants :

- Format HTM
- Format DBK

Pour créer une documentation de scénario

- 1 Ouvrez ou créez un scénario avec l'éditeur de scénarios.
- 2 Sélectionnez **Fichier/Créer une documentation**.
- 3 Nommez le fichier de documentation.
- 4 Sélectionnez le format du descriptif dans le champ **Type**.
- 5 Cliquez sur **Enregistrer**.



Note :

Par défaut, la documentation du scénario est enregistré dans le même dossier que le fichier SCN de votre scénario.

Documentation au format DBK

Créer une documentation au format DBK crée un fichier XML conforme à la DTD DocBook.

Pour plus d'informations sur la DTD DocBook, consultez le site suivant : <http://www.docbook.org>.

10 | Exploitation d'un scénario d'intégration

CHAPITRE

L'exploitation d'un scénario vous permet d'automatiser le processus d'intégration entre les applications externes de votre scénario grâce à :

- la création d'une programmation déterminant quand les connecteurs source de votre scénario traitent les données
- la création, sous Windows 32 bit, d'un service associé à ce scénario vous permettant de l'utiliser en tâche de fond sur votre serveur Connect-It
- la gestion des problèmes de traitement par l'intermédiaire de la console Connect-It

Créer une programmation

La programmation permet de fixer les moments de réveil des connecteurs qui produisent des documents. Sans cette programmation, un scénario est incomplet.

Choisir une programmation revient à associer des types de document produits par un connecteur à des programmeurs.

Un programmeur joue le rôle d'une minuterie qui réveille un connecteur :

- à intervalle régulier à l'intérieur et à l'extérieur d'une plage horaire définie

- de façon ponctuelle (pour le 6 mars 2003 par exemple).

Des règles peuvent modifier le comportement d'un programmeur. Exemple : un programmeur réveille un connecteur tous les jours. Une règle peut s'appliquer changeant son comportement pour un jour donné.

Exemple :

Le connecteur Desktop Discovery du scénario Desktop Discovery vers Asset Management produit des documents de type **Machine** chaque jour entre 9 heures du matin et 10 heures du soir toutes les cinq minutes. En dehors de cette plage horaire, le connecteur Desktop Discovery produit des documents toutes les heures. Vous introduisez une règle changeant les paramètres précédents pour les jours de votre choix.

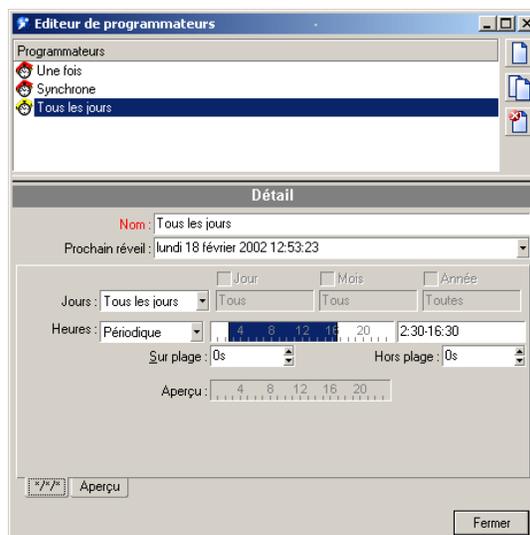
Edition d'un programmeur

Vous accédez à l'éditeur des programmeurs à l'aide du menu **Scénario/Programmeurs**.

Deux programmeurs non modifiables sont fournis :

- Le programmeur **Une fois**
Ce programmeur permet de lancer la production de document une fois. Quand un type de document produit est créé, il est associé par défaut à ce connecteur.
- Le programmeur **Asynchrone**
Ce programmeur permet de lancer la production de documents toutes les secondes sur la plage horaire 00h00-24h00.

Figure 10.1. L'éditeur de programmeurs



Icône



Fonction

Lance la création d'un nouveau programmeur



Duplique le programmeur sélectionné



Supprime le programmeur sélectionné

Champ Jours

Ce champ fixe les jours durant lesquels un type de document est produit.

Pour ce champ, vous disposez des options suivantes :

- **Tous les jours**

Tous les jours de l'année, sans exception.

- **Jour de l'année**

Un jour ou une sélection de jours, définie au moyen des cases à cocher Jour, Mois et Année. Exemple : 6 janvier 2002.

Premier, second, troisième, quatrième, avant-dernier, dernier jour

Vous définissez un jour de la semaine au moyen de la case à cocher Jour, et pour le(s) mois et année(s) définis au moyen des cases Mois et Année.

Exemple : le premier vendredi de chaque mois.

Champ Heures

Pour ce champ vous avez deux options :

- Périodique

En choisissant l'option périodique, vous fixez une plage pour une journée pour laquelle vous définissez une période dans le champ **Sur plage**.

Exemple : de 4 heures du matin à 10 heures du soir, toutes les cinq minutes.

Vous pouvez également définir une période en dehors de cette plage dans le champ **Hors plage**.

Vous saisissez une valeur dans le champ texte. Cette saisie manuelle doit respecter la syntaxe suivante :

```
<heure de début> -
<heure de fin>, <heure de début> - <heure de fin>...
```

Le format de saisie des heures dépend des options régionales de votre poste accessibles, par exemple, via le panneau de configuration Windows. Si le paramètre optionnel [AM|PM] n'est pas précisé, l'heure est considérée comme saisie au format 24 heures. Exemple : si vous entrez la valeur "18", la valeur 6:00 PM est affichée automatiquement après validation.

Note :

La saisie graphique ne permet l'édition qu'avec une précision d'une demi-heure. La saisie manuelle permet d'affiner les plages horaires à la minute près.

- Enumération

Dans ce cas, vous saisissez les heures de votre choix en les séparant par un point virgule. Cela vous permet de réveiller vos connecteurs de manière ponctuelle :

```
5:00AM;8:00PM;...
```

Créer un programmeur (exemple)

Ce paragraphe décrit comment créer un programmeur pour un réveil :

- tous les jours de la semaine
- toutes les dix minutes entre 8 heures du matin et 10 heures du soir et toutes les heures en dehors de cette plage horaire.

Vous lui appliquez une règle déclenchant la production de documents :

- tous les 1er dimanche du mois
- toutes les 20 minutes entre 6 heures du matin et 10 heures du soir et toutes les demi-heures en dehors de cette plage horaire

Pour créer un programmeur, cliquez sur .

Renseignez les champs de l'onglet qui apparaît sous votre volet de détail du programmeur.

Tous les jours de la semaine

Sélectionnez l'option **Tous les jours** dans le champ **Jours**

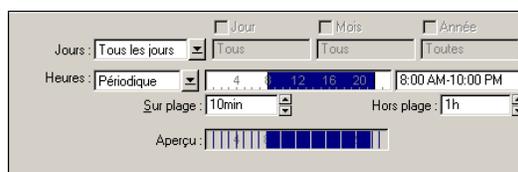
Toutes les 10 minutes entre 8 heures du matin et 10 heures du soir et toutes les heures en dehors de cette plage horaire

Sélectionnez l'option **Périodique** dans le champ **Heures**.

Pour sélectionner la plage horaire 8 heures du matin - 10 heures du soir, vous pouvez :

- Soit la saisir directement dans le champ texte (8:00 AM-10:00 PM ou 8-22)
- Soit la saisir dans la représentation graphique d'une journée de 24 heures.

Ensuite, renseignez les champs **Sur plage** et **Hors plage** en entrant les valeurs "10 min" et "1 h". Ces deux champs règlent la fréquence de réveil, respectivement dans la plage sélectionnée et en dehors de la plage sélectionnée. Si vous avez suivi les instructions précédentes, votre onglet doit ressembler à la capture d'écran suivante :



Cet onglet est nommé automatiquement en fonction des données saisies dans le champ **Jours**.

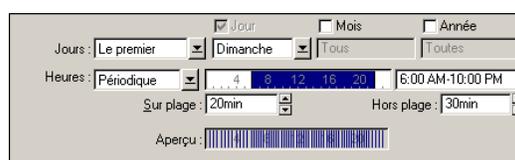
Validez la création de votre programmeur en cliquant sur le bouton **Créer**.

Créer une règle

Pour créer une règle, cliquez droit sur un des onglets de l'éditeur de programmeurs. Dans le menu contextuel, sélectionnez l'option **Ajouter une règle**.

Un nouvel onglet apparaît que vous remplissez en fonction de la règle à appliquer à ce programmeur.

Dans notre exemple, l'onglet de cette règle doit ressembler à la capture d'écran suivante :



Vous validez la création de votre règle en cliquant sur le bouton **Créer**.

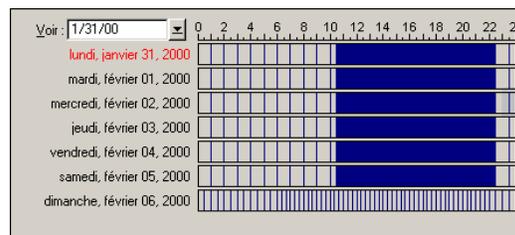
Supprimer une règle

Pour supprimer une règle, cliquez droit dans l'onglet de cette règle. Dans le menu contextuel, sélectionnez l'option **Supprimer une règle**.

Aperçu

En sélectionnant l'onglet **Aperçu**, vous obtenez le comportement du programmeur durant la semaine courante.

A chaque sélection d'un jour dans le champ **Voir**, le calendrier propose le comportement du programmeur la semaine suivant ce jour.



Modifier un programmeur

Pour modifier un programmeur :

- 1 Sélectionnez un programmeur dans la liste
- 2 Modifiez les paramètres de votre choix dans les onglets
- 3 Validez vos modifications en cliquant sur le bouton **Modifier**

Supprimer un programmeur

Pour supprimer un programmeur :

- Sélectionnez un programmeur dans la liste
- Appuyez sur la touche **Supprimer** ou cliquez sur .

Edition de la programmation

Pour éditer la programmation, utilisez le menu **Scénario/Programmation**.

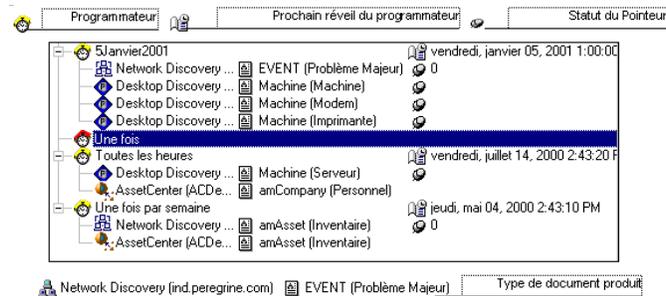
La programmation par défaut de votre scénario associe vos types de document produits au programmeur **Une fois**.

Pour associer vos types de document produits à un programmeur :

- cliquez sur un type de document produit et glissez-le sous le programmeur de votre choix
ou bien
- sélectionnez un type de document produit et utilisez les flèches situées à droite de votre fenêtre d'édition pour le déplacer.

Vos programmeurs apparaissent dans la fenêtre d'édition, classés par ordre alphabétique. Vous les associez aux types de document produits présents dans votre scénario.

Figure 10.2. Fenêtre d'édition de la programmation



Classement des types de document produits dans la programmation

Déplacement d'un type de document produit vers un autre programmeur

▲	Déplacement vers le programmeur précédent
▼	Déplacement vers le programmeur suivant

Déplacement d'un type de document dans la liste au sein du même programmeur

▲	Avancer d'un rang
▼	Reculer d'un rang

Le traitement de vos données dépend de deux classements :

- Le classement des types de document produits dans la programmation
- Le classement des mappings d'un même type de document produit

L'ordre dans lequel vos types de document sont classés détermine leur ordre de production.

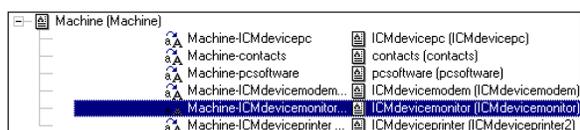
Le classement des mappings dans l'onglet **Mappings** de la fenêtre principale de Connect-It est également important. Cet ordre indique à Connect-It quel mapping effectuer en premier dans le cas où il existe plusieurs mappings pour un même type de document source.

Pour classer vos mappings :

- Sélectionnez une boîte de mapping dans le volet du schéma du scénario
- Sélectionnez l'onglet **Mappings**
- Sélectionnez un type de document produit dans la liste
- Déplacez-le dans la liste en cliquant sur les flèches

Exemple : dans le scénario Desktop Discovery vers ServiceCenter, il existe six mappings pour le même type de document **Machine** [Machine].

Figure 10.3. Classement des mappings



Icône	Fonction
	Lance la création d'un mapping.
	Permet d'éditer le mapping sélectionné.
	Supprime le mapping sélectionné.
	Avance le mapping sélectionné d'un rang
	Fait reculer le mapping sélectionné d'un rang

Quand un scénario est lancé, Connect-It prend en compte pour le même programmeur :

- En premier l'ordre des types de document produits
- En second, pour un même type de document produit, l'ordre de ses mappings.

Statut des pointeurs

Dans la plupart des cas, le statut d'un pointeur concerne une heure spécifiée. Cette information, fournie par le pointeur, est essentielle. Elle permet de situer l'avancement du traitement des données de vos applications externes.

Le type de ces pointeurs diffère en fonction de l'application externe concernée :

- Connecteur Asset Management

Pour chaque mapping impliqué par votre scénario, le pointeur correspond à la date de dernier lancement de votre mapping. Tous les enregistrements dont la date de dernière modification (Champ **dtLastModif**) est postérieure au pointeur sont traités.

Note :

Le décalage entre le poste sur lequel est installé Connect-It et le serveur AssetCenter est automatiquement compensé.

- Connecteur Sécurité NT

A chaque fois que le programmeur se réveille, le connecteur récupère l'ensemble des informations du domaine à explorer et non juste les informations complémentaires.

 **Note :**

Aucun pointeur de programmation n'apparaît lorsque vous créez un nouveau programmeur et lui affectez un domaine NT

- Connecteur Desktop Discovery

Le pointeur utilisé est la date de création la plus récente des fichiers **.fsf** déjà traités.

- Connecteur Network Discovery

Le pointeur utilisé est le numéro d'événement (pour le type de document de type EVENT).

- Connecteur ServiceCenter

La signification du pointeur de ce connecteur diffère en fonction de la nature des données provenant de l'application source :

- Pour des événements, le pointeur correspond à leur numéro de séquence (**evsysseq**).

Pour chaque mapping impliqué par votre scénario, le pointeur correspond au numéro de séquence du dernier événement traité lors du dernier lancement de votre mapping. Au lancement suivant, tous les événements ayant un numéro de séquence supérieur à cette valeur sont traités.

- Pour des tables, le pointeur correspond à la date de dernier lancement de votre mapping. Au lancement suivant, tous les enregistrements dont la date de dernière modification (Champ **sysmodtime**) est postérieure au pointeur sont traités.
-

 **Note :**

En fonction de la configuration de votre connecteur ServiceCenter, le décalage entre le poste sur lequel est installé Connect-It et le serveur ServiceCenter est :

- soit compensé automatiquement
 - soit compensé en fonction du décalage saisi par l'utilisateur
-

En double-cliquant sur le statut de chaque pointeur, vous pouvez forcer une programmation à prendre en compte un fichier **.fsf** ou un numéro d'événement généré par Network Discovery.

Exemple :

Plusieurs dizaines de milliers d'événements sont mémorisés par Network Discovery. En remplaçant le statut du pointeur (ayant la valeur 0 avant le premier lancement d'une programmation) par le numéro 20 000, seuls les événements portant un numéro supérieur à 20 000 sont traités par le scénario.

Pointeur des fichiers **.fsf**

Exemple :

Lors de la première mise en route de votre connecteur InfraTools Desktop Discovery vous disposez de deux fichiers :

- le fichier **paris.fsf** daté du 1er janvier
- le fichier **rome.fsf** daté du 15 janvier

Lors de la mise en route suivante du connecteur, la date du 15 janvier sert de date repère (pointeur). Ainsi tout fichier dont la date de création est antérieure à cette date **n'est pas traité** par Connect-It. La seule façon de forcer le traitement de ce fichier **.fsf** dans votre scénario est de changer manuellement l'état de votre pointeur dans la fenêtre d'édition de la programmation.

En mode d'exploitation de votre scénario, si l'option **Le laisser dans le dossier** est sélectionnée dans la configuration de votre connecteur InfraTools Desktop Discovery, la date du fichier le plus récent sert de repère au connecteur InfraTools Desktop Discovery.

Rafraîchissement des pointeurs dans la fenêtre d'édition de la programmation

Si votre fenêtre d'édition de la programmation est ouvert alors qu'un scénario est lancé, l'état des pointeurs n'est pas rafraîchi. Pour que l'état des pointeurs soit rafraîchi dans la fenêtre d'édition de la programmation, vous devez fermer cette fenêtre et l'ouvrir de nouveau.

Lancement de la programmation

Lancer une programmation revient à démarrer un scénario. Pour démarrer un scénario :

- sélectionnez l'option **Démarrer tous les programmeurs** dans le menu **Scénario**
ou
- cliquez directement sur l'icône .

Arrêt de la programmation

Stopper une programmation revient à arrêter un scénario. Pour arrêter un scénario :

- sélectionnez l'option **Arrêt** dans le menu **Scénario** ou
- cliquez directement sur l'icône .

Créer un service Connect-It

Connect-It vous permet d'associer un service sous Windows à un scénario. Ce service permet à votre serveur Connect-It de lancer le traitement des données en tâche de fond en fonction de la programmation associée à votre scénario.

Avertissement :

Si l'un des connecteurs de votre scénario utilise une connexion ODBC, celle-ci doit absolument utiliser une source de données système (DNS système). Si cette connexion utilise une source de données utilisateur (DNS utilisateur), elle ne sera pas visible par le service auquel le scénario est associé

Un service sous Windows permet de disposer d'une application qui tourne en tâche de fond. Connect-It vous permet de créer; et lancer autant de services que vous disposez de scénarios.

Note :

Sous Windows 32 bit, si vous avez modifié une variable d'environnement (par exemple, le chemin de recherche), cette modification n'est propagée au service qu'après avoir redémarré l'ordinateur.

IMPORTANT :

Si un connecteur spécifie un répertoire ou un fichier sur un autre ordinateur, il ne doit pas être indiqué par un disque mappé par l'utilisateur mais doit référencer l'ordinateur distant. Par exemple :

\\FSFStore\Scan

et non pas :

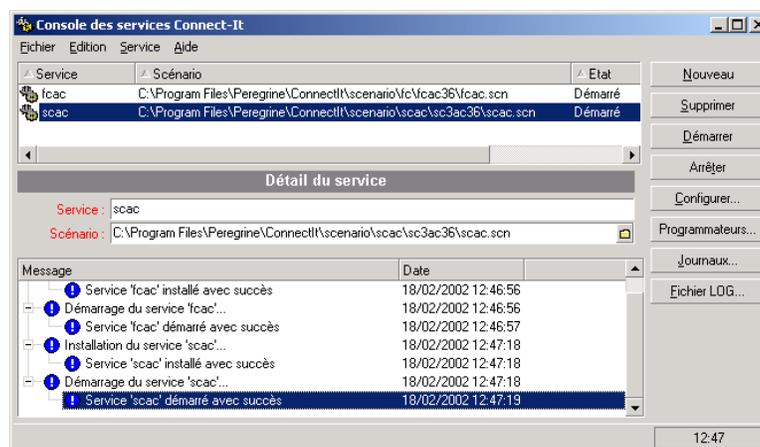
Z:\Scan

si le disque z est mappé sur \\FSFStore.

La console des services

La console des services est un composant de l'application Connect-It. C'est une interface graphique vous permettant de gérer vos scénarios et de créer des services que vous leur associez.

Figure 10.4. Console des services



Pour lancer la console, choisissez **Console des services** dans le groupe de programme Peregine\ Connect-It de votre barre de démarrage. Vous pouvez également lancer l'exécutable **console.exe** présent dans le dossier **bin32** de votre dossier d'installation de Connect-It.

Menus

La console des services dispose de quatre menus.

Menu Fichier

Commande	Fonction
Quitter	Permet de quitter la console Connect-It

Menu Edition

Commande	Fonction
Couper, copier, coller	Permettent de disposer des fonctions d'édition courantes pour la console.

Menu Service

Commande	Fonction
Démarrer	Permet de démarrer un service Connect-It
Arrêter	Permet d'arrêter un service Connect-It
Configurer	Permet de configurer les connecteurs d'un scénario en lançant l'assistant Configuration du connecteur
Programmateurs	Permet de modifier les programmeurs utilisés dans le scénario associé au service
Journaux	Permet de lancer l'éditeur de scénarios en affichant les onglets Journal de Connect-It et Journal des documents
Fichier LOG	Permet de voir le fichier LOG accompagnant chaque service Connect-It créé

Menu Aide

Commande	Fonction
A propos...	Permet d'afficher la boîte A propos

Créer un service Connect-It sous Windows

Pour créer un service Connect-It

- 1 Lancez la console des services.
- 2 Cliquez sur **Nouveau**.
- 3 Nommez votre service en renseignant le champ **Service**.
- 4 Indiquez le chemin et le nom de votre scénario en cliquant sur .
- 5 Cliquez sur **Créer**.

Le service créé apparaît dans le volet supérieur de la console.

Propriétés de connexion du service

Dans certains cas, vous devez modifier les propriétés de connexion du service sous Windows pour qu'il puisse fonctionner convenablement.

Exemples

- le connecteur source du scénario lié au service lit les fichiers d'un dossier se trouvant sur un lecteur réseau auquel le compte local du service n'a pas accès.
- le connecteur e-mail d'un scénario ne peut fonctionner qu'avec un compte particulier.

Dans ces deux cas, vous devez spécifier un compte particulier dans les propriétés de connexion du service.

Pour modifier les propriétés de connexion d'un service sous Windows XP

- 1 Démarrez Windows XP.
- 2 Choisissez **Démarrer/ Paramètres/ Panneau de configuration/ Outils d'administration/Services**.
- 3 Double-cliquez sur le service dans la liste.
- 4 Sélectionnez l'onglet **Connexion** dans la boîte de dialogue qui s'affiche.
- 5 Sélectionnez l'option **Ce compte**.
- 6 Saisissez les valeurs vous permettant d'utiliser convenablement le scénario lié à votre service dans les champs **Ce compte** et **Mot de passe**.
- 7 Cliquez sur **Appliquer**.

Chemin du scénario

En indiquant le chemin d'un scénario dans le champ **Scénario**, vous devez respecter les règles suivantes :

- 1 Le scénario est enregistré sur le serveur Connect-It

Si le scénario est situé dans un dossier associé à un lecteur réseau, n'indiquez pas la lettre de ce lecteur réseau dans le champ **Scénario**.

Exemple : le scénario **scenario\myscenario.scn** se situe dans un dossier **C:\Program Files\ConnectIt** associé au lecteur réseau **R:**. Dans ce cas, saisissez la valeur **C:\Program Files\ConnectIt\scenario\myscenario.scn** dans le champ **Scénario** et non pas la valeur **R:\scenario\myscenario.scn**. En effet, la plupart du temps, le service utilise le compte système local ou un compte spécifique pour lequel l'association du dossier **C:\Program Files\ConnectIt** au lecteur réseau **R:** n'existe pas.

- 2 Le scénario est installé sur le réseau

Vérifiez que le service peut utiliser le chemin réseau indiqué dans le champ **Scénario**.

Pour que le service puisse utiliser un chemin réseau, associez ce service à un nom d'utilisateur et à un mot de passe pouvant utiliser ce chemin réseau.

Créer un service Connect-It sous UNIX

Pour créer un service (démon) Connect-It sous UNIX :

- 1 Editez le fichier **conitsvc.ini** situé à la racine du client.
- 2 Créez une section [Service] à laquelle vous ajoutez la ligne suivante :

```
[Nom du service]=[chemin complet du scénario]
```

```
[Service]
indsc=/export/home/jean/test/ConnectIt/scenario/ind/indsc.scn
```

Exemple :

Pour lancer ce démon UNIX, exécutez la ligne de commande suivante :

```
conitsvc -svc indsc &
```

Démarrer ou arrêter un service Connect-It

Pour démarrer un service Connect-It :

- Lancez la console Connect-It

- Sélectionnez le service dans le volet principal de Connect-It
 - Cliquez sur **Démarrer**
-

 **Note :**

Pour être démarré, un scénario doit avoir été préalablement configuré.

Pour arrêter un service Connect-It :

- lancez la console Connect-It
- Sélectionnez le service dans le volet principal de Connect-It
- Cliquez sur **Arrêter**

Supprimer un service Connect-It

Pour supprimer un service Connect-It :

- Lancez la console Connect-It
- Sélectionnez le service dans le volet principal de Connect-It
- Cliquez sur **Supprimer**

Ligne de commande

Pour lancer un service en ligne de commande :

- 1 Placez-vous dans le dossier **bin32** du dossier d'installation de Connect-It

Exemple : C:\Peregrine\ConnectIt\bin32\

- 2 Exécutez la ligne suivante dans une invite de commande :

```
NetStart <nom du service>
```

Exemple :

Si vous lancez le service **Asset Management-ServiceCenter**, votre ligne de commande est :

```
NetStart Asset Management-ServiceCenter
```

Gérer le suivi d'un scénario en utilisant la console des services

Pour gérer le suivi d'un scénario, utilisez la console des services pour :

- modifier la configuration de vos connecteurs
- éditer les programmeurs de votre scénario
- consulter les journaux de Connect-It
- consulter un fichier LOG associé à votre service

Modifier la configuration de vos connecteurs

Pour modifier la configuration des connecteurs d'un scénario pour lequel vous avez créé un service :

- 1 Sélectionnez le service correspondant à votre scénario
- 2 Cliquez sur **Configurer**
- 3 Renseignez les pages des différents assistants **Configuration du connecteur** qui s'affichent les uns après les autres

Une fois terminée la configuration du dernier connecteur, l'éditeur de scénarios se ferme automatiquement.

Editer les programmeurs

Pour éditer les programmeurs d'un scénario pour lequel vous avez créé un service :

- 1 Sélectionnez le service correspondant à votre scénario.
- 2 Cliquez sur **Programmeurs**.
- 3 Attendez que Connect-It se lance et que la fenêtre Editeur de programmeurs s'affiche.

Pour plus d'informations sur l'édition des connecteurs, consultez dans ce chapitre, la section [Créer une programmation](#) [page 179].

Consulter les journaux de Connect-It

Deux journaux sont consultables dans l'éditeur de scénarios :

- le journal de Connect-It

Ce journal décrit toutes les actions réalisées dans Connect-It au lancement d'un scénario. Exemple : la désérialisation du scénario, la connexion des connecteurs à leurs applications externes respectives, etc.

- le journal des documents

Ce journal vous permet de voir le détail les documents produits et consommés par les composants de votre scénario. Il permet également d'identifier la cause des problèmes pouvant survenir dans le traitement de certains documents.

Pour voir s'afficher les onglets correspondant à ces journaux dans l'éditeur de scénario :

- 1 Sélectionnez le service correspondant à votre scénario
- 2 Cliquez sur **Journaux**

L'éditeur de scénarios se lance avec le schéma de votre scénario et les onglets correspondant aux journaux de Connect-It.

Consulter le fichier LOG du service

A chaque fois qu'un service Connect-It est démarré, un fichier LOG est disponible.

Ce fichier porte le nom de votre service (Exemple : **test.log**). Pour consulter ce fichier LOG :

- 1 Sélectionnez le service correspondant à votre scénario
- 2 Cliquez sur **Fichier LOG**

Le fichier LOG de votre scénario s'ouvre dans l'application associée aux fichiers **.log** sous votre système d'exploitation.

Optimiser les performances d'un scénario

Cette section vous explique comment optimiser les performances d'un scénario en diminuant le délai de traitement des documents par les différents composants du scénario.

Evaluation des durées de traitement des documents à l'aide des statistiques

Les journaux de Connect-It consultables dans l'interface graphique de l'éditeur de scénario ou dans les fichiers journaux (log) vous permettent d'obtenir des statistiques sur :

- les délais de production des documents par les connecteurs source
- les délais de transformation des documents par les boîtes de mapping
- les délais de consommation des documents par les connecteurs destinations
- le nombre de documents rejetés

Exemple : les statistiques du connecteur Asset Management vous informe que les documents correspondant aux enregistrements de la table des produits n'ont pas pu être traités.

Pour obtenir des statistiques du traitement de données par un composant

- 1 Ouvrez un scénario.
- 2 Ouvrez les connecteurs du scénario (**Ctrl + F4**).
- 3 Sélectionnez le connecteur source du scénario.

Exemple : le connecteur InfraTools Desktop Discovery du scénario `idd\iddac36\iddac.scn`.

- 4 Cliquez sur **▶ (F5)**.
- 5 Positionnez le pointeur de la souris sur un connecteur source, un connecteur destination ou une boîte de mapping.

Une fenêtre contextuelle s'affiche comprenant une rubrique **Statistiques**.

Note :

Ces données sont également disponibles dans l'onglet Journal de Connect-It s'affichant sous le schéma du scénario.

Analyse des statistiques

Les données suivantes, affichées dans le journal de Connect-It, représentent les statistiques d'un scénario comprenant un connecteur Action Request System source, une boîte de mapping et un connecteur LDAP destination :

```

Statistiques concernant le connecteur 'Action Request System (fdcitsrv01)
' (session: 26min 57.310s / API: 26min 04.550s)
Document(s) consommé(s) : 7143
Document(s) rejeté(s) : 1
Enregistrement(s) inséré(s) : 1199
Enregistrement(s) mis à jour : 5943
Statistiques concernant le connecteur 'LDAP (mail-sd.peregrine.com)' (se
ssion: 47.628s / API: 35.765s)
Document(s) produit(s) : 7143
Statistiques concernant le connecteur 'Mapping (Basic engine)' (Session:
01.404s)
Script(s) analysé(s) : 14286
Document(s) consommé(s): 7143
Document(s) produit(s): 7143

```

L'analyse des statistiques produites est la suivante :

- Traitement des documents par le connecteur Action Request System = 27 minutes soit **4.4** documents par seconde.

Le délai de traitement se décompose en 26 minutes pour les API concernées par le connecteur Action Request System et de 1 minute pour le traitement traitement dans Connect-It.

Les API englobent le temps de réponse du réseau, l'exécution des 'commit', le respect des règles d'intégrité de la base de données, etc.

- Traitement des documents par le connecteur LDAP = 48 secondes soit **150** documents par seconde.

Le délai de traitement se décompose en 36 secondes pour l'API LDAP et de 12 secondes pour le traitement dans Connect-It.

- Traitement des documents par la boîte de mapping : 2 secondes soit **5100** documents par seconde.

Ce délai correspond au traitement des documents dans Connect-It.

Exemples de vitesse de traitement des documents

Voici quelques exemples de vitesse de traitement des documents :

- Connecteur Base de données
1500 à 2000 documents par seconde.
- connecteur ServiceCenter
450 documents par seconde.
- connecteur Asset Management
400 documents par seconde.

Amélioration de la production des documents

Améliorer la production des documents par un connecteur consiste à obtenir plus rapidement les données provenant d'une application source.

Cette section présente différentes manières de réduire les délais de production des documents.

Optimisation du serveur de données source

La production de documents dépend largement de l'utilisation du serveur de données source ainsi que de sa connexion.

Connect-It exécute très peu de conversion et de traitement des données en production.

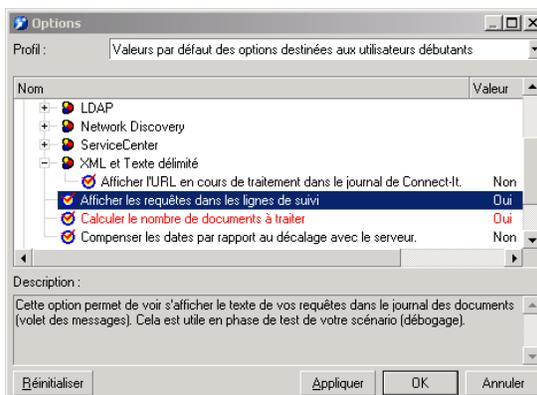
Pour améliorer les résultats, vérifiez :

- le chargement par le serveur des données source
Ce chargement varie en fonction des capacités techniques du serveur : processeur, mémoire disponible, etc.
- la connexion réseau (WAN, LAN)

Contrôle des requêtes exécutées par le connecteur

Pour pouvoir tracer les requêtes transmises à la base de données source :

- 1 Lancez l'éditeur de scénarios.
- 2 Choisissez **Edition/ Options**.
- 3 Dépliez la catégorie **Connecteur**.
- 4 Attribuez la valeur **Oui** à l'option **Afficher les requêtes dans les lignes de suivi**.
- 5 Cliquez sur **OK**.



Une fois cette option sélectionnée, le journal des documents affiche des messages correspondant aux requêtes exécutées par les connecteurs source de votre scénario.

Exemple :

```
SELECT AcctCode, AssetTag, BarCode, dDispos, DisposProfit, dtListPriceCv, Field
2, FullName FROM amAsset
```

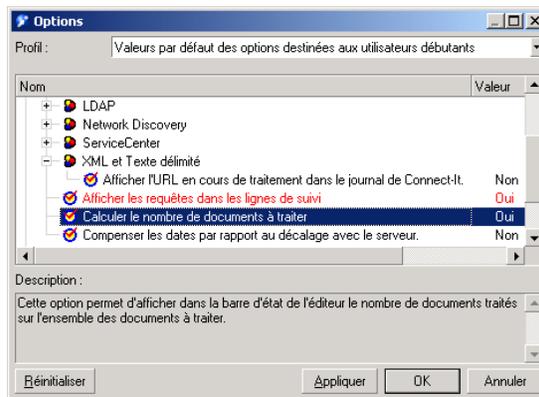
Ces informations vous permettent de :

- vérifier que la requête est générée en conformité avec le type de document défini et les directives de production saisies.
- relancer la requête à l'aide d'un autre outil et observer le délai nécessaire à son exécution

Calcul du nombre de documents à produire

Pour calculer le nombre de documents qu'un connecteur source doit produire :

- 1 Lancez l'éditeur de scénarios.
- 2 Choisissez **Edition/ Options**.
- 3 Dépliez la catégorie **Connecteur**.
- 4 Attribuez la valeur **Oui** à l'option **Calculer le nombre de documents à traiter**.
- 5 Cliquez sur **OK**.



Cette option permet d'afficher une barre de progression dans l'éditeur de scénario indiquant le nombre de documents traités par le connecteur source.

Figure 10.5. Traitement des documents - barre de progression



Pour un connecteur Asset Management, cette option déclenche une requête de type SELECT COUNT.

```
SELECT COUNT(AcctCode) FROM amAsset
```

Exemple :

Avertissement

Cette option doit être désélectionnée en phase d'exploitation de votre scénario car cette requête prend beaucoup de temps.

Utilisation de l'option cache des connecteurs

Pour les connecteurs de type Base de données, l'utilisation d'un cache pour la structure des types de document publiés (métadonnées) est conseillée.

Le cache permet à un connecteur de s'ouvrir plus rapidement puisque le chargement de la description des types de document publiés (métadonnées) est effectué localement.

Pour plus d'informations sur l'utilisation des caches de connecteur, consultez le manuel **Connecteurs**, chapitre **Configuration des connecteurs**, section **Configurer le cache**.

S'il n'utilise pas de cache, un connecteur, à chaque ouverture, obtient un ensemble de données correspondant au descriptif de la base source.

Pour les connecteurs de type Base de données, les données de description sont :

- la liste des tables
- la liste des champs
- la liste des formulaires
- la liste des index
- la liste des liens
- la liste des jointures
- Etc.

Plus ces données sont nombreuses et plus l'ouverture du connecteur est retardée.

 **Note :**

Le temps d'ouverture du connecteur est également allongé quand la bande passante disponible au moment de cette ouverture est faible. Ceci est lié au performance de la connexion au réseau (WAN, LAN).

Avertissement

L'utilisation d'un cache est conseillée lors de l'exploitation d'un scénario.

Cependant, si les données de description de la base source changent, n'oubliez pas de synchroniser le cache de votre connecteur.

Pour synchroniser le cache d'un connecteur :

- 1 Ouvrez l'éditeur de scénarios.
- 2 Ouvrez votre scénario.
- 3 Sélectionnez le connecteur dont vous voulez synchroniser le cache.
- 4 Ouvrez le connecteur (F4).
- 5 Choisissez **Composants\ Cache\ Synchroniser le cache**.

Après la synchronisation du cache, vérifiez que vos mappings ne comprennent pas des éléments qui n'existent plus dans les nouvelles données de description de votre base de données source.

Reconnexion automatique

Pour les connecteurs utilisant comme source de données un serveur distant, il est nécessaire que l'option de reconnexion automatique au serveur soit sélectionnée.

Les options de reconnexion automatique ont un impact sur les délais de latence du réseau : transmission des paquets de données

Pour plus d'informations sur l'option de reconnexion automatique, consultez le manuel **Connecteurs**, chapitre **Configuration des connecteurs**, section **Configurer les paramètres de reconnexion**.

Les informations de reconnexion sont affichées dans le journal de Connect-It.

Exemple de tentative de reconnexion du connecteur Action Request System :

```

Connexion au serveur...
Utilisation de la librairie dynamique 'arapi50.dll'.
(ARS ERROR 90) Impossible d'établir une connexion réseau avec le serveur
AR System. fcitsrv01 : RPC: Name to address translation failed - No such
hostname
Le prochain essai de reconnexion au serveur sera tenté dans 4 s
Tentative de reconnexion...
Utilisation de la librairie dynamique 'arapi50.dll'.
(ARS ERROR 90) Impossible d'établir une connexion réseau avec le serveur
AR System. fcitsrv01 : RPC: Name to address translation failed - No such
hostname
Le prochain essai de reconnexion au serveur sera tenté dans 8 s
Tentative de reconnexion...
Utilisation de la librairie dynamique 'arapi50.dll'.
(ARS ERROR 90) Impossible d'établir une connexion réseau avec le serveur
AR System. fcitsrv01 : RPC: Name to address translation failed - No such
hostname
Le prochain essai de reconnexion au serveur sera tenté dans 16 s

```

Le délai d'attente peut s'avérer important si la connexion au réseau est instable.

Vous devez prendre garde à certaines configurations de connexion de type **Routeur/ Pare-feu/ Serveur** qui closent toute connexion datant de plus d'une durée donnée. Ces paramètres sont à vérifier auprès de l'équipe responsable du réseau local.

Reconnexion à chaque session (spécifique au connecteur Asset Management)

Une option de configuration propre au connecteur Asset Management, vous permet de clore la connexion à la fin de chaque session d'un scénario et de rétablir cette connexion uniquement à l'ouverture d'une nouvelle session.

En effet, plus les ressources du serveur Connect-It sont épargnées de manière optimale, plus les opérations de reconnexion au serveur sont rapides.

Pour plus d'informations sur cette option, consultez le manuel **Connecteurs**, chapitre **Connecteurs Peregrine Systems**, section **Connecteur Asset Management / Configuration du connecteur Asset Management**.

Avertissement

Cette option doit être désélectionnée si la fréquence d'exécution du scénario est importante (Exemple : toutes les cinq minutes).

Pour plus d'informations sur la fréquence d'exécution d'un scénario à l'aide des programmeurs, consultez le chapitre [Exploitation d'un scénario d'intégration](#) [page 179], section [the section called «Créer une programmation»](#) [page 179].

Optimisation de l'utilisation de la clause WHERE

Les directives de production des connecteurs de type Base de données vous permettent de saisir des clauses WHERE.

Pour plus d'informations sur la saisie des clauses WHERE, consultez le manuel **Connecteurs**, chapitre **Directives des connecteurs**, section **Directives de production/ Clauses WHERE et ORDER BY**.

Pour optimiser l'utilisation de vos clauses WHERE, vérifiez que les champs sur lesquels portent ces clauses sont indexés. Si une clause WHERE porte sur un champ non indexé, l'exécution de la clause effectue le balayage de tous les enregistrements de la table.

Gestion des pointeurs

Les pointeurs peuvent être utilisés pour réduire la quantité des données traitées par un scénario.

Un pointeur correspondant à une date de dernière modification d'un enregistrement permet à un connecteur de ne traiter que les enregistrements créés ou mis à jour depuis son dernier lancement.

Exemple : un connecteur source ne traite que les enregistrements d'une table des employés correspondant aux nouvelles personnes arrivées dans l'entreprise. Pour plus d'informations sur l'utilisation des pointeurs, consultez le manuel **Connecteurs**, chapitre **Configuration des connecteurs**, section **Configurer les pointeurs de programmation**.

Dans la plupart des cas, l'état du pointeur se réfère à une date et heure spécifique que vous pouvez modifier dans la fenêtre de programmation du scénario (**Scénario/ Programmation**).

Si le pointeur du scénario utilise un champ non indexé, le traitement des données est long. Afin d'éviter cet inconvénient, il est conseillé d'indexer le champ utilisé comme pointeur.

Amélioration de la consommation des documents

Améliorer la consommation des documents par un connecteur consiste à traiter plus rapidement les données à destination d'une application cible.

Cette section présente différentes manières de réduire les délais de consommation des documents.

Choix des clés de réconciliation

La réconciliation concerne le rapprochement d'anciennes et de nouvelles données. Dans le cas d'un connecteur de type Base de données (Asset Management), la réconciliation consiste à définir les champs qui permettent d'identifier de manière unique les enregistrements des tables que le connecteur crée ou met à jour.

Le choix de ces champs désignés comme clés de réconciliation influe de manière significative sur la consommation des documents par un connecteur destination.

La consommation des documents impliquent les deux actions suivantes :

- 1 Envoi d'une requête utilisant les champs choisis comme clés de réconciliation pour savoir si l'enregistrement existe dans l'application cible.
- 2 Action d'insertion ou de mise à jour de l'enregistrement

Si les champs choisis comme clés de réconciliation ne sont pas indexés, la consommation des documents est ralentie. Exemple : si les champs **memo** ou **FeatParam** sont choisis comme clés de réconciliation pour le connecteur Asset Management, la consommation des documents est très inefficace.

Gestion des transactions

Par défaut, les connecteurs de type Base de données effectue un 'commit' chaque fois qu'il ont consommé un document.

Si la bande passante à votre disposition n'est pas assez importante, vous avez la possibilité d'effectuer le 'commit' par groupe de documents.

Pour fixer le nombre de documents pour lequel un connecteur effectue un commit :

- 1 Ouvrez l'éditeur de scénarios.
- 2 Ouvrez votre scénario.
- 3 Sélectionnez le connecteur pour lequel vous voulez modifier les paramètres de transaction.
- 4 Lancez l'assistant de configuration du connecteur (F2)
Attention : vérifiez que vous êtes en mode de configuration avancée (icône  sélectionnée dans la barre d'outils).
- 5 Cliquez sur **Suivant** plusieurs fois afin d'obtenir la page **Gérer les transactions**.
- 6 Sélectionnez l'option **Faire un 'commit' par un groupe de documents**.
- 7 Indiquez un nombre de documents pour lequel vous voulez que la fonction 'commit' se déclenche.

Pour la consommation d'un nombre important de documents, il est conseillé de lancer la fonction 'commit' avec un groupe contenant de nombreux documents.

Exemples d'amélioration des performances du scénario `ldaplac\complete.scn` :

- 1 Commit pour chaque document (option par défaut)
Performance : 4 minutes pour 6000 documents soit 25 documents par seconde.
- 2 Commit par groupe de 500 documents
Performance : 3 minutes pour 6000 documents soit 33 documents par seconde.

Traitement synchrone vs traitement asynchrone - connecteur ServiceCenter

A chaque document consommé par un connecteur ServiceCenter, correspond une requête envoyée vers une application cible.

Pour améliorer la vitesse de consommation des documents, vous pouvez choisir entre

- un traitement synchrone des données

Chaque document consommé est envoyé après le traitement du document précédent par l'application cible.

- un traitement asynchrone des données

Chaque document consommé est envoyé même si l'application cible n'indique pas que le document précédent a été consommé.

Amélioration du traitement des documents par le moteur de Connect-It

Cette section présente différentes manières de réduire les délais de traitement des documents par le moteur de Connect-It.

Configuration du journal des documents

Configurer le journal des documents de Connect-It vous permet d'indiquer :

- Le type d'erreur que vous voulez obtenir dans le journal des documents (champ **Filtre**).
- Le nombre de maximum de lignes de suivi sauvegardées en mémoire sur le serveur Connect-It.
- un fichier texte dans lequel les messages du journal des documents sont enregistrés.

Pour améliorer les délais de traitement des documents :

- 1 Lancez l'éditeur de scénarios.
- 2 Choisissez **Journal\ Configurer le journal des documents**.
- 3 Limitez le type d'erreur que vous voulez sauvegarder dans le journal des documents en sélectionnant l'option de votre choix dans le champ **Filtre**.

Exemple : Rejet total.

- 4 Limitez le nombre de lignes de suivi sauvegardées en mémoire.

Le nombre maximum de lignes de suivi est 5000.

Ne pas sélectionner l'option **Utiliser les fichiers** permet également d'épargner les ressources du moteur de Connect-It.

Utilisation en mode non graphique

Si l'éditeur de scénarios est affiché durant le déroulement d'un scénario, le moteur de Connect-It est ralenti chaque fois qu'une commande est

sélectionnée : rafraîchissement, consultation du journal des documents, saisie d'une nouvelle option, etc.

Réglage de la programmation

La programmation dans Connect-It consiste à associer un ou plusieurs programmeurs aux types de document produits d'un scénario. Exemple : un programmeur réveille le connecteur Asset Management toutes les heures. A chaque réveil, le connecteur produit des documents correspondant aux enregistrements d'une application Asset Management qu'il consulte.

A chaque réveil du connecteur, Connect-It calcule la quantité de données à traiter et ne s'arrête pas tant que l'ensemble des données n'ait été traité. En fonction des quantités de données à traiter, vous devez régler les programmeurs de votre scénario. Dans le cas d'une migration de base de données importante, il est conseillé de programmer le réveil du connecteur source durant les périodes d'activité réduite (programmation de nuit). Pour les scénarios d'intégration devant traiter peu de données, il est conseillé d'utiliser le programmeur **Synchrone** prédéfini (réveil du connecteur source toutes les secondes).

Pour plus d'informations sur la programmation, consultez dans ce chapitre, la section [Créer une programmation](#) [page 179].

Lancement de scénarios en parallèle

Quand un seul scénario doit migrer de nombreuses données, il est parfois conseillé de le diviser en plusieurs scénarios que l'on lance chacun sur un service (Windows) ou démon (UNIX) différent.

Pour plus d'informations sur les services et les démons, consultez la section [Créer un service Connect-It](#) [page 190] de ce chapitre.

Exemple

Un scénario importe la liste des employés enregistrés dans une base de données vers une autre base de données.

Pour améliorer les délais de migration, créez deux scénarios :

- Le premier scénario migre la liste des employés dont l'initiale du nom est comprise entre A et J.
- Le second scénario migre la liste des employés dont l'initiale du nom est comprise entre K et Z.

La sélection des employés se fait dans la saisie des directives de production (clause WHERE) des connecteurs source de vos deux scénarios.

Pour plus d'informations sur la saisie des clauses WHERE, consultez le manuel **Connecteurs**, chapitre **Directives des connecteurs**, section **Directives de production/ Clauses WHERE et ORDER BY**.

Amélioration des performances d'un scénario utilisant le connecteur Asset Management

Cette section présente différentes manières de réduire les délais de traitement des documents dans un scénario utilisant le connecteur Asset Management.

Indexation du champ dtLastModif

Dans un scénario utilisant le connecteur Asset Management, vous devez ajouter des index sur le champ **dtLastModif** de toutes les tables AssetCenter concernées par le scénario. Ce champ est utilisé systématiquement par Connect-It pour vérifier les enregistrements créés ou modifiés depuis la dernière session.

Pour créer un index sur les tables des biens (amAsset), celle des produits (amProduct) et celle des personnes et services (amEmplDept) exécutez la commande suivante :

```
CREATE INDEX Ast_dtLastModif ON amAsset (dtLastModif)
GO
CREATE INDEX Prod_dtLastModif ON amProduct (dtLastModif)
GO
CREATE INDEX EmplDept_dtLastModif ON amEmplDept (dtLastModif)
GO
```

Pour vérifier que les champs **dtLastModif** sont tous indexés, l'utilisation de l'exécutable adbldg vous permet de vérifier l'exécution des requêtes SQL comportant un filtre sur le champ **dtLastModif** dans la clause WHERE.

Réglages pour une base de données Asset Management - moteur Sybase ASE

Si les fichiers LOG vous indiquent que certaines requêtes prennent du temps car elles comportent de nombreuses tables dans leur partie FROM, il est conseillé d'utiliser l'application Sybase Query Optimizer pour réduire les délais de traitement.

Modification du fichier amdb.ini

Pour améliorer le traitement des requêtes SQL, vous pouvez également modifier le fichier **amdb.ini** de votre application Asset Management en ajoutant la ligne suivante :

```
PostConnectSql= set forceplan on
```

Les lignes suivantes montrent la configuration du fichier **amdb.ini** pour une base de données appelée **DB ASE COPPER** :

```
[DB ASE COPPER]
PostConnectSql=set forceplan on
stmtcache=500
LongDesc=
Engine=Sybase
Location=COPPER
EngineLogin=itam
EnginePassword=78C6143D43925F46F924205FBB42F0FED21594428DDCAC641ED76CDA1
7050EA1A124254200200
ReadOnly=0
CacheDir=
CacheSize=5120000
Base=EDS
Owner=
TableSpace=
TableSpaceIndex=
AmApiDll=aamapi35.dll
UseNTSecurity=0
```

Ces paramètres doivent être configurés dans le fichier **amdb.ini** sur le serveur de la base de données Asset Management.

Si la partie cliente de votre application Asset Management est installée sur votre serveur Connect-It, vous pouvez établir deux connexions différentes reliées à la même base de données Sybase :

- La première connexion utilise les options `PostConnectSql=set forceplan on` et `stmtcache=500`.

```
[DB ASE ConnectIt]
PostConnectSql=set forceplan on
stmtcache=500
LongDesc=
Engine=Sybase
Location=COPPER
EngineLogin=itam
EnginePassword=78C6143D43925F46F924205FBB42F0FED21594428DDCAC641ED76CD
AA17050EA1A124254200200
ReadOnly=0
CacheDir=
CacheSize=5120000
```

```
Base=EDS
Owner=
TableSpace=
TableSpaceIndex=
AmApiDll=aamapi35.dll
UseNTSecurity=0
```

- La seconde connexion n'utilise aucun de ces paramètres

```
[DB ASE COPPER ACGUI]
LongDesc=
Engine=Sybase
Location=COPPER
EngineLogin=itam
EnginePassword=78C6143D43925F46F924205FBB42F0FED21594428DDCAC641ED76CD
AA17050EA1A124254200200
ReadOnly=0
CacheDir=
CacheSize=5120000
Base=EDS
Owner=
TableSpace=
TableSpaceIndex=
AmApiDll=aamapi35.dll
UseNTSecurity=0
```

Pour les clients AssetCenter classiques (autre que Sybase ASE), l'option `stmtcache=500` ne doit pas être utilisé.

Si vous rencontrez des problèmes de performance sur les clients AssetCenter classiques, vous pouvez essayer une des options suivantes :

- `PostConnectSql=set forceplan on`
- `PostConnectSql=set table count 3`
- `PostConnectSql=set table count 2`

Amélioration des performances d'un scénario utilisant un connecteur de type Base de données

Pour les connecteurs de type Base de données utilisant une connexion Sybase native, l'exécution des requêtes SQL peut être améliorée en saisissant l'option `PostConnectSql=set forceplan on` dans les options avancées.

Pour plus d'informations sur les options avancées, consultez le manuel **Connecteurs**, chapitre **Configuration des connecteurs**, section **Configuration avancée/ Options avancées**.

11 Bilans de traitement

CHAPITRE

Un bilan de traitement est un document qu'un connecteur ou la boîte de mapping peut produire chaque fois qu'il ou elle consomme un document. Chaque bilan permet de savoir si document a été traité convenablement ou non. Un document traité normalement correspond à un document dont aucun des éléments n'a été rejeté.

Pour qu'un connecteur source utilise un bilan de traitement, il doit publier un type de document **SuccessReport** (appelé **DirectoryPoolerAction** pour le connecteur InfraTools Desktop Discovery).

Grâce à un mapping entre le bilan de traitement **ProcessReport** et le type de document **SuccessReport**, il est possible au connecteur source d'effectuer une action particulière sur les fichiers d'où il extrait les données qu'il transforme en documents.

Un assistant **Gestion des bilans de traitement** vous permet de créer automatiquement un mapping entre un bilan de traitement et le type de document **SuccessReport** d'un connecteur source.

Contenu d'un bilan de traitement

Le bilan de traitement se présente comme un type de document pouvant être consommé par chaque connecteur et boîte de mapping. Le tableau suivant détaille les informations contenues dans un bilan de traitement.

Tableau 11.1. Informations contenues dans un bilan de traitement

Élément	Information	Type de champ
 ProcessReport	Noeud racine du bilan de traitement	
 DocumentType	Nom du type de document consommé	Texte
 DocumentTypeID	Identifiant unique du document	Texte
 ErrorNumber	Numéro de l'erreur survenue durant la consommation d'un document	Nombre entier 32 bits
 Success	Traitement normal (valeur : 1) ou anormal du document (valeur : 0) Un traitement anormal correspond au rejet total ou partiel du document.	Booléen
 WarningNumber	Nombre d'avertissements émis durant la consommation d'un document	Nombre entier 32 bits
 Logs	Collection correspondant aux messages explicitant les échecs de traitement	
 Date	Date du message	Date et heure
 LogType	Type du message Les valeurs possibles sont : <ul style="list-style-type: none"> • erreur • avertissement • information • information additionnelle 	Nombre entier 32 bits
 Msg	Contenu du message	Texte

Élément	Information	Type de champ
▣ Path	Chemin du document concerné par le message	Texte

Connecteurs pouvant utiliser les bilans de traitement

Pour qu'un connecteur utilise un bilan de traitement, il doit :

- publier un type de document **SuccessReport**
Ce type de document est pour mapper les éléments contenus dans un bilan de traitement (type de document **ProcessReport**).

Les connecteurs disposant d'un type de document **SuccessReport** sont les suivants :

- Connecteur XML
- Connecteur Texte délimité
- Connecteur Peregrine Desktop Inventory
- Connecteur Infratools Desktop Discovery
- Connecteur e-mail (réception)
- Connecteur MQSeries
- disposer d'une page **Définir les actions après traitement** dans son assistant de configuration.

Cette page vous permet de fixer des actions après traitement d'un document une fois que le connecteur a consommé le document **SuccessReport**.

Exemple : la page **Définir les actions après traitement** permet au connecteur de déplacer dans un dossier **error** tous les documents rejetés par un connecteur destination.

Note :

Les connecteurs ne répondant pas à ces deux critères peuvent utiliser les bilans de traitement. Cependant, ils ne peuvent pas utiliser l'assistant **Gestion des bilans de traitement**.

Contenu du type de document SuccessReport

Le contenu du type de document **SuccessReport** varie en fonction des connecteurs.

Les tableaux suivants détaillent le contenu de ces types de documents.

Connecteurs	Élément	Information	Type de champ
<ul style="list-style-type: none"> Connecteur XML Connecteur Texte délimité Connecteur PDI 	SuccessReport	Noeud racine du bilan de traitement	
	Success	Etat du traitement du fichier (réussite ou échec)	Booléen
	UrlFileInfo	Structure contenant les informations sur le document traité	
	CreationDate	Date de création du fichier	Date
	LastModificationDate	Date de dernière modification du fichier	Date
	Path	Chemin du fichier	Texte

Connecteurs	Élément	Information	Type de champ
<ul style="list-style-type: none"> Connecteur InfraTools Desktop Discovery 	DirectoryPoolerAction	Noeud racine du bilan de traitement	
	FileName	Nom du fichier FSF	Texte
	Success	Etat du traitement du fichier (réussite ou échec)	Booléen

Connecteurs	Élément	Information	Type de champ
<ul style="list-style-type: none"> Connecteur e-mail (réception) 	SuccessReport	Noeud racine du bilan de traitement	
	MailInfo		Texte
	UniqueID	Identifiant du message	Entier

Connecteurs	Élément	Information	Type de champ
<ul style="list-style-type: none"> Connecteur MQSeries 	SuccessReport	Noeud racine du bilan de traitement	
	Success	Etat du traitement du fichier (réussite ou échec)	Booléen

MessageInfo	Structure correspondant à un message traité	
MsgID	Identifiant du message	Entier
PutDate	Date et heure de traitement du message	Date

Utilisation de la page Définir les actions après traitement

Le bilan de traitement permet à un connecteur produisant des documents à partir de fichiers d'effectuer des actions en cas de réussite ou d'échec de traitement de ces fichiers.

- Conservation des fichiers dans leur dossier d'origine
- Suppression des fichiers de leur dossier d'origine
- Déplacement des fichiers vers un autre dossier.

Exemple : un dossier **error**.

Pour l'ensemble des connecteurs, les actions après traitement disponibles sont les suivantes :

Pour afficher la page Définir les actions après traitement

- 1 Sélectionnez un des connecteurs disposant de cette page.
Exemple : le connecteur **Texte délimité**.
- 2 Choisissez **Composants/ Configurer (F2)** pour lancer l'assistant de configuration.
- 3 Cliquez sur **Suivant** jusqu'à l'obtention de la page **Définir les actions après traitement**.

La capture d'écran suivante montre la page **Définir les actions après traitement** du connecteur XML.



Assistant Gestion des bilans de traitement

L'assistant **Gestion des bilans de traitement** vous permet, pour un type de document produit par le connecteur source de créer :

- une boucle de rétroaction entre un connecteur destination consommant ce type de document et le connecteur source.

Cette boucle de rétroaction mappe les informations du bilan de traitement produit par le connecteur destination et le type de document SuccessReport du connecteur source.

- éventuellement une boucle de rétroaction entre une boîte de mapping et le connecteur source

Utiliser l'assistant Gestion des bilans de traitement

- 1 Démarrez l'éditeur de scénarios.

- 2 Ouvrez un scénario dans lequel un connecteur source publie un type de document **SuccessReport**.
Pour connaître la liste des connecteurs publiant ce type de document, consultez la section [Connecteurs pouvant utiliser les bilans de traitement](#) [page 215].
- 3 Choisissez **Composants/ Assistant/ Gestion du bilan de traitement**.
- 4 Indiquez dans la page **Choisir un type de document source** le type de document pour lequel vous voulez créer un boucle de rétroaction.
- 5 Indiquez dans la page **Choisir un connecteur destination** le connecteur destination pour lequel vous voulez récupérer le bilan de traitement du type de document source sélectionné à la page précédente.
- 6 Sélectionnez ou non l'option **Forcer la production d'un bilan de traitement à partir d'une boîte de mapping liée au connecteur source**.
- 7 Cliquez sur **Terminer**.

Bilan de traitement - exemple d'utilisation

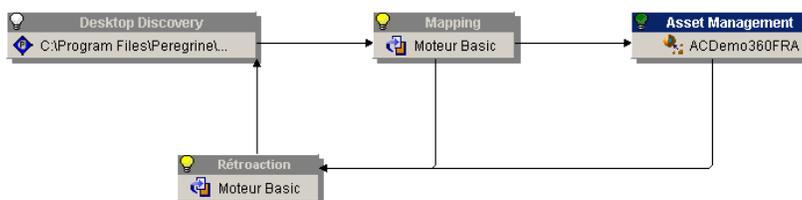
Dans plusieurs scénarios livrés avec Connect-It, les bilans de traitement sont utilisés pour conditionner une action (suppression, déplacement) sur les fichiers utilisés par le connecteur source.

Pour intégrer les bilans de traitement à vos scénarios, il est conseillé d'étudier les scénarios fournis les utilisant.

La section suivante détaille la manière dont sont utilisés les bilans de traitement du connecteur InfraTools Desktop Discovery dans le scénario **idd\iddac36\iddac.scn**. Ce scénario permet d'insérer ou mettre à jour des enregistrements dans une base de données AssetCenter à partir du traitement de fichiers FSF par le connecteur InfraTools Desktop Discovery.

Utilisation des bilans de traitement dans le scénario idd\iddac36\iddac.scn

Figure 11.1. Schéma du scénario idd\iddac36\iddac.scn



Dans ce scénario, une boîte de mapping nommée **Rétroaction** permet de renvoyer les bilans de traitement produits par la boîte de mapping Mapping et le connecteur Asset Management vers le connecteur InfraTools Desktop Discovery.

Cette boîte de mapping contient deux mappings :

- Un mapping entre le bilan de traitement de la boîte de mapping **Mapping** et le type de document **DirectoryPoolerAction** du connecteur source InfraTools Desktop Discovery
- Un mapping entre le bilan de traitement du connecteur Asset Management et le type de document **DirectoryPoolerAction** du connecteur source InfraTools Desktop Discovery

Tableau 11.2. Détail du mapping entre le bilan de traitement produit par la boîte de mapping Mapping et le type de document DirectoryPoolerAction consommé par le connecteur InfraTools Desktop Discovery

Élément du type de document	Élément du bilan de traitement ou script
DirectoryPoolerAction	
DirectoryPoolerAction (noeud racine)	<pre> if [Success] = 1 then PifIgnoreDocumentMapping end if </pre>

Élément du type de document	Élément du bilan de traitement ou script
DirectoryPoolerAction	
Commentaire : script de mapping permettant à la boîte de mapping de ne produire aucun bilan de traitement si un document a été consommé normalement.	
Dans ce scénario, le succès de traitement d'un document est subordonné à son traitement normal par le connecteur Asset Management destination.	
FileName	[\$ParentDoc\$.FileInfo.FileName]
Permet de récupérer le champ FileInfo.FileName du type de document Machine produit par le connecteur InfraTools Desktop Discovery. Ce champ correspond au nom d'un fichier FSF traité par le connecteur.	
Success	Success
Commentaire : champ booléen	
<ul style="list-style-type: none"> la valeur '0' indique que le document a été rejeté partiellement ou en totalité par la boîte de mapping la valeur '1' indique que le document a été traité normalement par la boîte de mapping 	

Tableau 11.3. Détail du mapping entre le bilan de traitement produit par le connecteur Asset Management et le type de document DirectoryPoolerAction consommé par le connecteur InfraTools Desktop Discovery

Élément du type de document	Élément du bilan de traitement ou script
DirectoryPoolerAction	
📁 DirectoryPoolerAction (noeud racine)	
FileName	[\$ParentDoc\$. \$ParentDoc\$.FileInfo.FileName]
Commentaire : permet de récupérer le champ FileInfo.FileName du type de document Machine produit par le connecteur InfraTools Desktop Discovery. Ce champ correspond au nom d'un fichier FSF traité par le connecteur.	
Success	Success
Commentaire : champ booléen	
<ul style="list-style-type: none"> la valeur '0' indique que le document a été rejeté partiellement ou en totalité par le connecteur Asset Management la valeur '1' indique que le document a été traité normalement par le connecteur Asset Management 	

 Note :

Utilisation de la variable \$ParentDoc\$

- La notation [\$ParentDoc\$. champ] permet de récupérer la valeur de ce champ dans le document parent d'un document traité.
 - La notation [\$ParentDoc\$. \$ParentDoc\$. champ] permet de récupérer la valeur de ce champ dans le document parent du document parent d'un document traité.
 - Etc.
-

Effet des bilans de traitement sur les fichiers FSF traités par le connecteur InfraTools Desktop Discovery

Chaque bilan de traitement permet au connecteur InfraTools Desktop Discovery d'effectuer une action particulière sur le fichier FSF correspondant au document concerné par le bilan. Le champ **FileInfo.FileName** de chaque document traité permet d'identifier le fichier FSF lui correspondant.

Cette action particulière est sélectionnée au moment de la configuration du connecteur InfraTools Desktop Discovery.

Exemple :

Vos options de gestion des fichiers FSF sont les suivantes :

- En cas de réussite de traitement d'un document, le fichier FSF lui correspondant est supprimé de son dossier d'origine.
- En cas d'échec de traitement d'un document, le fichier FSF lui correspondant est déplacé vers un dossier **echec**.

Une fois effectué le traitement des documents par le scénario, il vous suffit d'ouvrir le dossier **echec** pour y trouver tous les fichiers FSF dont les données n'ont pas pu être transférées convenablement vers votre application Asset Management.

Pour prendre connaissance des problèmes advenus lors du rejet partiel ou total d'un document, consultez le journal des documents de l'éditeur de scénario.

Pour les fichiers FSF dont les documents correspondants ont été traités convenablement par le connecteur Asset Management, les bilans de traitement positifs (champ **Success** dont la valeur a été "1") ont permis au connecteur InfraTools Desktop Discovery de les supprimer de leur dossier d'origine.

Didacticiel

Le didacticiel de cette section vous permet d'utiliser l'assistant **Gestion du bilan de traitement** dans un scénario utilisant un connecteur XML source et un connecteur destination Asset Management.

L'ensemble des données nécessaires à ce didacticiel est contenu dans le dossier suivant : **[Dossier d'installation de Connect-It]\datakit\tutorial\pro_rep**.

Prérequis

Pour utiliser ce didacticiel, vous devez :

- ouvrir le scénario **pro_rep.scn** se trouvant dans le dossier du didacticiel.
- avoir accès à une base AssetCenter 4.1 ou 4.2 de test.

Description du scénario du didacticiel

Dans le scénario du didacticiel :

- un connecteur XML source produit des documents à partir des fichiers **.xml** contenus dans le sous-dossier **files** du dossier du didacticiel.
- Une boîte de mapping consomme ces documents et les produit pour que le connecteur Asset Management puisse les consommer.
- un connecteur Asset Management crée des enregistrements dans la table **amAbsence** d'une base AssetCenter de test.

Mapping entre le connecteur XML et le connecteur Asset Management

Le tableau suivant détaille le mapping du scénario entre le type de document **amAbsence** du connecteur XML et le type de document **amAbsence** du connecteur Asset Management.

Élément du type de document destination	Élément du type de document source	Script de mapping
amAbsence	amAbsence	<pre>if [Nature] = "" then PifRejectDocumentMapping("Document rejected: missing nature") end if</pre> <p>Ce script permet de rejeter le document si le champ Nature du document produit par le connecteur XML est vide.</p>
AccountingType	AccountingType	AccountingType
Nature	Nature	Nature (clé de réconciliation)
PhoneContact	PhoneContact	PhoneContact (clé de réconciliation)

Fichiers du didacticiel

Le sous-dossier **files** du dossier du didacticiel contient les fichiers **.xml** suivants :

- **success.xml**

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<amAbsence>
  <AccountingType>Convention</AccountingType>
  <bArchived>1</bArchived>
  <dtBack>2002-06-14 18:00:00</dtBack>
  <dtLastModif>2002-07-12 05:30:32</dtLastModif>
  <dtOut>2002-06-14 09:00:00</dtOut>
  <dtValidation>2002-01-15 18:00:00</dtValidation>
  <fDays>1</fDays>
  <Field1>Test</Field1>
  <Nature>Meeting</Nature>
  <PhoneContact>06.12.11.81</PhoneContact>
  <Reason>San Diego Convention</Reason>
  <seValidated>1</seValidated>
</amAbsence>
```

Ce fichier doit être traité avec succès par tous les composants du scénario et doit être déplacé après le test du scénario dans le dossier **success**.

- **src_fail.xml**

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<amAbsent>
  <AccountingType>Convention</AccountingType>
  <bArchived>1</bArchived>
  <dtBack>2002-06-14 18:00:00</dtBack>
  <dtLastModif>2002-07-12 05:30:32</dtLastModif>
  <dtOut>2002-06-14 09:00:00</dtOut>
```

```

<dtValidation>2002-01-15 18:00:00</dtValidation>
<fDays>1</fDays>
<Field1>Field1</Field1>
<Nature>Meeting</Nature>
<PhoneContact>06.12.11.81</PhoneContact>
<Reason>San Diego Convention</Reason>
<seValidated>1</seValidated>
</amAbsent>

```

Ce fichier doit être rejeté par le connecteur XML car la structure **amAbsent** ne correspond pas à celle définie dans la DTD **pro_rep.dtd** utilisée par le connecteur.

Après le test du scénario, ce fichier doit être déplacé dans le dossier **error**.

- **map_fail.xml**

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<amAbsence>
  <AccountingType>Convention</AccountingType>
  <bArchived>1</bArchived>
  <dtBack>2002-06-14 18:00:00</dtBack>
  <dtLastModif>2002-07-12 05:30:32</dtLastModif>
  <dtOut>2002-06-14 09:00:00</dtOut>
  <dtValidation>2002-01-15 18:00:00</dtValidation>
  <fDays>1</fDays>
  <Field1>Test</Field1>
  <Nature></Nature>
  <PhoneContact>06.12.11.81</PhoneContact>
  <Reason>San Diego Convention</Reason>
  <seValidated>1</seValidated>
</amAbsence>

```

Ce fichier doit être rejeté par la boîte de mapping car l'élément **<Nature>** est vide.

Après le test du scénario, ce fichier doit être déplacé dans le dossier **error**.

- **dest_fail.xml**

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<amAbsence>
  <AccountingType>Convention</AccountingType>
  <bArchived>1</bArchived>
  <dtBack>2002-06-14 18:00:00</dtBack>
  <dtLastModif>2002-07-12 05:30:32</dtLastModif>
  <dtOut>2002-06-14 09:00:00</dtOut>
  <dtValidation>2002-01-15 18:00:00</dtValidation>
  <fDays>1</fDays>
  <Field1>Test</Field1>
  <Nature>Meeting</Nature>
  <PhoneContact></PhoneContact>
  <Reason>San Diego Convention</Reason>
  <seValidated>1</seValidated>
</amAbsence>

```

Ce fichier doit être rejeté par le connecteur Asset Management car l'élément **<PhoneContact>** choisi comme clé de réconciliation (directives de consommation du connecteur) est vide.

Après le test du scénario, ce fichier doit être déplacé dans le dossier **error**.

Etape n°1 - Configuration du connecteur XML du scénario d'exemple

- 1 Sélectionnez le connecteur XML du scénario.
- 2 Cliquez sur **F2**.
- 3 A la page **Choisir un mode de traitement**, sélectionnez l'option **Lecture**.
- 4 A la page **Sélectionnez un protocole de connexion**, choisissez l'option **Fichier(s) en local ou réseau**.
- 5 A la page **Choisir un fichier ou un dossier** :
 - 1 Sélectionnez l'option **Lecture des fichiers d'un dossier**.
 - 2 Cliquez sur  et indiquez l'emplacement du sous-dossier **files** du dossier du didacticiel.
- 6 A la page **Définir les actions après traitement**, sélectionnez l'option **Le déplacer vers un dossier** dans les deux cadres de la page.

Pour le cadre **En cas de traitement réussi d'un fichier**, indiquez le chemin du sous-dossier **success**.

Pour le cadre **En cas d'échec de traitement d'un fichier**, indiquez le chemin du sous-dossier **error**.
- 7 A la page **Choisir une DTD/XSD**, indiquez le chemin de la DTD **pro_rep.dtd** du dossier du didacticiel.
- 8 Cliquez sur **Terminer**.

Etape n°2 - Configuration du connecteur Asset Management du scénario d'exemple

- 1 Sélectionnez le connecteur Asset Management du scénario.
- 2 Cliquez sur **F2**.
- 3 Dans la page **Définir les paramètres de connexion**, saisissez les paramètres de connexion de votre base AssetCenter de test.

- 4 Cliquez sur **Terminer**.

Etape n'3 - Utilisation de l'assistant Gestion du bilan de traitement

- 1 Choisissez **Composants/ Assistant/ Gestion du bilan de traitement**.
- 2 A la page **Choisir un type de document source**, sélectionnez le type de document source **amAbsenceSrc**.
- 3 Cliquez sur **Suivant**.
- 4 A la page **Choisir un connecteur destination**, sélectionnez le connecteur **Asset Management**.
- 5 A la page **Options avancées**, sélectionnez **Forcer la production du bilan de traitement à partir d'une boîte de mapping liée au connecteur source**.
- 6 Cliquez sur **Terminer**.

L'assistant crée une boîte de mapping entre le connecteur XML et le connecteur Asset Management. Dans cette boîte de mapping, un mapping est créé. Il est détaillé dans le tableau suivant.

Élément du type de document SuccessReport consommé par le connecteur XML	Élément du bilan de traitement produit par le connecteur Asset Management
Success	Success
Commentaire : champ booléen	
<ul style="list-style-type: none"> • la valeur '0' indique que le document a été rejeté par le connecteur Asset Management. • la valeur '1' indique que le document a été traité normalement par le connecteur Asset Management 	
UrlFileInfo	[\$ParentDoc\$. \$ParentDoc\$.UrlFileInfo.Path]
Permet de récupérer le champ UrlFileInfo.Path du type de document amAbsence produit par le connecteur XML. Ce champ correspond au nom du fichier .xml lu par le connecteur XML.	

Etape n'4 - Test de la boucle de rétroaction créée par l'assistant Gestion du bilan de traitement

- 1 Sélectionnez le connecteur XML.

- 2 Cliquez sur ▶.
Dans le schéma du scénario, des icônes  doivent s'afficher sur les composants du scénario. Ces icônes signalent un échec de traitement des fichiers **.xml**. Sélectionnez l'onglet Journal des documents de l'éditeur de scénario et lisez les messages d'erreur générés.
- 3 Ouvrez le sous-dossier **success** et vérifiez que le fichier **success.xml** y est bien présent.
- 4 Ouvrez le sous-dossier **error** et vérifiez que les fichiers **dest_fail.xml**, **map_fail.xml** et **src_fail.xml** y sont bien présents.

12 | Kit de développement Java de Connect-It

CHAPITRE

Le kit de développement Java de Connect-It vous permet de développer vos propres connecteurs.

Ce kit de développement repose sur les technologies XML et la norme Java Connector Architecture (JCA) 1.0.

Pour une présentation complète de cette norme, consultez le site suivant : <http://java.sun.com/j2ee/connector/>

Contenu du kit de développement java

Le tableau suivant détaille les packages du kit de développement Java de Connect-It.

Tableau 12.1. Contenu du kit de développement java de Connect-It

Packages	Fonction
com.peregrine.common.assert	Mécanisme d'assertion (mode debug)
com.peregrine.common.collection	Classe des collections

Packages	Fonction
com.peregrine.common.ids	Fournit des outils permettant de traiter les fichiers STR. Utile pour localiser les chaînes de vos applications.
com.peregrine.common.io	Classes relatives à la gestion de fichiers.
com.peregrine.common.log	Fournit un mécanisme créant et distribuant les messages des fichiers journaux (LOG).
com.peregrine.common.text	Outils de formatage des chaînes de caractères.
com.peregrine.common.thread	Gestion des threads
com.peregrine.common.util	Outils divers
com.peregrine.common.xml	Manipulation XML
com.peregrine.conit.core	Coeur du framework Connect-It
com.peregrine.conit.document	Norme Unified Document Content utilisée par Connect-It
com.peregrine.conit.document.description	Métadonnées XML Connect-It
com.peregrine.conit.document.schema	API de support XMLSchema
com.peregrine.conit.ra	Connecteurs JCA
com.peregrine.resourceimpl.cci	Implémentation des classes CCI
com.peregrine.resourceimpl.event	Implémentation des classes événementielles (EVENT)
com.peregrine.resourceimpl.spi	Implémentation des classes SPI
com.peregrine.resourceimpl.tools	Outils de déploiement
com.peregrine.resourceimpl.util	Outils d'aide à l'implémentation JCA

Informations sur l'environnement Java

Pour obtenir les informations relatives à votre environnement Java :

- 1 Choisissez **Aide/ A propos**.
- 2 Cliquez sur **Plus**.

Une fenêtre s'affiche récapitulant les informations de votre environnement Java (noeud **Java**).

Implémentation de la norme JCA

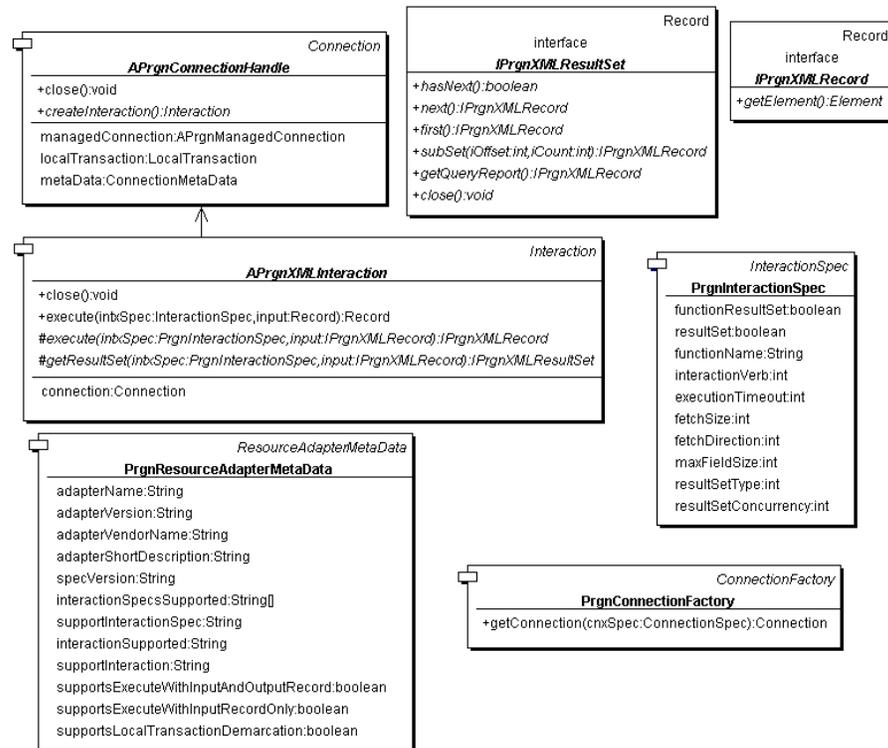
Cette section est divisée en trois sous-sections :

- L'implémentation des classes **CCI** (Common Client Interface)
- L'implémentation des classes **SPI** (Service Provider Interface)
- L'extension de la norme JCA avec la création des classes **EVENT**

L'implémentation des classes CCI

Interface	<code>javax.resource.cci.Connection</code>
Classe implémentée	<code>com.peregrine.resourceimpl.cci.APrgnConnectionHandle</code>
Interface	<code>javax.resource.cci.ConnectionFactory</code>
Classe implémentée	<code>com.peregrine.resourceimpl.cci.PrgnConnectionFactory</code>
Interface	<code>javax.resource.cci.ConnectionMetaData</code>
Classe implémentée	<code>com.peregrine.resourceimpl.spi.APrgnManagedConnectionMetaData</code>
Interface	<code>javax.resource.cci.Interaction</code>
Classe implémentée	<code>com.peregrine.resourceimpl.cci.APrgnXMLInteraction</code>
Interface	<code>javax.resource.cci.InteractionSpec</code>
Classe implémentée	<code>com.peregrine.resourceimpl.cci.PrgnInteractionSpec</code>
Interface	<code>javax.resource.cci.ResourceAdapterMetaData</code>
Classe implémentée	<code>com.peregrine.resourceimpl.cci.PrgnResourceAdapterMetaData</code>
Interface	<code>javax.resource.cci.Record</code>
Classe implémentée	<code>com.peregrine.resourceimpl.cci.IPrgnXMLRecord</code> <code>com.peregrine.resourceimpl.cci.IPrgnXMLResultSet</code>
Interface	<code>javax.resource.cci.LocalTransaction</code>
Classe implémentée	<code>com.peregrine.resourceimpl.spi.APrgnLocalTransaction</code>

Figure 12.1. Kit de développement Java - Classes CCI



Les sections suivantes détaillent les classes principales de la norme JCA.

javax.resource.cci.Connection

Cette classe représente le niveau applicatif client (référence (handle) de connexion) nécessaire à l'établissement d'une connexion physique.

Implémentation

La classe de base **APrgnConnectionHandle** est une classe abstraite. Elle permet d'accéder via un objet **ManagedConnection** à la connexion physique. Le pont effectué par l'instance **ManagedConnection** permet le partage de la ressource physique par différents clients. Pour assurer une cohérence entre ces deux niveaux, un mécanisme de validation de connexion est utilisé.

javax.resource.cci.ConnectionFactory

C'est le point d'entrée du composant client permettant l'établissement d'une connexion via l'une des deux méthodes `getConnection()`.

Implémentation

La classe `PrgnConnectionFactory` est une classe concrète prête à l'emploi. Dans le cas d'un environnement non-managé, cette classe crée par défaut une instance de l'interface `ConnectionManager` représentée par l'objet `PrgnNonManagedConnectionManager`.

javax.resource.cci.ConnectionMetaData

Cette classe a pour but de fournir des informations sur le système à travers la connexion de type `Connection`. C'est le pendant client de l'interface spi `ManagedConnectionMetaData`.

Implémentation

L'implémentation de base choisie s'appuie sur le contrat de "programmation par interface". La classe abstraite `APrgnManagedConnectionMetaData` regroupe ainsi les rôles décrits par les interfaces `cci` et `spi`.

javax.resource.cci.LocalTransaction

Cette classe représente le niveau applicatif d'une transaction. Cette interface est liée à l'interface spi `LocalTransaction`.

Implémentation

Comme précédemment, les rôles `cci` et `spi` sont regroupés dans la classe de base `APrgnLocalTransaction` du package `Service Provider Interface`.

javax.resource.cci.ResourceAdapterMetaData

Cette classe décrit l'information correspondant au connecteur.

Implémentation

La classe `PrgnResourceAdapterMetaData` est une classe concrète fournissant les accesseurs (méthodes `getXXX` et `setXXX`) des propriétés décrites par l'interface.

javax.resource.cci.Record

Cette classe représente une donnée d'entrée ou de sortie lors d'une interaction avec le système.

Implémentation

Anticipant la norme future, tous les objets de type **Record** supportés par le framework d'implémentation sont basés sur le standard XML. Deux types sont ainsi manipulés :

- **IPrgnXMLRecord**
Correspond à un document XML simple.
- **IPrgnXMLResultSet**
Correspond à un itérateur sur un ensemble de documents XML.

L'implémentation des classes SPI

Interface	javax.resource.spi.ConnectionEventListener
Classe implémentée	com.peregrine.resourceimpl.spi.APrgnConnectionEventListener

Interface	javax.resource.spi.ConnectionManager
Classe implémentée	com.peregrine.resourceimpl.spi.PrgnNonManagedConnectionManager

Interface	javax.resource.spi.ConnectionRequestInfo
Classe implémentée	com.peregrine.resourceimpl.spi.PrgnConnectionRequestInfo

Interface	javax.resource.spi.LocalTransaction
Classe implémentée	com.peregrine.resourceimpl.spi.APrgnLocalTransaction

Interface	javax.resource.spi.ManagedConnection
Classe implémentée	com.peregrine.resourceimpl.spi.APrgnManagedConnection

Interface	javax.resource.spi.ManagedConnectionFactory
Classe implémentée	com.peregrine.resourceimpl.spi.APrgnManagedConnectionFactory

Interface	javax.resource.spi.ManagedConnectionMetaData
-----------	--

Classe implémentée `com.peregrine.resourceimpl.spi.APrgnManagedConnectionMetaData`

Figure 12.2. Kit de développement Java - Classes SPI



javax.resource.spi.ConnectionEventListener

Cette classe permet de recevoir les notifications de l'instance **ManagedConnection** concernant les connexions client.

Implémentation

La classe abstraite **APrgnConnectionEventListener** implémente les méthodes décrites par l'interface. Ces méthodes étant vides, il convient de surcharger les fonctionnalités appropriées (design pattern 'adapter').

javax.resource.spi.ConnectionManager

Permet de relier le connecteur et le serveur applicatif pour la gestion du pooling, la gestion des transactions, etc.

Implémentation

Une implémentation concrète par défaut est fournie par le connecteur (requis dans un environnement non managé). La fonction pooling est ajoutée.

javax.resource.spi.ConnectionRequestInfo

Ensemble (container) des propriétés d'un client spécifique faisant partie intégrante de la requête en continu de la connexion.

Implémentation

PrgnConnectionRequestInfo supporte les propriétés **Nom d'utilisateur** et **Mot de passe**. Une implémentation de connecteur spécifique peut ajouter d'autres propriétés.

Permet également l'implémentation de la classe **javax.resource.cci.ConnectionSpec**.

javax.resource.spi.LocalTransaction

Permet de supporter la gestion des transactions locales.

Implémentation

APrgnLocalTransaction est liée à l'instance **APrgnManagedConnection** et fournit les méthodes pour la notification événementielle de transaction.

javax.resource.spi.ManagedConnection

Représente la connexion physique à l'instance d'EIS.

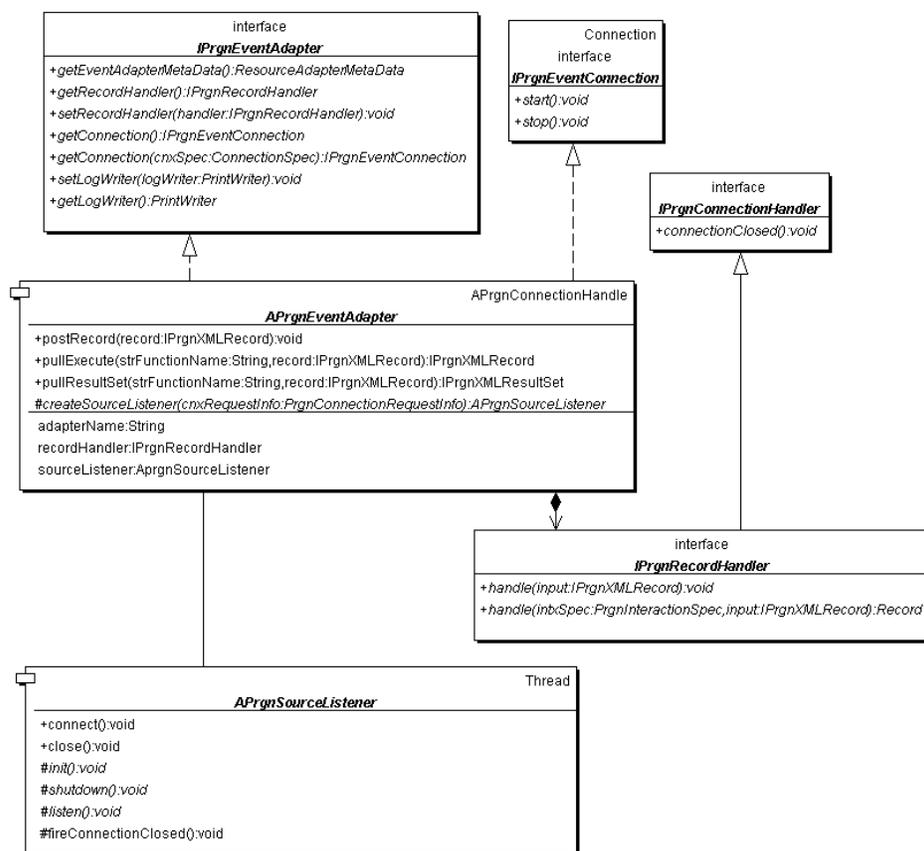
Implémentation

APrgnManagedConnection supporte le partage de connexions en maintenant un ensemble de références (handles) de connexions.

Extension de JCA : les classes événementielles

La norme JCA ne couvre pas le mode de fonctionnement asynchrone. Pour que ce mode événementiel soit utilisé, il faut que l'EIS soit capable de gérer des interactions et des notifications de renvois d'appels. De telles interactions (événements) sont gérées à l'aide des classes présentées dans les tableaux suivants.

Figure 12.3. Kit de développement Java - classes événementielles



com.peregrine.resourceimpl.event.IPrgnEventAdapter

```
//Métadonnées du connecteur événementiel
javax.resource.cci.ResourceAdapterMetaData getEventAdapterMetaData()

//Enregistrement du connecteur d'écoute événementiel auprès de l'EIS
IPrgnRecordHandler getRecordHandler()
void setRecordHandler(IPrgnRecordHandler handler)

//Méthodes de connexion
IPrgnEventConnection getConnection() throws javax.resource.ResourceException
IPrgnEventConnection getConnection(javax.resource.cci.ConnectionSpec cnxSpec) throws javax.resource.ResourceException

//Méthodes d'ouverture de session
void setLogWriter(PrintWriter logWriter) throws ResourceException
PrintWriter getLogWriter() throws ResourceException
```

com.peregrine.resourceimpl.event.IPrgnEventConnection

```
//Méthodes de traitement événementiel
void start() throws javax.resource.ResourceException
void stop() throws javax.resource.ResourceException
```

com.peregrine.resourceimpl.event.IPrgnRecordHandler

```
//Notification de la connexion
void connectionClosed();
```

Norme UDC (Unified Document Content)

Dans Connect-It, les données transitent sous forme de documents XML conformes à des types de document produits et consommés par les connecteurs.

La structure de ces types de document respecte la norme UDC (**Unified Document Content**) spécifique à Connect-It.

La norme UDC définit trois types d'élément (ou noeud) dans un type de document :

- **<STRUCTURE>**

Cet élément correspond à une structure dans un type de document publié par un connecteur.

- **<COLLECTION>**
Cet élément correspond à une collection dans un type de document publié par un connecteur.
- **<ATTRIBUTE>**
Cet élément correspond à un champ dans un type de document publié par un connecteur.

Élément	Sous-éléments supportés	Contenu mixte supporté	Contenu de données supporté	Cardinalité de contenu
<COLLECTION>	<ul style="list-style-type: none"> • <COLLECTION> • <STRUCTURE> • <ATTRIBUTE> 	• Non	• Non	Unbounded
<STRUCTURE>	<ul style="list-style-type: none"> • <COLLECTION> • <STRUCTURE> • <ATTRIBUTE> 	• Oui	• Non	Bounded
<ATTRIBUTE>	• Non	• Non	• Oui	N/A

Les types de données supportés par les éléments **<ATTRIBUTE>** sont les suivants :

- Boolean
- Char
- Short
- Integer
- Long
- Float
- Double
- String
- Date
- Time
- Timestamp
- Blob
- Memo

Les types de document publiés par les connecteurs Java respectent cette norme. Le noeud racine de chaque type de document publié correspond à un élément **<STRUCTURE>** dont le nom correspond à un ensemble de données de

l'application auquel se connecte le connecteur. **Exemple** : cet ensemble de données est le nom d'une table pour les connecteurs de type Base de données.

Exemple de type de document publié par le connecteur Java JDBC

```
<STRUCTURE name="myTable">
  <ATTRIBUTE name="field1" type="String"/>
  <ATTRIBUTE name="field2" type="Long"/>
  <COLLECTION name="link">
    <STRUCTURE name="myLinkedTable">
      <ATTRIBUTE name="field3" type="String"/>
      <ATTRIBUTE name="field4" type="Memo"/>
    </STRUCTURE>
  </COLLECTION>
</STRUCTURE>
```

Tous les noeuds d'un document doivent avoir un nom unique sauf :

- si leurs champs (éléments **<ATTRIBUTE>**) contiennent les mêmes données
- si les éléments **<COLLECTION>** et **<STRUCTURE>** qu'ils englobent sont identiques.

Implémentation de la norme UDC

Le kit de développement Java de Connect-It fournit de nombreuses classes pour la manipulation des documents XML et UDC.

Elles sont regroupées dans le package **com.peregrine.conit.document** :

- PrgnDocument
Englobe la définition générique des documents XML.
- APrgnElement
Classe parente des wrappers UDC.
- PrgnAttribute
Wrapper du noeud **<ATTRIBUTE>**
- APrgnElementContainer
Classe parente des noeuds **<STRUCTURE>** et **<COLLECTION>**
- PrgnStructure
Wrapper du noeud **<STRUCTURE>**
- PrgnCollection
Wrapper du noeud **<COLLECTION>**

Un type de document doit être implémenté avec le wrapper **PrgnStructure**.

Pour plus d'informations sur ces classes, consultez la documentation HTML **Javadoc** en cliquant sur le fichier **[Dossier d'installation de Connect-It]\doc\javadoc\conit\index.html**.

Création d'un connecteur Java

Cette section présente la méthodologie de création d'un connecteur Java sous Connect-It.

Prérequis

La création d'un connecteur Java dépend du mode de fonctionnement de l'application externe (EIS).

Les deux grands modes de fonctionnement d'une application externe (EIS) sont les suivants :

- Le mode Requête/Réponse (Exemple : une base de données relationnelle)
- Le mode Événement/Ecoute (Exemple : une messagerie)

Les sections suivantes détaillent la création d'un connecteur Java utilisant le mode Requête/Réponse. La création des connecteurs événementiels est présentée dans ce chapitre, section [Création d'un connecteur événementiel](#) [page 261].

Définir le contrat de connexion

Pour définir un contrat de connexion, vous devez définir :

- Le type de connexion de l'application externe (EIS)
- Les paramètres nécessaires à un poste client pour établir cette connexion
Ces paramètres correspondent aux noeuds `<config-property>` du fichier de déploiement XML (Exemple : **ra.xml**).

Type de connexion

La connexion physique est englobée par la classe dérivée **com.peregrine.resource.impl.spi.APrgnManagedConnection**.

Vous devez écrire les méthodes suivantes :

- `cleanupPhysicalConnection(): void`

Cette méthode doit implémenter :

- l'action à effectuer une fois la connexion poolée et envoyée
- la préparation de la connexion pour une prochaine utilisation
- `destroyPhysicalConnection() : void`

Cette méthode doit implémenter la fermeture de la connexion à l'application externe (EIS).

Vous devez également rédiger le code permettant l'accès aux métadonnées de l'application externe (EIS) à travers la connexion physique :

- en écrivant une classe héritant de **`com.peregrine.resourceimpl.spi.APrgnManagedConnectionMetaData`**.
- en surchargeant la méthode **`getMetaData()`** : **`ManagedConnectionMetaData`** pour renvoyer une instance de cette classe.

Paramètres de création de la connexion

Le rôle de la classe **`APrgnManagedConnectionFactory`** est de créer des connexions physiques vers l'application externe (EIS).

Pour toutes les propriétés de connexion définies dans le fichier de déploiement, vous devez rédiger des méthodes publiques de collecte (`get`) et d'initialisation (`set`) des données pour créer les connexions physiques (Exemples : le nom d'une base de données, le nom d'un serveur, etc.)

Vous devez également surcharger la méthode **`getMetaData()`** :

`ResourceAdapterMetaData` pour exposer les métadonnées du connecteur. Ceci permet de connaître le type d'interaction supporté. Cette méthode doit retourner une instance de **`com.peregrine.resourceimpl.cci.PrgnResourceAdapterMetaData`**.

Paramètres requis pour obtenir la connexion d'un client

Le code utilisant ces classes JCA réclame une connexion grâce à un objet **`PrgnConnectionRequestInfo`**. C'est un composant prêt à l'emploi utilisant les paramètres de connexion **Nom d'utilisateur** et **Mot de passe**. C'est à cet endroit du code que vous pouvez placer des paramètres additionnels nécessaires à l'obtention d'une connexion utilisateur.

Définir le contrat de transaction

Définir le contrat de transaction implique que l'application externe (EIS) supporte le niveau transactionnel.

 **Note :**

Connect-It ne supporte que les transactions locales.

Pour définir le contrat de transaction, vous devez utiliser la méthode `getLocalTransaction() : LocalTransaction` de la classe **`com.peregrine.resourceimpl.spi.APrgnManagedConnection`**.

Si cette méthode n'est pas supportée, lancez une exception **`javax.resource.NotSupportedException`**. Dans le cas contraire, écrivez une classe dérivant de **`com.peregrine.resourceimpl.spi.APrgnLocalTransaction`** et utilisez les méthodes suivantes :

- `begin() : void`
Démarre une nouvelle transaction.
- `commit() : void`
Commit de la transaction.
- `rollback() : void`
Roll back de la transaction.

Vous pouvez également utiliser les méthodes de notification d'état :

- `fireLocalTransactionStarted() : void`
Informe la connexion managée liée que la transaction locale a démarré.
- `fireLocalTransactionCommitted() : void`
Informe la connexion managée liée que le commit de la transaction locale a été effectué.
- `fireLocalTransactionRolledBack() : void`
Informe la connexion managée liée que le roll back de la transaction locale a été effectué.

Définir le contrat de sécurité

Définir un contrat de sécurité vous permet de sécuriser l'accès à l'application externe (EIS).

Les connexions physiques sont créées par la classe de base **APrgnManagedConnectionFactory** dans un contexte sécurisé.

L'implémentation de base par défaut est le mécanisme d'authentification **BasicPassword**. Cette implémentation convient dans la plupart des cas.

L'information de sécurité utilisée pour l'authentification est l'objet **javax.resource.spi.security.PasswordCredential** (Nom d'utilisateur et mot de passe).

La méthode **getPasswordCredential(Subject, ConnectionRequestInfo)** extrait les informations de sécurité pour la Factory à partir de l'objet **javax.security.auth.Subject** donné. Si il n'y a pas de sujet (C'est le cas pour Connect-It), la méthode utilise directement l'information contenue dans l'objet **com.peregrine.resourceimpl.spi.PrgnConnectionRequestInfo** à travers un **PasswordCredential** créé.

Vous devez rédiger la méthode suivante :

```
createManagedConnection(PasswordCredential) :
APrgnManagedConnection
```

Si un autre mécanisme d'authentification est nécessaire, vous devez surcharger la méthode de plus haut niveau :

```
createManagedConnection(Subject,
ConnectionRequestInfo) : ManagedConnection
```

Dans la classe dérivée **APrgnManagedConnection**, surchargez la méthode suivante :

```
newConnectionHandle(Subject, ConnectionRequestInfo) :
APrgnConnectionHandle
```

Pour créer la référence à une nouvelle connexion :

- Si le client ne supporte pas la réauthentification sur l'instance physique de l'application externe (EIS), il faut ignorer les informations de sécurité. Dans les cas contraires, utilisez le même mécanisme de sécurité que celle de **APrgnManagedConnectionFactory**.
- Si la réauthentification est supportée, le connecteur change le contexte de sécurité de l'instance de **APrgnManagedConnection** avec celui de l'instance **Subject** passée.

Définir les données

Dans Connect-It, les connecteurs échangent des données sous forme de documents XML. Le contenu de ces documents repose sur la norme Unified Document Content (UDC). Chaque document est basé sur un Type de document, reposant lui-aussi sur la norme UDC. Un document est donc représenté par une structure (noeud <STRUCTURE>) contenant des champs (noeud <ATTRIBUTE>), des structures (noeud <STRUCTURE>) et des collections (noeud <COLLECTION>).

Pour une présentation de la norme UDC, consultez dans ce chapitre, la section [Norme UDC \(Unified Document Content\)](#) [page 238].

Définir l'interaction sur les données

Définir l'interaction sur les données permet de définir les opérations exécutées par la partie cliente de la norme JCA. Le plus important concerne les méthodes d'accès aux données de l'application externe (EIS).

Les interactions des connecteurs de Connect-It sont des interactions XML. Les composants de base fournissent le support pour ces interactions avec la classe **com.peregrine.resourceimpl.APrgnXMLInteraction**. Le résultat unique ou multiple d'une méthode d'interaction XML peut être de deux types :

- XMLRecord
- XMLResultset

Les composants de base d'un connecteur définissent un certain nombre d'interactions pour la configuration, la connexion, l'auto-description, etc. Ces opérations sont communes à tous les connecteurs et ne doivent pas être modifiées ou surchargées.

En conséquence, les interactions propres au connecteur ne doivent pas avoir le même nom. Voici la liste des noms réservés:

- **connect**
- **disconnect**
- **start**
- **stop**
- **setConfig**
- **getConfig**
- **getOperations**
- **describe**

- **getRecords**
- **viewRecords**
- **getRequests**
- **getRequest**
- **setEventFilter**
- **startTransaction**
- **endTransaction**
- **commitTransaction**
- **prepareTransaction**
- **rollbackTransaction**
- **forgetTransaction**
- **recoverTransaction**
- **reconc**
- **getDocumentChildrenName**
- **getDocumentsName**

Pour définir les opérations propres à un connecteur, nous fournissons la classe **com.peregrine.conit.ra.APrgnDocumentInteraction**, qui est une dérivation de la classe **APrgnXMLInteraction**. Sa fonction est d'encapsuler les méthodes d'interaction en tant qu'opérations XML du connecteur. Elle fournit également le support pour l'enregistrement de ces méthodes.

Il existe trois types d'opération :

- **status**

Une opération **status** correspond au fonctionnement du connecteur en mode consommation. Cette opération requiert des paramètres de type "données" (data) et retourne un état du traitement de ces données (échec ou réussite).

Exemple : pour un connecteur de type Base de données, '**insert**' est une opération de type **status**.

- **resultSet**

Une opération **resultSet** correspond au fonctionnement du connecteur en mode production. Cette opération requiert des paramètres de type "schéma de données" (type de document) et retourne un ensemble de données.

Exemple : pour un connecteur de type Base de données, '**query**' est une opération de type **resultSet**. Cette opération prend en paramètre le nom

d'une table et une liste de champs sur lesquels porte une requête et retourne les enregistrements correspondants.

- **document**

Cette opération n'est pas documentée mais fera prochainement l'objet d'une note technique rédigée par l'équipe de développement de Connect-It.

Un type de méthode concrète doit respecter l'une des trois signatures publiques suivantes :

- **method(PrgnDocument in) : PrgnMessage**
Opération de type **status**
- **method(PrgnDocument in) : IPrgnXMLResultSet**
Opération de type **resultSet**
- **method(PrgnDocument in) : PrgnDocument**
Opération de type **document**

Autodescription des connecteurs

Un connecteur doit pouvoir s'autodécrire pour permettre à la partie cliente d'obtenir :

- les types de document qu'il peut produire ou consommer
- les opérations qu'il peut effectuer sur ces types de document
- les directives qu'il peut appliquer
- les pointeurs de programmation qu'il utilise

L'autodescription d'un connecteur se fait par l'intermédiaire de l'interface **com.peregrine.conit.document.IPrgnDescribable**. Cette interface, implémentée par la classe **com.peregrine.conit.ra.APrgnDocumentInteraction** définit les méthodes suivantes :

- **getSubFormats(PrgnPath path, String strMode) : PrgnDocument**
- **getOperations(): Vector**
- **getDirectives() : Vector**
- **getPointers() : Vector**

Pour que votre connecteur s'auto-décrive, vous devez rédiger ces méthodes. Le comportement par défaut implémenté par **APrgnDocumentInteraction** est un connecteur sans types de document ni opérations.

L'opération **describe** commune à tous les connecteurs de Connect-It permet ensuite de parcourir cette auto-description.

Types de document publiés - autodescription d'un connecteur Java

Pour définir les types de document publiés par un connecteur Java, vous devez surcharger la méthode **getSubFormat(PrgnPath path, String strMode)** : **PrgnDocument** de la classe **com.peregrine.conit.ra.APrgnDocumentInteraction**.

Le paramètre **strMode** indique quelle description des types de document est voulue. Ce paramètre peut prendre les valeurs suivantes :

- resultSet
ou **com.peregrine.conit.core.IPrgnOperation.RESULTSET** correspondant aux types de document produits.
- status
ou **com.peregrine.conit.core.IPrgnOperation.STATUS** correspondant aux types de document consommés.
- document
ou **com.peregrine.conit.core.IPrgnOperation.DOCUMENT**

 **Note :**

Ce paramètre n'est pas documenté mais fera prochainement l'objet d'une note technique rédigée par l'équipe de développement de Connect-It.

La méthode **getSubFormat** retourne en principe une description à plat des données de l'application externe. La description complète est obtenue à l'aide d'appels successifs.

Quand le paramètre **path** est vide, la méthode doit retourner la liste de types de document disponibles en mode consommation ou production. Pour un connecteur de type Base de données, un paramètre **path** vide peut retourner ce type de liste :

```
<schema>
  <STRUCTURE name="docType1" />
  <STRUCTURE name="docType2" />
  <STRUCTURE name="docType3" />
  <STRUCTURE name="docType4" />
  (...)
</schema>
```

 **Note :**

Le noeud racine du document XML retourné n'est jamais significatif et peut être n'importe quel élément XML valide.

Quand le paramètre **path** n'est pas vide, la méthode doit retourner le contenu d'un sous-noeud du type de document indiqué par la valeur de ce paramètre. Si le chemin est invalide, une exception peut être générée.

Exemple 1

docType2 comme valeur du paramètre path

```
<schema>
  <ATTRIBUTE name="a1" type="Long" />
  <ATTRIBUTE name="a2" type="Byte" />
  <STRUCTURE name="s1" />
  <COLLECTION name="c2" />
  <STRUCTURE name="s2" />
  (...)
</schema>
```

Exemple 2

docType2.s1 comme valeur du paramètre path. Dans ce cas, le document voulu est le sous-noeud **s1** du type de document **docType2**.

```
<schema>
  <ATTRIBUTE name="a5" type="String" />
  <COLLECTION name="c1" />
  (...)
</schema>
```

 **Note :**

- Un **null** peut être retourné si le connecteur n'expose aucun type de document dans le mode demandé.
- La classe **PrgnPath** contient la sous-classe publique **PrgnPathTokenizer** qui permet d'itérer sur les éléments du chemin.

Opérations - autodescription d'un connecteur Java

Une opération est définie par :

- son nom
- son type
- la liste des types de documents pour laquelle elle est disponible

Un connecteur peut exposer plusieurs opérations en mode consommation et en mode production.

Pour exposer les opérations d'un connecteur, vous devez surcharger la méthode **getOperations() : Vector** de la classe **com.peregrine.conit.ra.APrgnDocumentInteraction**.

Cette méthode doit retourner un **Vector** de **com.peregrine.conit.document.description.PrgnOperationDesc**. Si le **Vector** retourné est vide, le connecteur ne dispose d'aucune opération. **Attention** : ce n'est pas le cas pour l'opération **getRecords** des connecteurs événementiels. Le constructeur de la classe **PrgnOperationDesc** a besoin des paramètres suivants :

- un nom d'opération
- un type d'opération

Ce paramètre est une chaîne de caractères et peut être une opération **status** (classe **com.peregrine.conit.core.IPrgnOperation.STATUS**) ou une opération **resultSet** (classe **com.peregrine.conit.core.IPrgnOperation.RESULTSET**).

La méthode **PrgnOperationDesc.addParam(String strParam) : void** permet d'ajouter un paramètre à une opération. Le paramètre **strParam** correspond au nom du type de document sur lequel cette opération s'applique.

Une opération peut s'appliquer à plusieurs types de document. Par défaut, une opération s'applique sur tous les types de document.

 **Note :**

Les opérations de type **status** et **resultSet** s'appliquent respectivement sur des types de document de type **status** et **resultSet**.

Dans l'éditeur de scénarios, la liste des opérations est exposée par le connecteur dans les directives du connecteur.

Directives des opérations - autodescription d'un connecteur Java

Les directives permettent de qualifier les opérations supportées par le connecteur.

Exemple : pour un connecteur de type Base de données, les clauses WHERE et ORDERBY qui lui permettent de filtrer les enregistrements d'une base de données source sont des directives de consommation.

Pour plus d'informations sur la saisie des directives des connecteurs dans l'éditeur de scénarios, consultez le manuel **Connecteurs**, chapitre **Directives des connecteurs**.

Directives et groupes de directives

Les directives sont groupées par fonction. Exemple : groupe des directives de réconciliation.

Définition d'une directive

Une directive est définie par :

- son nom
- son type
- une valeur par défaut facultative
- la liste des types de noeud sur lesquels elle s'applique
- une description facultative qui constitue son libellé dans l'éditeur de scénarios

Une directive peut s'appliquer sur les noeuds du document XML suivants :

- **<STRUCTURE>**

Cet élément correspond à une structure dans un type de document publié par un connecteur.

Le noeud racine (**root**) d'un type de document est de type **<STRUCTURE>** traité de manière différente.

- **<COLLECTION>**

Cet élément correspond à une collection dans un type de document publié par un connecteur.

- **<ATTRIBUTE>**

Cet élément correspond à un champ dans un type de document publié par un connecteur.

Définition d'un groupe de directives

Un groupe de directives est défini par :

- son nom

- une liste de directives
- la liste des opérations qui l'utilisent
- une description facultative qui constitue son libellé dans l'éditeur de scénarios

Les tableaux suivants présentent deux exemples de groupes de directives **FILTER** et **RECONCILIATION** utilisés par des connecteurs de type Base de données.

Nom du groupe	Opération	Directives (2)	Type	Valeur par défaut	Noeud du fichier de déploiement concerné
FILTER	query	WHERE	Memo	Aucune (memo vide)	S'appliquent au noeud racine (Tables de la base de données)
		ORDERBY	Type Chaînes de caractères	Aucune (chaîne de caractères vide)	

Nom du groupe	Opération	Directives (1)	Type	Valeur par défaut	Noeud du fichier de déploiement concerné
RECONCILIATION	insert	RECONCKEY	Booléen	false	S'applique aux noeuds ATTRIBUTE (Champs de la base de données)

Traitement des valeurs des directives

Quand une opération est appelée, les valeurs des directives sont passées dans l'élément **<layer>** du document XML passé comme paramètre.

Exemple

```
<operation>
  <STRUCTURE name="format">
    <ATTRIBUTE name="a1" type="String"/>
    <ATTRIBUTE name="a2" type="Long"/>
  </STRUCTURE>
</operation>
```

```

<STRUCTURE name="s1">
  <ATTRIBUTE name="sa1" type="Boolean"/>
</STRUCTURE>
</STRUCTURE>
<layer>
  <STRUCTURE name="format">
    <dir1>value</dir1>
    <dir2>value</dir2>
    <ATTRIBUTE name="a1" type="String">
      <dir3>value</dir3>
    </ATTRIBUTE>
    <STRUCTURE name="s1">
      <dir2>value</dir2>
      <ATTRIBUTE name="sa1" type="Boolean">
        <dir3>value</dir3>
      </ATTRIBUTE>
    </STRUCTURE>
  </STRUCTURE>
</layer>
</operation>

```

La valeur d'une directive est associée à un noeud spécifique du type de document publié par le connecteur. L'élément **<layer>** reproduit la structure d'un type de document publié par le connecteur sans la valeur de ses champs (éléments **<ATTRIBUTE>**) ni les noeuds sur lesquels aucune directive ne s'applique ou sur lesquels la valeur de la directive est égale à la valeur par défaut de la directive.

Dans la classe **com.peregrine.conit.ra.APrgnDocumentInteraction**, la méthode **getDirectiveValue(PrgnDocument doc, PrgnPath path, String strDirectiveName, int iDirectiveType) : Object** permet d'obtenir la valeur d'une directive.

Le paramètre **doc** doit contenir le document XML passé en paramètre de l'opération.

Le paramètre **path** indique le noeud pour lequel on souhaite obtenir la valeur de la directive. Ce chemin ne doit jamais comprendre le noeud racine du type de document.

Le paramètre **iDirectiveType** indique le type de base de la directive. Pour les directives de type étendu, le type à passer est **PrgnTypes.STRING**.

La classe de l'objet retourné dépend du type de base de la directive. Si aucune valeur de directive n'est trouvée, **null** est retourné.

L'exemple de code suivant permet de retrouver la valeur **dir3** du noeud **Format.s1.sa1**. On suppose que **dir3** a pour type **PrgnTypes.INT**.

Exemple

```

public PrgnDocument operation(PrgnDocument doc) throws ResourceException
{
    (...)
    PrgnPath path = new PrgnPath();
    path.addRight("s1").addRight("sa1");
    int iVal = ((Integer)getDirectiveValue(doc, path, "dir3", PrgnTypes.INT
)).intValue();
    (...)
}

```

Exposition des groupes de directives

Pour exposer les groupes de directives, vous devez surcharger la méthode **getDirectiveGroups() : Vector** de la classe **com.peregrine.conit.ra.APrgnDocumentInteraction**.

Cette méthode retourne un **Vector** de **com.peregrine.conit.document.description.PrgnDirectiveGroupDesc**. Si le **Vector** retourné est vide, aucune directive n'est disponible pour le connecteur.

Le constructeur de la classe **PrgnDirectiveGroupDesc** possède un seul paramètre : le nom du groupe de directives.

Pour définir les opérations utilisant le groupe de directives, servez-vous des méthodes suivantes :

- **addOperation(PrgnOperationDesc desc) : void**
- **addOperation(String strOperationName, String strOperationType) : void**

Pour ajouter une directive dans le groupe, utilisez la méthode **addDirectiveDesc(PrgnDirectiveDesc desc) : void**

Le constructeur de la classe **com.peregrine.conit.document.description.PrgnDirectiveDesc** possède deux paramètres :

- le nom de la directive
- le type de la directive

Le type d'une directive détermine le type de la valeur attendue pour cette directive. Exemple : pour une directive booléenne, les valeurs possibles sont **true** ou **false**. Ce type conditionne également le contrôle graphique d'édition de la directive. Exemple : un champ avec une icône pour une directive de type **nom de fichier**.

Les types de directives possibles sont définis dans la classe **com.peregrine.conit.document.PrgnTypes**.

A ces types, il faut ajouter les types étendus définis dans la classe **com.peregrine.conit.document.description.PrgnDirectiveDesc**:

- Nom de fichier
TYPE_FILE_NAME
 - Nom de dossier
TYPE_DIR_NAME
 - Énumération fermée
TYPE_ENUM
 - Énumération ouverte
TYPE_FREE_ENUM
-

 **Note :**

Le type de base de tous les types étendus est **PrgnTypes.STRING**.

La méthode **setDefaultVal(Object) : void** permet de définir la valeur par défaut de la directive. La classe de l'objet passé en paramètre doit correspondre au type de la directive ou au type de base pour les types de directive étendus. Cette valeur par défaut est optionnelle.

Pour définir les types de noeuds sur lesquels s'applique la directive, utilisez les méthodes suivantes :

- **setApplyOnRoot(boolean bApply) : void**
- **setApplyOnStructures(boolean bApply) : void**
- **setApplyOnCollections(boolean bApply) : void**
- **setApplyOnAttributes(boolean bApply) : void**

Par défaut, une directive ne s'applique sur aucun noeud.

Les méthodes **addPossibleValue(Object) : void** et **setPossibleValues(Collection) : void** permettent de définir :

- les valeurs possibles de la directive si son type est **PrgnDirectiveDesc.TYPE_ENUM** (énumération fermée)
- les valeurs prédéfinies de la directive si son type est **PrgnDirectiveDesc.TYPE_FREE_ENUM** (énumération ouverte).

Pointeurs de programmation - autodescription d'un connecteur Java

Quand un scénario utilise le mode programmé, le pointeur de programmation est une valeur utilisée par les connecteurs source pour éviter de traiter les mêmes données que lors de la dernière session.

La notion de pointeurs de programmation n'est utile que pour les opération de type **resultSet**.

Pour les connecteurs de type Base de données, le pointeur de programmation permet au connecteur de ne traiter que les enregistrements modifiés ou créés depuis la dernière session. Dans ce cas, le pointeur de programmation correspond le plus souvent au champ indiquant la date de dernière modification de l'enregistrement.

Pour les connecteurs traitant des données stockées dans des files d'attente (connecteur MQSeries), le pointeur de programmation permet au connecteur de ne traiter que les ensembles de données créés depuis la dernière session. Ces ensembles de données peuvent être des messages, des documents, des fichiers XML, etc. Dans ce cas, le pointeur de programmation correspond au numéro d'un ensemble de données. **Exemple** : lors d'une première session, un connecteur MQSeries traite les messages numérotés de 0001 à 0999. Lors de la seconde session, il ne traitera que les messages dont le numéro est supérieur à 0999.

Définition du pointeur de programmation

Un pointeur est défini par :

- son type de données
- une valeur par défaut

Cette valeur est celle prise lors du lancement de la première session d'un scénario en mode programmé.

- une liste de types de document

Cette liste regroupe les types de document dont la production dépend de ce pointeur de programmation.

Traitement des valeurs du pointeur

Lorsqu'une opération de type **resultSet** est appelée en mode programmé, la valeur du pointeur est passée comme la valeur d'une directive **pointer**.

Dans la classe **com.peregrine.conit.ra.APrgnDocumentInteraction**, la méthode **getPointerValue(PrgnDocument doc) : Object** permet d'obtenir la valeur du pointeur.

Le paramètre **doc** doit contenir le document XML passé en paramètre de l'opération.

La classe de l'objet retourné dépend du type du pointeur. Dans les deux cas suivants, la valeur **null** est retournée :

- aucun pointeur de programmation n'a été défini
- le scénario n'est pas lancé en mode programmé

Le passage de la nouvelle valeur du pointeur vers Connect-It se fait par l'intermédiaire du compte-rendu de requête. Ce compte-rendu est créé par la méthode **getQueryReport() : IPrgnXMLRecord**. Il se présente sous la forme d'un document XML.

La nouvelle valeur du pointeur est passée dans un noeud **<pointer>** (contenu dans un noeud **<pointers>**) possédant deux attributs :

- **'type'**
Type du pointeur
- **'name'**
Nom du type de document courant

Exemple de compte-rendu de requête

```
<report>
  (...)
  <pointers>
    <pointer name="format" type="String">new pointer value</pointer>
    (...)
  </pointers>
  (...)
</report>
```

Exemple d'implémentation de getQueryReport

```
public IPrgnXMLRecord getQueryReport() throws ResourceException
{
    PrgnDocument report = new PrgnDocument("report");
    (...)
    PrgnDocument pointers = report.createChildDocument(IPrgnTags.POINTERS);
    PrgnDocument pointer = pointers.createChildDocument(IPrgnTags.POINTER);
    pointer.setAttribute(IPrgnTags.ATTR_NAME, "tableName");
    pointer.setAttribute(IPrgnTags.ATTR_TYPE, PrgnTypes.getTypeName(PrgnTypes.LONG));
    PrgnXMLUtil.addTextNode(pointer.getElement(), m_pointerValue.toString());
};
(...)
```

```
return report;
```

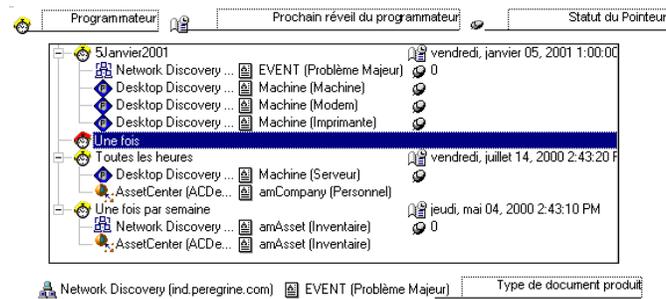
Note :

Si une nouvelle valeur de pointeur est passée dans le compte-rendu et que le scénario n'est pas lancé en mode programmé, cette valeur est ignorée.

Les valeurs des pointeurs de programmation sont enregistrées dans les journaux de Connect-It.

Pour consulter l'état (valeur) d'un pointeur de programmation :

- 1 Lancez l'éditeur de scénarios.
- 2 Ouvrez un scénario ou votre connecteur est en mode production.
- 3 Choisissez (**Scénario/ Programmation**).



Exposition des pointeurs de programmation

Pour exposer les pointeurs de programmation, vous devez surcharger la méthode **getPointers() : Vector** de la classe **com.peregrine.conit.ra.APrgnDocumentInteraction**.

Cette méthode doit retourner un **Vector** de **com.peregrine.conit.document.description.PrgnPointerDesc**. Si le **Vector** retourné est vide, le connecteur n'utilise aucun pointeur de programmation. Le constructeur de la classe **PrgnPointerDesc** possède un seul paramètre : le type du pointeur.

Pour définir les types de document utilisant le pointeur, vous pouvez utiliser les méthodes suivantes :

- **addSchema(String strDocTypeName) : void**
- **addSchemas(Collection cDocTypeNames) : void**

Par défaut, un pointeur de programmation s'applique sur tous les types de document.

Configurer la machine virtuelle Java (JVM)

La JVM utilisée par Connect-It pour les connecteurs Java est directement configurable dans l'éditeur de scénarios.

Pour configurer la JVM

- 1 Lancez l'éditeur de scénarios.
- 2 Choisissez **Java\ Configurer la JVM**.
- 3 Saisissez le chemin de vos classes Java dans le champ **Class path** (fichiers **jar** ou **zip**).
Pour indiquer plusieurs chemins, utilisez le point-virgule (;).
- 4 Saisissez vos options JVM dans le champ **Options**.
Exemple : **-DmyDefine=value**.
- 5 Activez ou non l'option **Mode Debug**.
Cette option permet d'afficher les messages de débogage (**PrgnLog.debug**) et les traces de freinage (**stacktrace**) des exceptions.

Déployer le connecteur Java

Pour déployer un connecteur Java, vous devez fournir un fichier XML de déploiement (souvent **ra.xml**) qui permet le déploiement dynamique de ce connecteur. Ce fichier de description est conforme à la norme JCA ou à l'extension événementielle de cette norme. Pour un exemple de fichier de déploiement, consultez dans ce chapitre, la section [Création d'un connecteur événementiel \[page 261\]](#)/ [Déployer le connecteur Java - connecteur événementiel \[page 263\]](#).

Une fois ce fichier créé, vous pouvez déployer le connecteur dans Connect-It :

- 1 Lancez Connect-It.
- 2 Choisissez **Java/ Déployer un connecteur**.
- 3 Cliquez sur .
- 4 Nommez votre connecteur.

Le champ **Nom** est obligatoire et correspond au nom de votre connecteur dans la boîte à outils.

5 Indiquez le fichier de déploiement de votre connecteur.

6 Choisissez le type de connecteur.

Vous avez le choix entre les types suivants :

- **Événements**

Extension événementielle de la norme JCA.

- **Ressources**

Implémentation de la norme JCA.

7 Choisissez une icône pour votre connecteur dans la boîte à outils.

Cette icône doit avoir pour dimension 16 pixels sur 16 pixels.

8 Indiquez dans le champ **Clé de réconciliation** la directive du connecteur permettant d'afficher les clés de réconciliation dans la fenêtre d'édition des types de document consommés.

Exemple : pour le connecteur JDBC fourni, cette directive est la directive **RECONCKEY**.

Pour une présentation des directives d'un connecteur Java, consultez dans cette section, la section [Autodescription des connecteurs](#) [page 247], [Directives des opérations - autodescription d'un connecteur Java](#) [page 250].

9 Sélectionnez les options de votre choix.

- **Mode consommation**

Sélectionnez cette option si le connecteur supporte des opérations de consommation des documents (écriture, envoi, etc.)

- **Mode production**

Sélectionnez cette option si le connecteur supporte des opérations de production des documents (lecture, réception, etc.)

- **Utiliser les pointeurs de programmation**

Sélectionnez cette option si le connecteur utilise les pointeurs de programmation.

Pour une présentation de l'implémentation des pointeurs de programmation pour les connecteurs Java, consultez dans cette section, la section [Autodescription des connecteurs](#) [page 247] \ [Pointeurs de programmation - autodescription d'un connecteur Java](#) [page 256].

- **Le connecteur émet des requêtes**

Sélectionnez cette option si le connecteur émet des requêtes à destination d'autres connecteurs. Exemple : Exemples : Connecteurs d'écoute XML et connecteur d'écoute SCAuto. Cette option n'est disponible que pour les connecteurs événementiels.

Création d'un connecteur événementiel

Cette section vous présente les différentes étapes de création d'un connecteur événementiel.

Définir le contrat de transaction - connecteur événementiel

L'interface **com.peregrine.resourceimpl.event.IPrgnEventAdapter** est le point d'entrée majeur de l'architecture du connecteur. Il prend en charge les rôles de la classe **ManagedConnectionFactory** concernant les contrats de connexion et de sécurité et utilise les méthodes **getConnection()**. Par conséquent, une implémentation concrète doit fournir les méthodes de collecte (get) et d'initialisation (set) pour les paramètres requis lors de la création de la connexion correspondant aux paramètres de configuration définis dans le fichier de déploiement.

L'implémentation fournie par défaut est **com.peregrine.resourceimpl.event.APrgnEventAdapter**. Cette classe abstraite implémente l'interface **Connection** et permet la manipulation de la connexion. Le contrôle du traitement de l'événement est ajouté par les fonctions **start()** et **stop()** sur une connexion événementielle.

Quand un client provoque une connexion (méthode **getConnection()**), un listener d'événements spécifique de l'application externe est créé dynamiquement par la méthode **createSourceListener()**. L'objet créé **com.peregrine.resourceimpl.event.APrgnSourceListener** se comporte comme un **java.lang.Thread** et effectue les tâches suivantes de l'application externe :

- Initiation de la connexion
Méthodes **connect()** et **init()**
- clôture de la connexion
Méthodes **disconnect()** et **shutdown()**

- Ecoute d'un événement entrant provenant de l'application externe et production d'un enregistrement XML.

Méthode **listen()**

- Notification des événements de connexion

Méthodes **addConnectionHandler()** et **fireConnectionClosed()**

Comme la méthode **getMetaData()** de **ManagedConnectionFactory**, la méthode **getEventAdapterMetaData()** renvoie les métadonnées du connecteur. Cette méthode ne requiert pas qu'une connexion active soit établie avec l'application externe.

Définir les données - connecteur événementiel

Un connecteur de type événementiel supporte également la norme **Unified Document Content**.

La définition des données pour un connecteur événementiel est la même que pour les autres connecteurs Java. Pour plus d'informations sur cette définition des données, consultez dans ce chapitre, la section [Création d'un connecteur Java](#) [page 241] \ [Définir les données](#) [page 245].

Définir l'interaction sur les données - connecteur événementiel

Un autre rôle de **IprgnEventAdapter** est l'enregistrement de l'objet cible des événements et des requêtes produits. Cet objet, implémentant l'interface **com.peregrine.resourceimpl.event.IPrgnRecordHandler**, peut consommer un fichier XML sans produire de réponse (réception d'événements) ou agir comme interaction de rappel renvoyant un enregistrement de sortie à partir d'un enregistrement entrant (requête).

Le routage d'un événement XML vers son référencieur enregistré est effectué via la méthode **postRecord()**. Les requêtes sont effectuées via les méthodes **pullExecute()** et **pullResultSet()**.

L'écriture de méthodes d'interaction personnalisées est également permise dans le mode événementiel. Cependant ces méthodes font partie d'un fonctionnement en mode Requête/Réponse : production ou consommation de documents sur demande du client.

Pour plus d'informations sur l'interaction sur les données, consultez dans ce chapitre, la section [Création d'un connecteur Java \[page 241\]](#) \Définir l'interaction sur les données [page 245].

Autodescription - connecteur événementiel

L'autodescription d'un connecteur événementiel est identique à celle des autres connecteurs Java.

Pour plus d'informations sur l'autodescription des connecteurs, consultez dans ce chapitre, la section [Création d'un connecteur Java \[page 241\]](#) \Autodescription des connecteurs [page 247].

Un connecteur événementiel n'a pas besoin d'exposer d'opération pour produire des événements ou émettre des requêtes. Les seules opérations à exposer sont des opérations personnalisées.

Le format des événements produits par un connecteur événementiel correspond aux types de document qu'il produit (mode **resultSet**).

Un connecteur événementiel ne consomme a priori aucun type de document sauf s'il supporte des opérations personnalisées de type **status**.

Déployer le connecteur Java - connecteur événementiel

Pour déployer un connecteur Java événementiel, vous devez fournir un descripteur XML (Fichier **ra.xml**) qui permet un déploiement dynamique de ce connecteur.

Ce descripteur est proche des spécifications de la norme JCA.

```
<?xml version="1.0" encoding="UTF-8"?>
<connector>
  <display-name> My Event Adapter< /display-name>
  <description> Event Adapter< /description>
  <vendor-name> Peregrine Systems< /vendor-name>
  <eis-type> My EIS Type< /eis-type>
  <version> 1.0< /version>
  <license>
    <description> < /description>
    <license-required> true< /license-required>
  </license>
  <resourceadapter>
    <eventadapter-interface> com.peregrine.resourceimpl.event.IPrgnEventA
dapter< /eventadapter-interface>
    <eventadapter-class> MyEventAdapterClass< /eventadapter-class>
    <config-property>
```

```

<description> < /description>
<config-property-name> MyProperty< /config-property-name>
<config-property-type> MyPropertyType< /config-property-type>
<config-property-value> MyPropertyValue< /config-property-value>
</config-property>
...
</resourceadapter>
</connector>

```

Création d'un connecteur JCA - exemple

Cette section présente les différentes étapes de création d'un connecteur d'exemple de type JDBC.

Vous trouverez les sources du connecteur d'exemple Java dans le dossier suivant :

- **[Dossier d'installation de Connect-It]\datakit\javasample\com\peregrine\sample\ra\jdbc**

Définir le contrat de connexion - connecteur d'exemple

Les paramètres de connexion du connecteur sont :

- Driver
Le pilote JDBC utilisé pour accéder au SGDB relationnel.
- URL
Doit respecter la forme suivante : <Subprotocol> : <DatabaseName>
- UserName
Première information de sécurité utilisée pour créer la connexion.
- Password
Deuxième Information de sécurité utilisée pour créer la connexion.

Ces paramètres permettent d'étendre la classe

com.peregrine.resourceimpl.spi.APrgnManagedConnectionFactory.

```

public class JDBCManagedConnectionFactory extends APrgnManagedConnectionF
actory
{
// La classe doit être publique pour être déployée par un outil externe.
}

```

Cette partie du code permet d'obtenir les méthodes publiques de collecte (get) et d'initialisation (set) des données pour les paramètres de connexion.

```

/**
 * Sets the JDBCManagedConnectionFactory URL property.
 */
public void setURL(String strURL)
{
}

/**
 * Obtient la propriété JDBCManagedConnectionFactory.
 */
public String getURL()
{
}

```

Vous devez ensuite dériver la classe **com.peregrine.resourceimpl.spi.APrgnManagedConnection** pour rendre compte de la connexion physique.

Définir le contrat de transaction - connecteur d'exemple

Le connecteur d'exemple ne supporte pas la démarcation de transactions.

La méthode **getResourceMetaData** de la classe **ManagedConnectionFactory** doit retourner l'objet **ResourceMetaData** avec la propriété **supportsLocalTransactionDemarcation** initialisé à **false**.

Les méthodes suivantes de **ManagedConnection** doivent générer un **NotSupportedException** :

- **getLocalTransaction()** : **LocalTransaction**
- **getXAResource()** : **XAResource()**
- **getAutoCommit()** : **boolean**
- **setAutoCommit(boolean b)** : **void**

Définir le contrat de sécurité - connecteur d'exemple

Le mécanisme de sécurité est basé sur une authentification **Nom d'utilisateur/Mot de passe**.

Les méthodes liées à la sécurité de **ManagedConnectionFactory** sont :

```

createManagedConnection(,PasswordCredential pc) : APrgnManagedConnection
matchPasswordCredential(APrgnManagedConnection managedCnx, PasswordCredent
tial pc) : boolean

```

La réauthentification n'est pas supportée.

Les méthodes liées à la sécurité de **ManagedConnection** sont :

```
newConnectionHandle(Subject subject, PrgnConnectionRequestInfo cnxRequestInfo) : APrgnConnectionHandle
```

Définir les données - connecteur d'exemple

Le connecteur d'exemple opère des SELECT sur des applications externes de type Base de données relationnelle.

Comme l'interaction sur les données est basée sur la technologie XML, une opération consomme un document de la forme suivante :

```
<query>
  <STRUCTURE name="Employee">
    <ATTRIBUTE name="LastName">
      <ATTRIBUTE name="FirstName">
    </STRUCTURE>
</query>
```

Il résulte de ce document la déclaration SQL suivante : `SELECT LastName,FirstName FROM Employee`. Cette déclaration retourne un **XMLResultSet** opérant sur les enregistrements de la base.

Dans l'implémentation **ManagedConnectionFactory**, il faut ajouter la méthode :

```
public ResourceAdapterMetaData getMetaData() throws javax.resource.ResourceException
{
  // Doit retourner une instance de PrgnResourceAdapterMetaData
  PrgnResourceAdapterMetaData raMetaData = new PrgnResourceAdapterMetaDat
a();

  return raMetaData;
}
```

Définir l'interaction sur les données - connecteur d'exemple

Le code dérive **APrgnDocumentInteraction** et écrit la méthode publique suivante :

```
query(PrgnDocument document) : IPrgnXMLResultSet
```

Création d'un assistant de configuration amélioré

Le fichier **ra.xml** vous permet de déployer le connecteur dans l'éditeur de scénarios de Connect-It. Il permet également de générer un assistant de configuration par défaut très simple.

Pour plus d'informations sur l'utilisation des assistants de configuration des connecteurs, consultez le manuel **Connecteurs**, chapitre **Configuration des connecteurs**.

Cette section vous permet de créer un fichier DSD améliorant l'assistant par défaut afin que celui-ci intègre des fonctions complexes :

- Listes à choix multiples
- Propositions de valeurs par défaut
- Boutons radio
- Boutons de navigation dans les dossiers du système d'exploitation
- Zones de saisie cryptée pour les mots de passe
- Etc.

Avertissement :

Un fichier DSD invalide peut entraîner l'échec d'exécution du connecteur.

Les scénarios existants créés avant l'introduction du fichier DSD se lanceront correctement. Par contre, un scénario enregistré avec un fichier DSD ne se lancera pas si le fichier DSD est supprimé.

Convention : dans les exemples de code de cette section, le rouge signale les éléments toujours présents dans une DSD. Ces éléments ne varient pas.

Le bleu signale les attributs obligatoires.

Création du fichier DSD

Le fichier DSD doit être inséré dans le dossier contenant le fichier **ra.xml** de votre connecteur.

Exemple : si votre connecteur est nommé **MyCon**, le déploiement du connecteur crée un dossier correspondant dans le dossier d'installation de Connect-It : **[Dossier d'installation de Connect-It]//config/java_MyCon**.

Le fichier DSD doit porter le même nom que le fichier de déploiement avec pour nom d'extension de fichier DSD.

Exemple : votre fichier de déploiement s'appelle **MyCon.xml**, le fichier DSD doit s'appeler **MyCon.dsd**.

Contenu du fichier DSD

Un fichier DSD doit contenir les cinq noeuds suivants:

- Un noeud **<DSD>**
Ce noeud possède un attribut '**rootnode**' et englobe les quatre noeuds suivants.
- Un noeud **<module>**
Ce noeud est identique dans tous les fichiers DSD.
- trois noeuds **<node>** obligatoires :
 - un premier noeud **racine** qui contient la déclaration des propriétés de configuration.
La valeur de l'attribut '**name**' de ce noeud doit être égale à celle de l'attribut '**rootnode**' du noeud **<DSD>**.
Les attributs '**class**' et '**edit**' sont obligatoires et doivent avoir pour valeur respective "**Folder**" et "**CWBAutoDet**".
 - deux noeuds **default** et **advanced** dont les attributs '**name**' ont pour valeur respective "**default**" et "**advanced**".
Ces deux noeuds définissent la page de l'assistant en mode simple et en mode avancé. L'attribut '**class**' de ces noeuds doit avoir pour valeur celle de l'attribut '**name**' du noeud **racine**.

Exemple

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<DSD name="Wiz" rootnode="ConWiz" version="1.0" warning="true">
  <!-- You must specify the root node -->
  <module name="core.dsd" local="true"/>
  <!-- This module node is mandatory -->
  <node name="ConWiz" class="Folder" edit="CWBAutoDet">
    (...)
  </node>
  <node name="default" class="ConWiz">
    (...)
  </node>
  <node name="advanced" class="ConWiz">
    (...)
  </node>
</DSD>
```

Si le noeud **advanced** ne diffère pas du noeud **default**, vous pouvez le simplifier en le faisant de la manière suivante :

```
<node name="advanced" class="default" />
```

 **Avertissement :**

L'attribut '**name**' doit toujours être le premier attribut d'une balise.

Définition des propriétés de configuration

Le noeud racine doit comporter un sous-noeud par propriété de configuration du connecteur. Ce sous-noeud porte le même nom utilisé dans le fichier de déploiement (**ra.xml**).

 **Avertissement :**

Toutes les propriétés définies dans le noeud racine doivent être présentes dans le fichier de déploiement.

Attributs obligatoires

Chaque propriété de configuration a deux attributs obligatoires : '**name**' et '**class**'.

Les classes supportées sont :

- **Enum** (énumération fermée)
- **FreeEnum** (énumération ouverte)
- **Boolean** (booléen)
- **Long** (entier long)
- **Double** (entier double)
- **Date** (date/heure)
- **Duration** (durée)
- **String** (chaîne de caractère)
- **Password** (crypté)
- **FileName** (nom de fichier)
- **DirName** (nom de dossier)
- **Memo** (texte multiligne)

Attributs optionnels

Chaque propriété de configuration a trois attributs optionnels :

- **'mandatory'**
Booléen
- **'inshortdesc'**
Si ce booléen a pour valeur **true**, la valeur du paramètre est utilisée dans la description sommaire du connecteur.
- **'default'**
Cet attribut définit la valeur par défaut des propriétés de configuration. Si les attributs **'inshortdesc'** ou **'mandatory'** ne sont pas définis, leur valeur par défaut est **false**.

Propriétés de configuration

Chaque propriété de configuration a deux sous-noeuds :

- **<shortdesc>**
Cette propriété définit la description du champ apparaissant dans l'assistant de configuration quand il est lancé dans l'éditeur de scénarios. Cette propriété est obligatoire.
- **<longdesc>**
Cette propriété définit le texte d'aide du connecteur. Cette propriété est optionnelle.

Exemple

```
<node name="ConWiz" class="Folder" edit="CWBAutoDet">
  <node name="field1" class="String" default="test" mandatory="true" inshortdesc="true">
<shortdesc>Field 1</shortdesc>
<longdesc>Field 1's help</longdesc>
  </node>
  <node name="field2" class="Boolean" default="true" mandatory="false" inshortdesc="false">
<shortdesc>Field 2</shortdesc>
<longdesc>Field 2's help</longdesc>
  </node>
  <node name="field3" class="Long" default="0">
<shortdesc>Field 3</shortdesc>
<longdesc>Field 3's help</longdesc>
  </node>
  <node name="field4" class="Double" default="1.52">
  <!-- Double nodes have a precision of two digits after the decimal point
.-->
```

```

<shortdesc>Field 4</shortdesc>
<longdesc>Field 4's help</longdesc>
</node>
  <node name="field5" class="Date" default="yyyy-mm-dd">
<shortdesc>Field 5</shortdesc>
<longdesc>Field 5's help</longdesc>
</node>
  <node name="field6" class="Duration" default="1y 2mon 3d 4h 5m 6s">
<shortdesc>Field 6</shortdesc>
<longdesc>Field 6's help</longdesc>
</node>
  <node name="field7" class="Password">
<shortdesc>Field 7</shortdesc>
<longdesc>Field 7's help</longdesc>
</node>
  <node name="field8" class="FileName" default="c:/tmp/test.txt">
<shortdesc>Field 8</shortdesc>
<longdesc>Field 8's help</longdesc>
</node>
  <node name="field9" class="DirName" default="c:/tmp">
<shortdesc>Field 9</shortdesc>
<longdesc>Field 9's help</longdesc>
</node>
  <node name="field10" class="Enum" default="op1">
<shortdesc>Field 10</shortdesc>
<longdesc>Field 10's help</longdesc>
<enum name="op1" value="0"><caption>Option 1</caption></enum>
<!-- The value of an enum is equal to the name attribute value-->
<enum name="op2" value="1"><caption>Option 2</caption></enum>
</node>
  <node name="field10" class="FreeEnum" default="Option 1" enum="Option 1 |
Option 2">
  <!--Values are separated by | -->
<shortdesc>Field 10</shortdesc>
<longdesc>Field 10's help</longdesc>
</node>
  <node name="field11" class="Memo">
<shortdesc>Field 11</shortdesc>
<longdesc>Field 11's help</longdesc>
<default>this is a multiline default value</default>
<!-- In the case of a Memo, the default value is set in a default node.-->
  >
</node>
</node>

```

Page de configuration

Les noeuds **default** et **advanced** englobent l'élément **<layout>**. Cette élément définit l'apparence de la page **Déploiement du connecteur** de l'assistant de configuration de votre connecteur Java.

Une page peut être organisée en cadres. Chaque cadre regroupe les propriétés d'une même famille. Exemple : un cadre consacré aux propriétés de connexion.

Chaque élément **<frame>** :

- est défini par son nom
- contient un élément **<caption>** qui permet de légender le cadre.

Chaque élément **<frame>** contient les références des propriétés de configuration à afficher.

Si une propriété de configuration n'est référencée dans aucun cadre, il est impossible de paramétrer sa valeur dans l'assistant. Si les propriétés n'apparaissent que dans le mode avancé, vous ne les référencez pas dans le noeud **default** mais dans le noeud **advanced**.

Exemple

```
<node name="default" class="ConWiz">
  <layout>
    <frame name="frame1">
      <caption>Frame #1</caption>
      <attribute name="field1"/>
      <attribute name="field2"/>
      <attribute name="field5"/>
    </frame>
    <frame name="frame2">
      <caption>Frame #2</caption>
      <attribute name="field6"/>
    </frame>
    <frame name="frame3">
      <caption>Frame #3</caption>
      <attribute name="field3"/>
      <attribute name="field4"/>
    </frame>
  </layout>
</node>
```

DSD - problèmes courants

Voici une liste des problèmes pouvant invalider un fichier DSD :

- Balises de fin manquantes
- Valeurs d'attributs invalides
- Attribut **'name'** non déclaré comme premier attribut de la balise
- Noeud **racine** non spécifié
- Classe inconnue utilisée
- Noeuds **default** ou **advanced** absents ou n'héritant pas du noeud **racine**.

- Attribut inconnu référencé
- Propriété définie par le noeud **racine** n'existant pas pas dans le fichier de déploiement (**ra.xml**)

A | Comment signaler un problème

ANNEXE

Pour nous aider à améliorer constamment la qualité de nos produits, il est primordial que vous reportiez au support de Peregrine Systems les problèmes rencontrés lors de leur utilisation. Pour que ces problèmes soient identifiés et résolus dans les meilleurs délais, il est impératif que certaines informations précises nous soient fournies. Bien souvent, en effet, les informations mises à notre disposition sont insuffisantes pour reproduire le problème, et à fortiori pour le corriger.

L'objectif de ce document est de vous aider à collecter les informations dont nous avons besoin et à reporter le problème de la façon la plus optimale possible.

Informations générales

Voici les informations dont nous avons besoin :

- [Informations fonctionnelles](#) [page 276]
- [Informations techniques](#) [page 276]

Informations fonctionnelles

Ces informations nous permettent de catégoriser le problème :

- Le nom de votre société,
- Le produit concerné par le problème,
- Le domaine fonctionnel associé au problème (par exemple, l'éditeur de scénarios dans Connect-It, la gestion des achats dans AssetCenter, etc.)
- La fréquence du problème (le problème peut être reproduit de façon systématique, sa reproduction dépend de certains facteurs ou de certaines données, etc.)

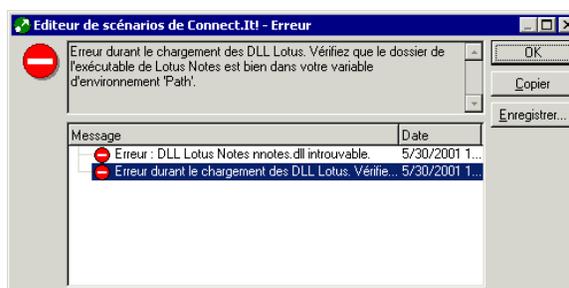
Informations techniques

Une partie des informations techniques essentielles sont directement accessibles dans nos produits. Pour récupérer ces informations :

- 1 Sélectionnez le menu **Aide/ A propos**,
- 2 Dans la boîte de dialogue qui s'affiche, cliquez sur le bouton **Plus**. Une liste des informations techniques sur votre système et sur le produit s'affiche.
- 3 Cliquez sur le bouton **Copier**. Les informations sont alors copiées dans le presse-papiers de Windows.
- 4 Collez ces informations dans un nouveau fichier et assurez-vous de l'envoyer au support de Peregrine Systems.

Si le problème que vous reportez se traduit par un message d'erreur envoyé par l'application, effectuez les opérations suivantes :

- 1 Lorsque le message d'erreur s'affiche (exemple ci-dessous), cliquez sur le bouton **Copier**. Les informations relatives au problème sont alors copiées dans le presse-papiers de Windows.



 **Note :**

Cette capture d'écran est donnée à titre d'exemple. Les informations du message d'erreur ainsi que le nom du logiciel concerné seront probablement différents dans votre cas.

- 2 Collez ces informations dans un nouveau fichier et assurez-vous de l'envoyer au support de Peregrine Systems.

Informations spécifiques au problème

La première chose dont nous avons besoin est une liste contextuelle des étapes qui permettent de reproduire le problème. Ce dernier peut dépendre de certains facteurs; veuillez donc à les préciser dans votre descriptif. Par exemple :

- 1 Sélectionnez le bien Peregrine Prosigma 850 dans la table des biens sous AssetCenter. Ce bien fait partie de la base de démonstration livrée avec le produit.
- 2 Cliquez sur l'onglet **Acquis.** du détail de ce bien,
- 3 Cliquez sur le sous-onglet **Approvisionnement** de l'onglet **Acquis.**
- 4 Changez la valeur du champ **Valeur d'achat** en 123.45
- 5 Cliquez sur **Modifier**,
- 6 etc...

 **Astuce :**

En règle générale, essayez de circonscrire au maximum le problème reporté.

Autres informations utiles

Si vous le souhaitez, vous pouvez nous envoyer des captures d'écran ou des fichiers additionnels qui nous aideront à mieux identifier le problème.

 **Note :**

N'envoyez pas de fichiers trop volumineux par e-mail. Limitez-vous à 500 Ko pour les attachements. Si nécessaire, nos équipes de support peuvent nous ouvrir l'accès à un ftp pour déposer des fichiers qui dépassent les 500 Ko.

Quelques pistes pour limiter la taille des fichiers envoyés :

- Pensez à utiliser un outil de compression. Dans bien des cas, on peut observer une réduction significative de la taille des fichiers.
- Pour les images, vous pouvez réduire le nombre de couleurs à 16. L'image prend alors moins de place.

Si vous signalez un problème relatif à Connect-It

Connect-It tient systématiquement des journaux des documents, un journal de l'application, et un journal des services. Les informations contenues dans ces fichiers de journal peuvent être très utiles pour reproduire un problème.

Ces fichiers sont stockés dans les répertoires suivants :

- **bin32\conitgui.log** pour le fichier journal de l'application,
- **bin32\ pour le fichier journal d'un service Connect-It,**
- Fichiers **.idx**, **.dat** et **.msg** pour les fichiers journaux associés à des scénarios. Le nom de ces fichiers est configurable par scénario à partir de l'interface graphique de Connect-It



Note :

Pensez à nous envoyer ces fichiers ainsi que le fichier de scénario (**.scn**) quand vous reportez un problème sur Connect-It

B | Glossaire

ANNEXE

Ce glossaire n'est pas exhaustif mais présente les principaux termes utilisés dans Connect-It. Vous y trouverez également une liste de termes transversaux utilisés dans le langage base de données, script Basic, Java, et application externe.

Termes Connect-It

Connecteur

Un connecteur permet à Connect-It de traiter des données provenant ou allant vers des applications externes. Afin d'échanger des données entre eux, les connecteurs les transcrivent en documents qu'ils peuvent produire ou consommer.

Les connecteurs se répartissent en plusieurs catégories :

- Connecteurs inventaires

Ces connecteurs traitent des données provenant de bases de données dont le rôle est d'inventorier les biens d'une entreprise et plus particulièrement

leur parc informatique : connecteur Intel LanDesk, connecteur SMS 1.x et SMS 2.x...

- Connecteurs Peregrine Systems

Ces connecteurs sont dédiés à des applications Peregrine Systems (connecteur Asset Management, connecteur ServiceCenter, connecteur InfraTools Management, connecteur InfraTools Desktop Discovery, Action Request System...)

- Connecteurs applicatifs

Ces connecteurs sont dédiés à des applications externes données : connecteur Lotus Notes, connecteur Sécurité NT, etc.

- Connecteur de type protocole

Ces connecteurs traitent des données formatées par des applications externes à l'aide d'un protocole particulier : connecteur XML, connecteur texte délimité, connecteur Base de données

- Connecteurs Java

Ces connecteurs sont développés en Java. En utilisant le kit de développement Java de Connect-It, vous pouvez créer vos propres connecteurs.

Pour plus d'informations sur la création de connecteurs Java, consultez le chapitre [Kit de développement Java de Connect-It](#) [page 229].

Le rôle d'un connecteur consiste à :

- Publier les types de document correspondant aux ensembles de données disponibles dans l'application externe à laquelle il est connecté.

Exemple :

Le connecteur e-mail (réception) publie un type de document correspondant à un e-mail comprenant, entre autres, une structure **From** (auteur), une collection **Copie carbone** et une collection **Attachment** (pièce jointe).

- Produire ou consommer des documents.

Pour produire un document, un connecteur s'appuie sur un type de document produit qui a été créé pour un scénario d'intégration précis.

Exemple :

Dans chacun des scénarios fournis entre **InfraTools Desktop Discovery vers Asset Management** (dossier **scenario/idd**) le connecteur Desktop Discovery produit des documents qui s'appuient sur le type de document publié **Machine**. Ce type de document correspond aux valeurs obtenues lorsque l'application externe InfraTools Desktop Discovery lance un inventaire sur un parc informatique.

Pour consommer un document, un connecteur extrait les valeurs d'un document provenant d'une boîte de mapping et les transmet à une application externe.

Quand les données proviennent d'une application externe source et se dirigent vers une application externe destination, les connecteurs sont appelés connecteur source et connecteur destination. Un connecteur source produit des documents alors qu'un connecteur destination en consomme.

Document et type de document

Un document est le support par lequel les données circulent dans Connect-It. Chaque document correspond à un ensemble de données d'une application externe. En fonction de l'application externe, un ensemble de données correspond à :

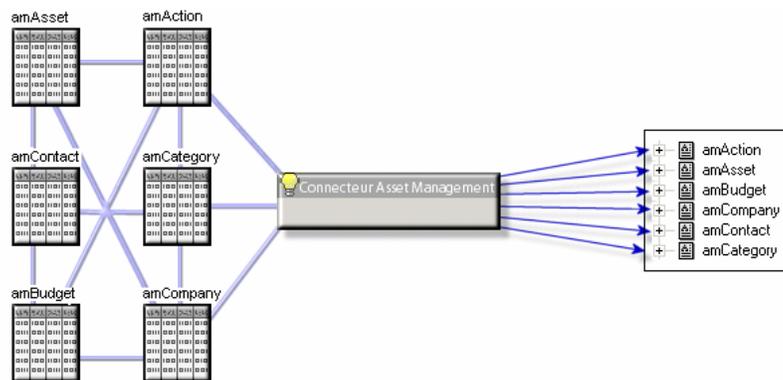
- une table de base de données
- une message e-mail
- un fichier contenant des informations d'inventaire
- un fichier texte délimité
- un fichier XML
- un ensemble d'informations de sécurité
- etc.

Quand un connecteur est ouvert dans l'éditeur de scénarios, il publie une liste de types de document. La liste des types de documents publiés correspond à tous les ensembles de données disponibles dans les applications externes.

Exemple :

Les différentes tables d'une base de données AssetCenter correspondent à tous les types de document publiés par le connecteur Asset Management dans l'éditeur de scénarios.

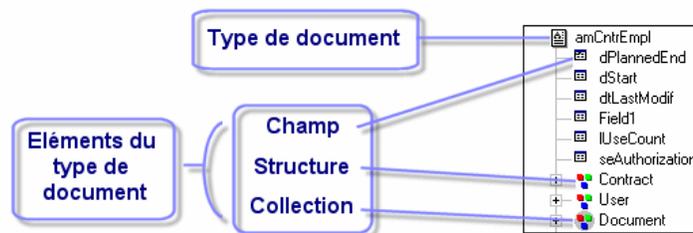
Figure B.1. Connecteur Asset Management - Types de documents publiés



Éléments d'un type de document

Chaque type de document se présente sous la forme d'une arborescence composée d'éléments simples terminaux (champs) et d'éléments complexes déplaçables : noeud racine, structures et collections contenant d'autres éléments simples et complexes.

Figure B.2. Éléments d'un type de document



Les éléments simples sont :

- les champs (📄)
Ils contiennent un type de données particulier : des nombres, des nombres entiers longs, des textes de longueur déterminée, des dates...
- les champs de type blob (📄)
Ils contiennent des objets binaires dont le contenu doit être préservé (binary large object) : des images, des fichiers son, vidéo...
- les champs de type memo (📄)
Ils contiennent des textes de longueur variable.

Les éléments complexes sont :

- les structures (📄) composées d'éléments simples ou complexes
Exemple 1 : dans un type de document correspondant à une table de base de donnée, chaque structure correspond à une autre table. Une structure indique que le lien entre les deux tables est de type 1:1 : un enregistrement de la première table ne peut être lié qu'à un et un seul enregistrement de l'autre table.
Exemple 2 : dans un type de document correspondant à un message e-mail reçu, une structure correspond aux informations relatives à l'auteur du message (un message ne peut avoir qu'un seul auteur).
- les collections (📄), composées d'éléments simples ou complexes
Exemple 1 : quand un type de document correspondant à une table de base de données comprend une collection, cette collection correspond à une autre table de la base de données. Une collection indique que le lien entre les deux tables est de type 1:N ou N:N : un enregistrement de la première table ne peut être lié qu'à un ou plusieurs enregistrements de l'autre table et inversement.
Exemple 2 : dans un type de document correspondant à un message e-mail envoyé, une collection correspond aux destinataires du message (un message peut avoir un nombre illimité de destinataires).

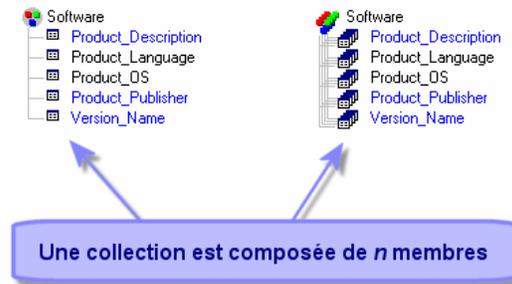
 **Note :**

Les collections d'attributs sont des collections comprenant un seul champ. Ce champ est un attribut. Chaque membre de cette collection est composé d'une structure à un élément se répétant **n** fois.

Avant le traitement des documents par un connecteur, chaque collection présente dans les documents contient un nombre indéterminé de membres.

Le nombre de ces membres ne pouvant être connu à l'avance, une collection est composée de n membres.

Figure B.3. Collection d'un type de document



Exemple :

Exemple : dans le type de document **Machine** publié par le connecteur Desktop Discovery, les logiciels installés sur un ordinateur sont représentés par une collection car un nombre indéterminé de logiciels sont installés sur un ordinateur. Chaque membre de cette collection contient la description d'un logiciel (nom, éditeur, version...).

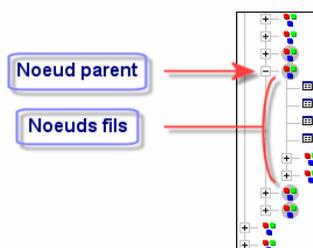
Noeud parent et noeuds fils dans un type de document

Dans une arborescence, chaque élément dépliable comporte des éléments fils. Dans la description d'un type de document, on parle d'un noeud parent contenant des noeuds fils.

 **Note :**

Durant le traitement des documents, les notions de parentés sont importantes. On évoque, par exemple, le rejet d'un champ contenu sous un noeud fils ayant entraîné le rejet du noeud parent.

Figure B.4. Arborescence d'un type de document



Types de document produits et types de document consommés

Pour qu'un connecteur produise ou consomme des documents, il faut que le connecteur s'appuie sur des types de document produits et des types de documents consommés. Pour créer un type de document produit ou un type de document consommé, l'utilisateur de Connect-It opère une sélection parmi les éléments d'un type de document publié par le connecteur. Exemple : un type de document produit par un connecteur de type base de données comporte les seuls champs que l'utilisateur veut importer dans une application externe destination.

Type de document publié Type de document produit ou consommé Document produit

amAsset	
AccelCode	Texte
AssetTag	Texte
BarCode	Texte
bCreatedOnTheFly	Nombre entier 16 bits
bisCnxClient	Nombre entier 16 bits
Brand	Texte
Comment	Champ texte long
dAccept	Date

amAsset	
AssetTag	
Brand	
mPrice	
Category	
Name	

amAsset	Hewlett Packard
AssetTag	PRNLND1006
Brand	Hewlett Packard
Category	C000007
BarCode	C000007
blnvent	1

Le connecteur publie les types de document. Ci-dessus, un extrait du type de document amAsset publié par le connecteur Asset Management.

Créer un type de document produit ou consommé revient à sélectionner certains éléments d'un type de document publié.

Le connecteur produit ou consomme un document où les éléments du type de document produit ou consommé prennent des valeurs.

Fichiers XML et DTD

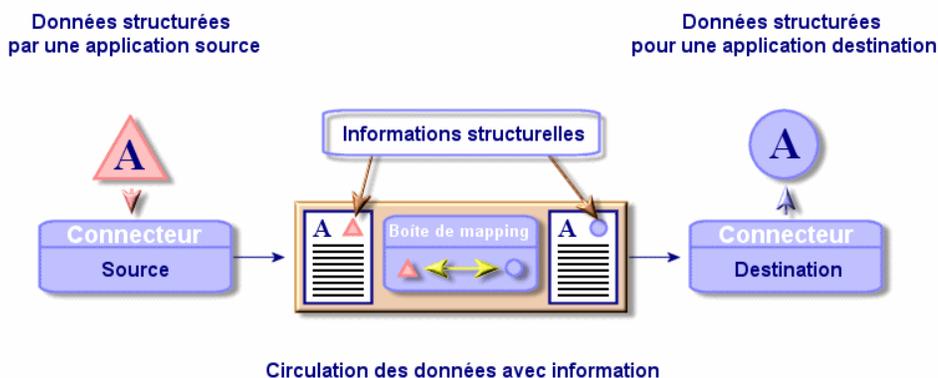
Les documents utilisés dans Connect-It correspondent à des fichiers XML (Extensive Markup Language) et les types de document sur lesquels ils s'appuient à des DTD (Document Type Definition).

L'éditeur de scénarios de Connect-It vous permet de transformer tous les documents en fichiers XML et tous les types de document en DTD.

Mapping

Un mapping consiste à établir des correspondances et des transformations entre les éléments d'un type de document produit par un connecteur source et un type de document consommé par un connecteur destination.

Dans Connect-It une boîte de mapping permet à un connecteur destination de consommer les documents produits par un connecteur source. La boîte de mapping est un outil interne qui produit et consomme des documents comme tous les autres connecteurs mais qui ne communique pas avec les applications externes.



Scénario d'intégration

Un scénario d'intégration permet de faire circuler des données entre différentes applications externes.

L'écriture d'un scénario consiste à :

- 1 Choisir des connecteurs source et destination

- Le nombre de ces connecteurs est en principe illimité.
- 2 Configurer ces connecteurs
 - 3 Lier ces connecteurs
 - 4 Créer des types de document produits pour le connecteur source et des types de document consommés pour le connecteur destination
 - 5 Créer des mappings entre des types de documents source et des types de document destination
 - 6 Choisir une programmation
- Cette programmation établit la périodicité avec laquelle les connecteurs source du scénario produisent des documents.
- Connect-It est livré avec des scénarios d'exemple.

 **Avertissement :**

Les scénarios d'exemple ne s'appliquent pas, la plupart du temps, à des configurations utilisateur spécifiques.

Directives

Les directives sont des informations qu'utilisent les composants de Connect-It pour produire et consommer des documents. Les directives se divisent en **directives de production** et **directives de consommation**.

Journal des documents

Le journal des documents est un onglet dans l'éditeur de scénarios permettant de prendre connaissance du contenu de tous les documents produits et consommés par les connecteurs et les autres composants de Connect-It (liens, boîtes de mapping).

Quand un connecteur rejette un document, le journal de document vous le signale en accompagnant ce document d'une icône significative. Le journal des documents vous avertit également quand un document n'a pu être produit ou consommé de manière parfaite par un connecteur ou une boîte de mapping.

Programmation

La programmation dans Connect-It consiste à indiquer à quel moment les connecteurs source d'un scénario d'intégration vont produire des documents. Créer une programmation consiste à associer des documents produits à un ou plusieurs programmeurs.

Chaque programmeur joue le rôle d'une minuterie qui fait produire des documents à un connecteur soit de manière ponctuelle (le 04 octobre 2002) soit à intervalles réguliers (toutes les deux heures pour une période donnée).

Bilans de traitement

Les bilans de traitement sont des documents que produisent les connecteurs ou la boîte de mapping chaque fois qu'ils traitent un document. Ce bilan permet de savoir si le document a été traité avec succès ou non et si des problèmes sont survenus. Ces bilans peuvent être consommés par d'autres connecteurs pour avertir, par exemple, de problèmes de traitement survenus dans Connect-It.

Exemple :

Le connecteur e-mail peut consommer les bilans de traitement d'un scénario et les envoyer sous forme de messages à l'administrateur de Connect-It.

Certificat d'autorisation

Le certificat d'autorisation est un fichier texte permettant d'activer les connecteurs de base et les connecteurs optionnels et additionnels compris dans la formule Connect-It que vous avez choisie.

Application externe

Une application externe évoque autant les applications que les sources de données auxquelles se connecte un connecteur. Exemples : l'application AssetCenter, un fichier XML, une messagerie.

Blob

Les champs de type Blob (Binary Large Object) sont des champs correspondant à des données binaires : fichier son, fichier vidéo, fichier image.

Cache

Les fichiers de cache utilisés par les connecteurs contiennent la description des types de documents qu'ils publient. Le fichier de cache a pour nom d'extension CCH. Un cache permet de travailler rapidement avec le connecteur sans avoir à attendre le rapatriement de la description des types de document.

Collection

Dans un type de document, une collection représente la relation entre un ou plusieurs éléments et plusieurs autres éléments.

Exemple : lorsqu'un type de document correspondant à une table de base de données a une collection, cette collection correspond à une autre table dans la base de données. Une collection implique que le lien entre les deux tables est un lien 1-n ou n-n. Ce type de lien implique qu'un enregistrement dans la première table peut être lié à un ou plusieurs enregistrements de la deuxième table, et vice versa.

Exemple : dans un type de document correspondant à un message e-mail envoyé, une collection correspond au destinataire du message (un message peut avoir plusieurs destinataires).

Composant

Un composant est un connecteur ou un outil interne (boîte de mapping).

Console des services

La console Connect-It est une interface graphique vous permettant de gérer vos scénarios et de créer des services que vous leur associez.

Consommation

La consommation est le processus par lequel un connecteur destination écrit ou envoie des documents dans ou vers une application externe.

DAT

Les fichiers portant l'extension DAT correspondent aux document traités par un scénario donné.

Ces fichiers sont uniquement produits si vous sélectionnez l'option **Utiliser des fichiers** dans la fenêtre de configuration du journal des documents (**Journal/ Configurer le journal des documents**).

DBK

Les fichiers DBK sont des fichiers XML correspondant à la documentation d'un scénario. Ces fichiers sont conformes à la DTD DocBook. Pour plus d'informations sur ce fichier, consultez le chapitre [Documentations des scénarios](#) [page 171].

Directives

Les directives d'un connecteur sont les instructions qu'un connecteur suit quand il produit ou consomme des documents.

Exemple : pour un connecteur de type Base de données, les clauses WHERE et ORDERBY qui lui permettent de filtrer les enregistrements d'une base de données source sont des directives de consommation.

Pour une présentation des directives, consultez le manuel **Connecteurs**, chapitre **Directives des connecteurs**.

Déploiement du connecteur

Déployer un connecteur Java dans l'éditeur de scénarios consiste à spécifier :

- un fichier de déploiement
- la possibilité pour le connecteur de produire ou consommer des documents
- une icône qui le symbolise dans la boîte à outils

- une série d'options

Pour plus d'informations sur le déploiement des connecteurs, consultez le chapitre [Kit de développement Java de Connect-It](#) [page 229], section [Création d'un connecteur événementiel](#) [page 261]/ [Déployer le connecteur Java - connecteur événementiel](#) [page 263].

Editeur de scénarios

L'éditeur de scénarios est l'interface graphique qui vous permet de créer des scénarios d'intégration. Ces scénarios sont identifiés par le nom d'extension de fichier SCN.

Élément

L'arborescence d'un type de document présente les éléments suivants :

- Structure
- Collection
- Champ

Cet élément est l'élément terminal d'un type de document.

Formats utilisateur

Les formats utilisateur sont des formats de type date ou numérique que l'utilisateur peut définir pour les utiliser dans ses scripts de mappings. Ces formats s'utilisent exclusivement avec les fonctions **PifUserFmtVarToStr** et **PifUserFmtStrToVar**. Pour l'utilisation de ces fonctions, reportez-vous à la référence de programmation en ligne disponible avec la touche F1 chaque fois que vous rédigez un script dans Connect-It.

Pour une présentation des formats utilisateur, consultez le chapitre [Scripts de mapping](#) [page 121], section [Formats utilisateur](#) [page 149].

IDX

Un fichier IDX est un fichier d'indexation qui permet de retrouver rapidement les données contenues dans les fichiers DAT et LOG dans le journal des documents.

Ces fichiers sont uniquement produits si vous sélectionnez l'option **Utiliser des fichiers** dans la fenêtre de configuration du journal des documents (**Journal/ Configurer le journal des documents**).

MSG

Un fichier MSG contient les messages présents dans le journal des documents. Ces fichiers sont uniquement produits si vous sélectionnez l'option **Utiliser des fichiers** dans la fenêtre de configuration du journal des documents (**Journal/ Configurer le journal des documents**).

Pointeur de programmation

Le pointeur de programmation permet de réduire la quantité des données traitées par un scénario.

Un pointeur correspondant à une date de dernière modification d'un enregistrement permet à un connecteur de ne traiter que les enregistrements créés ou mis à jour depuis son dernier lancement.

Quand un scénario est lancé en mode programmé, l'état du pointeur d'un type de document produit permet au connecteur de ne traiter que les données non traitées lors de la session précédente.

Exemple : un connecteur Asset Management a lu, lors de la dernière session, les enregistrements de la table des biens à 9h00mn00s. A la session suivante, il lira tous les enregistrements créés à partir de 9h00mn01s.

Production

La production est le processus par lequel un connecteur source lit ou reçoit des données dans ou d'une application externe afin de les transformer en documents.

Programmateur

Un programmateur est un minuteur que vous associer à un ou plusieurs types de document produits par les connecteurs source d'un scénario.

Pour éditer les programmeurs d'un scénario, choisissez **Scénario/Programmateurs**.

Deux programmeurs préconfigurés sont livrés avec l'éditeur de scénarios :

- le programmeur **Une fois**
Ce programmeur est le programmeur associé par défaut à tous les types de document produits d'un scénario.
- le programmeur **Synchrone**
Ce programmeur permet à un connecteur de produire un document toutes les secondes.

Statistiques

Après chaque session, l'éditeur de scénarios vous permet d'obtenir des statistiques sur le nombre de documents traités par chaque composant du scénario. Exemple : le nombre de documents traités par le connecteur Asset Management durant une session.

Pour obtenir les statistiques de chaque connecteur, positionnez votre pointeur sur la partie supérieure d'un connecteur dans le schéma d'un scénario.

Structure

Une structure dans un type de document indique une relation de type 1:1 avec le noeud contenant cette structure. Un noeud contenant une structure peut être le noeud racine du type de document, une structure ou une collection.

Exemple : dans un type de document correspondant à une table de base de donnée, chaque structure correspond à une autre table. Une structure indique que le lien entre les deux tables est de type 1:1 : un enregistrement de la première table ne peut être lié qu'à un et un seul enregistrement de l'autre table.

Action après traitement

Certains connecteurs (Exemple : le connecteur InfraTools Desktop Discovery) peuvent agir sur les documents qu'ils ont produits. Dans ce cas, une page de l'assistant de configuration de ces connecteurs vous permet de sélectionner

des actions après traitement des documents produits. Exemple : suppression des données source après leur traitement réussi dans Connect-It.

Type de documents pivot

Un type de document pivot est une représentation générique d'une entité (un bien, un logiciel, une personne, etc.). Un connecteur source établit la correspondance entre ses propres types de document et les types de document pivots.

Les types de document pivot permettent le transfert de données entre un connecteur source et un connecteur destination sans que la création d'un mapping soit nécessaire.

Visionneuse de journal

La visionneuse de fichiers journaux permet de lire les fichiers LOG créés chaque fois qu'un scénario tourne en tant que service sous Windows.

Vues

Des vues du schéma d'un scénario peuvent être créées afin de faciliter la lecture visuelle du scénario. Exemple : dans un scénario comprenant une dizaine de composants différents, vous pouvez créer une vue où seuls certains de ces composants sont visibles.

Pour une présentation des vues d'un scénario, consultez le chapitre [Editeur de scénarios](#) [page 31], section [Fenêtre principale](#) [page 31]/ [Vues d'un schéma de scénario](#) [page 37].

DSC

Un fichier de description (extension de nom de fichier .dsc) décrit la façon dont les données du fichier texte sont organisées. Ces données correspondent aux valeurs des champs des tables d'une base de données.

Le connecteur Texte délimité publie les types de document créés dans le fichier de description.

Exemple : Un fichier de description précise que :

- le fichier texte contient les données d'une seule table (table des employés)
- la première ligne du fichier indique le titre des colonnes (chaque colonne correspond à un champ de la table)
- le signe "~" est utilisé pour séparer les valeurs

Termes transversaux

AQL

L'AQL ("Advanced Query Language") est le langage de requête utilisé par AssetCenter pour accéder aux données de la base de données AssetCenter. C'est un langage comparable au SQL. Il est automatiquement traduit dans le langage SQL du moteur de la base de données lors de l'utilisation des requêtes.

DSE

Les DSE (Directory Entry Service) sont les entrées qui constituent l'arborescence d'un annuaire LDAP. Ces informations sont représentées sous la forme de branches.

Chaque entrée de l'annuaire LDAP correspond à un objet abstrait ou réel (par exemple une personne, un objet matériel, des paramètres).

FSF

Un fichier portant l'extension .fsf (Fingerprint Save File) contient toutes les données relatives à un ordinateur.

Un fichier .fsf est généré dans le cadre d'un inventaire de type InfraTools Desktop Discovery. Les informations collectées lors de l'inventaire et stockées dans un fichier .fsf peuvent ensuite être analysées pour permettre une mise à jour des données relatives à un ordinateur, comme dans un processus de réconciliation.

JCA

Mis point par Sun, JCA regroupe un ensemble de spécifications Java qui décrivent une interface: "Java Connector Architecture".

Son objectif principal est de définir une couche de standards permettant à un serveur Java de dialoguer avec des applications basées sur des technologies tierces.

JCA vise à faciliter la mise au point d'interfaces de connexions et de connecteurs standards.

Réconciliation

La réconciliation est l'intégration de données entrantes provenant d'une application source considérées plus à jour que les données déjà existantes dans une application destination.

- Si la réponse est "non", les données sont insérées
- Si la réponse est "oui", les données existantes sont mises à jour en fonction des nouvelles informations contenues dans les données entrantes

Ce processus est basé sur la question suivante : "Les données entrantes à réconcilier existent-elles déjà dans l'application cible ?"

Quand l'application cible est vide, ce processus n'est plus une réconciliation mais un import.

Schéma XML

Un schéma XML est la définition du contenu utilisé dans un document XML. Le schéma XML est un surensemble de la DTD, qui est elle-même un schéma SGML standard.

Contrairement à une DTD, les schémas XML sont écrits avec une syntaxe XML. Cette syntaxe est plus prolix qu'une DTD.

Les schémas XML peuvent être créés avec tout outil XML.

Table de correspondance

Une table de correspondance est un tableau qui fait correspondre des éléments d'un ensemble à d'autres éléments d'un autre ensemble.

Transaction

Une transaction est un élément de dialogue entre un ordinateur et un utilisateur ou entre deux ordinateurs, comme par exemple une demande d'information et la réponse.

Variables globales

Une variable globale est une variable dont la valeur est accessible et modifiable par toute instruction d'un programme, et pas uniquement dans le module qui la définit.

XSL

Le XSL (eXtensible StyleSheet Language) est un langage de feuilles de style extensible développé spécialement pour XML. Il n'est pas reconnu comme un standard officiel.

Index

INDEX

A

Action après traitement

Définition, 293

Application externe

Définition, 288

AQL

Définition, 295

Assistant de configuration amélioré

Attributs obligatoires, 269

Attributs optionnels, 270

Contenu du fichier DSD, 268

Création, 267

Définition des propriétés de configuration, 269

Fichier DSD, 267

Page de configuration, 271

Propriétés de configuration, 270

B

Barres d'outils, 32

Icônes, 32

Bilans de traitement, 213

Contenu, 214

Définition, 288

Utilisation, 219

Blob, 118

Définition, 289

C

Cache, 165

Définition, 289

CCI, 231

Certificat d'autorisation

Définition, 288

Saisie, 23

Chaîne de caractères, 138

Concaténation, 139, 138

Champ

Éléments d'un type de document, 282

Chr(), 139

Classes événementielles, 237

Clé de réconciliation, 206

Collection, 114

Concaténation de membres - Exemple, 153

Création de membres - Exemple, 152

Définition, 289

Éléments d'un type de document, 282

Exemple, 151

Mapper plusieurs champs - Exemple, 154

Composant

Définition, 289

Liaison, 81

Suppression, 83

Connecteur, 18

Amélioration du traitement des documents, 208

Assistant de configuration, 79

Autodescription, 247

Calcul du nombre de documents à produire, 201

Certificat d'autorisation, 23

- Connecteur Asset Management - Amélioration des performances, 210
 - Connecteur Asset Management - Reconnexion à chaque session, 204
 - Connecteur de base - Installation, 25
 - Connecteur de destination - Type de documents pivot, 162
 - Connecteur de type Base de données - Amélioration des performances, 212
 - Connecteur événementiel - Création, 261
 - Connecteur Java - Autodescription, 249
 - Connecteur Java - Exemple de création, 264
 - Connecteur Java - Norme UDC, 238
 - Connecteurs additionnels - Installation, 27
 - Connecteurs optionnels - Installation, 26
 - Contrôle des requêtes exécutées, 200
 - Définition, 279
 - Favoris, 48
 - Fichier de cache, 165
 - Java, 241
 - Liaison, 81
 - Modifier le connecteur associé à un service, 196
 - Principe de fonctionnement, 17
 - Programmation, 179
 - Reconfiguration, 81
 - Reconnexion automatique, 204
 - Suppression, 83
 - Travailler hors connexion, 167
 - Utilisation de l'option cache, 202
 - Connecteur Java
 - Autodescription, 247
 - Configuration de la machine virtuelle, 259
 - Contrat de connexion, 243 , 241
 - Contrat de connexion - Connecteur d'exemple, 264
 - Contrat de sécurité, 243
 - Contrat de sécurité - Connecteur d'exemple, 265
 - Contrat de transaction - Connecteur d'exemple, 265
 - Création, 241
 - Définir les données - Connecteur d'exemple, 266
 - Définition de l'interaction des données, 245
 - Définition des données, 245
 - Déploiement, 259
 - Directives des opérations - Autodescription, 250
 - Exemple de création, 264
 - Ineraction des données - Connecteur d'exemple, 266
 - Pointeurs de programmation - Autodescription, 256
 - Prérequis, 241
 - Connecteurs internes, 00-159
 - Console des services, 191
 - Définition, 289
 - Consommation, 206
 - Définition, 290
 - Contrat de connexion, 243 , 241
 - Exemple, 264
 - Paramètres de création de la connexion, 242
 - Type de connexion, 241
 - Contrat de sécurité, 265 , 243
 - Contrat de transaction, 265 , 261
- ## D
- DAT
 - Définition, 290
 - Format, 19
 - DBK, 178
 - Définition, 290
 - Déploiement du connecteur
 - Définition, 290
 - Directives
 - Définition, 290 , 287
 - Document
 - Amélioration de la production, 200
 - Calcul du nombre de documents à produire, 201
 - Chemin, 106
 - Consommation, 206
 - Définition, 281
 - Détail, 61
 - Détail - Recherche, 65
 - Document parent, 63
 - Journal des documents, 164
 - Ligne de suivi, 66
 - Pause, 169
 - Statistiques, 198

- Traitement, 84
 - XML, 286 , 64
 - Documentation de scénario, 171
 - Configurer le processeur XSL, 176
 - Contenu, 171
 - Création, 178
 - Edition des propriétés, 176
 - Feuilles de style XSL prédéfinies, 177
 - Format DBK, 178
 - Format HTM, 175
 - Propriétés, 175
 - Visualisation, 174
 - Documents
 - Typologie des icônes, 60
 - Données
 - Définition, 245
 - Prévisualisation, 109
 - DSC
 - Définition, 294
 - DSD, 267
 - Contenu du fichier, 268
 - Problèmes courants , 272
 - DSE
 - Définition, 295
 - DTD
 - Copie, 88
 - Définition, 286
- E**
- Editeur de scénarios, 40 , 31
 - Barres d'outils, 32
 - Définition, 291
 - Ergonomie, 32
 - Journaux, 54
 - Menu Administration, 45
 - Menu Affichage, 42
 - Menu Aide, 46
 - Menu Composant, 44
 - Menu Edition, 41
 - Menu Fichier, 40
 - Menu Java, 46
 - Menu Journal, 45
 - Menus, 40
 - Menu Scénario, 42
 - Menus contextuels, 46
 - Options, 69
 - Volets, 36
 - Vues, 37
 - Vues - règles de fonctionnement, 39
 - Élément, 105
 - Définition, 291
 - Recherche, 111
 - Remplacer un élément dans un script de mapping, 112
 - Else, 122
 - Else If, 122
 - End If, 122
- F**
- Fichier associé
 - Configuration de l'éditeur de texte, 144
 - Création, 141
 - Duplication, 142
 - Editer le texte, 143
 - Ouverture, 142
 - Suppression, 142
 - Validation des scripts, 143
 - Fichier LOG, 197
 - Fichiers de cache, 165
 - Fonctions Basic, 122
 - For, 123
 - Format, 149
 - Définition, 291
 - Format utilisateur de type date, 149
 - Format utilisateur de type numérique, 150
 - Syntaxe, 150
 - FSF
 - Définition, 295
- H**
- Hors connexion, 167
- I**
- IDX
 - Définition, 291
 - format, 20

- If, 122
- Installation, 24 , 19
 - Connecteurs additionnels, 28 , 27
 - Connecteurs de base installés, 25
 - Connecteurs optionnels, 26
 - Fichiers du dossier d'installation, 24
 - Kit de données, 27
 - UNIX, 22
 - Windows, 20
- J**
- Java
 - Contenu du kit de développement, 229
 - Kit de développement, 229
- JCA, 230
 - Définition, 296
- JMV, 259
- Journal, 54
 - Lignes de suivi, 59
 - Purge de la mémoire, 58
 - Rechargement, 59
 - Stockage disque, 58
- Journal des documents, 164
 - Configuration, 208
- Journaux
 - Configuration, 57
 - Consultation, 196
 - Définition, 287
 - Descriptif, 56
- L**
- Librairies dynamiques
 - Informations, 29
 - UNIX, 22
- Ligne de suivi
 - Définition, 66
 - Détail, 68
- M**
- Machine virtuelle Java, 259
- Mapping, 97
 - Actions - Icônes, 101
 - Aide à la rédaction des scripts, 132
 - Boîte de mapping, 98
 - Classement, 110
 - Clé de réconciliation, 206
 - Configuration de la boîte de mapping, 98
 - Copier/ Coller, 108
 - Création, 104
 - Création automatique - Documents pivots, 158
 - Définition, 286
 - Désactivation, 112
 - Description, 109
 - Documents pivots, 158
 - Edition, 100
 - Erreur syntaxique, 169
 - Fenêtre d'édition, 100
 - Fonctions PIF, 125
 - Mapping champ à champ direct, 113
 - Mapping champ à collection, 117
 - Mapping collection à collection, 114
 - Mapping collection à document, 116
 - Mapping des champs de type blob, 118
 - Modifier les mappings Type de document publié - type de document pivot, 161
 - Prévisualisation des données, 109
 - Rechercher un élément, 111
 - Remplacer un élément dans un script de mapping, 112
 - Scripts, 121
 - Suppression, 107
 - Tables de chaînes de caractères, 146
 - Types de mapping, 112
- Mode test, 168
- MSG
 - Définition, 292
 - format, 19
- N**
- Norme UDC, 238
 - Implémentation, 240
- P**
- PIF, 125
- PifIgnoreCollectionMapping, 128
- PifIgnoreDocumentMapping, 126

- PifIgnoreNodeMapping, 127
 - PifRejectDocumentMapping, 126
 - Pointeur de programmation, 187
 - Définition, 292
 - Production, 200
 - Définition, 292
 - Programmeur, 180
 - Définition, 292
 - Edition, 196
 - Programmation, 179
 - Définition, 288
- R**
- ra.xml, 267
 - Réconciliation, 206
 - Définition, 296
 - Référence de programmation, 122
 - Return, 124
- S**
- Scénario
 - Assistant de mise en place, 79
 - Composant, 80
 - Copie de sauvegarde, 94
 - Enregistrement, 93
 - Mise en place, 79
 - Scénarios, 159
 - Console, 196
 - Définition, 286
 - Exploitation, 179
 - Fichiers de cache, 165
 - Optimisation des performances, 197
 - Test et débogage, 163
 - Utilisation des bilans de traitement, 219
 - Schéma XML
 - Définition, 296
 - Scripts additionnels, 133
 - Scripts de mapping, 132 , 121
 - Associer une valeur fixe à un élément destination, 132
 - Compteur, 130
 - Copier le chemin d'un élément, 135
 - Création d'un fichier associé, 141
 - Dupliquer un élément destination, 134
 - Edition des fichiers associés, 140
 - Fonctions Basic, 122
 - Glisser-déplacer plusieurs champs, 133
 - Menus contextuels de la fenêtre d'édition des mappings, 136
 - Retrouver vos éléments source et destination, 134
 - Scripts additionnels, 133
 - Utiliser le menu contextuel du script de mapping, 135
 - Variables globales, 129
- Select, 125
- Service, 190
 - Créer un service sous UNIX, 194
 - Créer un service sous Windows, 193
 - Fichier LOG, 197
 - SPI, 234
 - Statistiques, 198
 - Définition, 293
 - Structure
 - Définition, 293
 - Éléments d'un type de document, 282
- T**
- Table de correspondance, 147
 - Création, 148
 - Définition, 296
 - Tables de chaînes de caractères, 146
 - Tester un type de document produit, 163
 - Then, 122
 - Transaction
 - Définition, 297
 - Type de document, 84
 - Chemin, 105
 - Classement, 110
 - Création, 87
 - Définition, 281
 - DTD, 88
 - Éléments d'un type de document, 282
 - Fenêtre d'édition, 85
 - Fenêtre d'édition - Menus contextuels, 92
 - Mapping, 97

- Noeud parent et noeuds fils, 284
 - Rechercher un élément, 91
 - Tester un type de document produit, 163
 - Voir les données, 89
 - Type de documents pivot, 159 , 157
 - Définition, 294
 - Mappings automatiques, 158
 - Modifier les mappings Type de document publié, 161
 - Principe de fonctionnement, 158
 - Types de document
 - Autodescription d'un connecteur Java, 248
 - Types de mapping, 112
- U**
- UDC, 238
 - Echange des données, 245
 - UNIX
 - Chaîne de caractères, 139
 - Créer un service, 194
- V**
- Variables globales, 146
 - Compteur, 130
 - Déclaration, 129
 - Définition, 297
 - Sauvegarde, 131
 - Script, 129
 - Scripts de mapping, 133
 - Variables locales, 129
 - Visionneuse de journal, 21
 - Définition, 294
 - Vues, 37
 - Définition, 294
- W**
- WHERE, 205
 - While, 123
- X**
- XML
 - Définition, 286
 - XSL, 176

