

Retek[®] Store Inventory Management[™] 11.0

Guide des opérations

Siège social :

Retek Inc.
Retek on the Mall
950 Nicollet Mall
Minneapolis, MN 55403
Etats-Unis
888. 61. RETEK (numéro
gratuit depuis les Etats-Unis)
Standard
+1 612 587 5000
Fax :
+1 612 587 5100

Siège européen :

Retek
110 Wigmore Street
London
W1U 3RW
Royaume-Uni
Standard :
+44 (0)20 7563 4600
Département commercial :
+44 (0)20 7563 46 46
Fax :
+44 (0)20 7563 46 10

Le logiciel décrit dans cette documentation fait l'objet d'un accord de licence. Il se compose d'informations confidentielles qui restent la propriété de Retek Inc. et son utilisation est soumise au respect des termes de cet accord.

Aucune partie de cette documentation ne peut être reproduite ou transmise, sous quelque forme ou par quelque moyen que ce soit, sans l'autorisation écrite expresse de Retek Inc., Retek on the Mall, 950 Nicollet Mall, Minneapolis, MN 55403, Etats-Unis. La mention de copyright ne peut être supprimée sans l'autorisation de Retek Inc.

Les informations contenues dans cette documentation sont susceptibles d'être modifiées sans préavis.

La documentation relative aux produits fournie par Retek est en lecture seule afin d'assurer l'intégrité de son contenu. Le support clientèle Retek n'est pas assuré si la documentation a été modifiée sans l'autorisation de Retek.

La fonctionnalité décrite dans le présent document s'applique à cette version, comme indiqué sur la page de titre, et à aucune autre version du logiciel, y compris mais sans s'y limiter aux versions ultérieures du composant logiciel. La fonctionnalité décrite dans le présent document pourra être modifiée dans les futures versions du logiciel et Retek se réserve le droit d'apporter ces modifications à sa propre discrétion.

Retek® Store Inventory Management™ est une marque de Retek Inc.

Retek et le logo Retek sont des marques déposées de Retek Inc.

Cet ouvrage non publié est protégé par un accord de confidentialité, par la législation sur le secret industriel, par copyright et par d'autres lois. En cas de publication, la mention suivante s'applique :

©2005 Retek Inc. Tous droits réservés.

Tous les autres noms de produits mentionnés sont des marques commerciales ou des marques déposées de leurs détenteurs respectifs et doivent être considérés comme tels.

Imprimé aux Etats-Unis d'Amérique.

Support clientèle

Horaires du support clientèle

Le support clientèle est disponible 7 jours sur 7, 24 heures sur 24 et 365 jours par an, par email, téléphone et Internet.

Selon l'option de support choisie par le client (Standard, Plus ou Premium), les heures d'accès à certains services peuvent être limitées. Les problèmes de gravité 1 (graves) sont traités 7 jours sur 7 et 24 heures sur 24. Ils reçoivent une attention permanente jusqu'à résolution du problème, à condition que le client soit titulaire d'un contrat de maintenance actif. Les clients Retek titulaires d'un contrat de maintenance actif peuvent contacter un représentant du support clientèle, selon les termes du contrat, de l'une des façons suivantes :

Méthode de contact	Informations de contact
--------------------	-------------------------

E-mail	support@retек.com
---------------	-------------------

Internet (ROCS)	rocs.retek.com Site Web client sécurisé de Retek pour la mise à jour et la consultation des problèmes
------------------------	--

Téléphone	+1 612 587 5800
------------------	-----------------

D'autres numéros gratuits sont disponibles dans plusieurs régions du monde :

Australie	+1 800 555 923 (AU-Telstra) ou +1 800 000 562 (AU-Optus)
France	0800 90 91 66
Hong Kong	800 96 4262
Corée	00 308 13 1342
Royaume-Uni	0800 917 2863
Etats-Unis	+1 800 61 RETEK ou 800 617 3835

Courrier électronique	Support clientèle Retek Retek on the Mall 950 Nicollet Mall Minneapolis, MN 55403
------------------------------	--

Lorsque vous contactez le support clientèle, veuillez fournir les informations suivantes :

- la version du produit et le nom du programme ou module.
- une description fonctionnelle et technique du problème (en incluant les effets de ce problème sur votre entreprise).
- des instructions détaillées, étape par étape, pour reproduire le problème.
- l'intitulé exact du message d'erreur reçu.
- des captures d'écran de chaque étape.

Table des matières

Chapitre 1 – Introduction	1
Présentation	1
Présentation de l'architecture technique	2
Points d'intégration entre SIM et le commerce de détail	3
Public auquel s'adresse ce guide	4
Javadoc pour SIM	4
Où trouver des informations supplémentaires	4
Chapitre 2 – Configuration du système d'arrière-plan	5
Environnements pris en charge	5
Produits Retek pris en charge	5
Fichiers de configuration	5
Fichier de configuration JNDI (jndi_providers.xml)	5
Fichier de configuration JDBC (jdbc.cfg)	5
Fichier de configuration DAO (dao.cfg)	7
Fichier de configuration réseau (network.cfg)	8
Fichier de configuration LDAP (ldap.cfg)	8
Fichier de configuration Client master (clientmaster.cfg)	9
rmsupload.cfg	9
Classe initiale exécutée	10
Reporting.cfg	14
Wireless_services.cfg	15
Informations de journalisation	15
Emplacement par défaut des fichiers journaux client et serveur	15
Niveaux de journalisation établis dans les fichiers de configuration (.cfg)	16
Gestion des exceptions	16
Options de machine virtuelle Java (JVM)	17
Chapitre 3 – Architecture technique	19
Présentation	19
SIM et ISO (Integrated Store Operations)	19
Avantages de l'architecture	20
Diagrammes et description de l'architecture technique de SIM	21
Niveau Présentation	22
Niveau intermédiaire	23
Niveau Base de données	24

Topologie distribuée.....	25
Paquetages contenant des services et des objets-métier clés.....	27
Présentation du verrouillage des activités	28
Services de support technique	28
Service de transaction.....	28
Service de journalisation	28
Service d'internationalisation	28
Service de sécurité.....	28
Sécurité et protocole LDAP (Light Directory Access Protocol).....	29
Normes et termes Java relatifs à SIM.....	30
Chapitre 4 – SIM, le RIB, JMS et RSL.....	33
Présentation de SIM et du RIB.....	33
Format des messages XML	33
Processus d'abonnement aux messages	34
Diagramme du processus d'abonnement aux messages.....	34
Description du processus d'abonnement aux messages.....	34
Messages d'erreur RIB relatifs à l'abonnement	36
RIB Error Hospital	36
Processus de publication des messages	37
Diagramme du processus de publication des messages	37
Description du processus de publication des messages.....	37
Tableau de mappage des abonnements	38
Tableau de mappage des publications.....	41
Descriptions fonctionnelles des messages.....	42
Présentation de SIM et RSL	46
Services RSL utilisés par SIM.....	47
SIM, RPOS et le logiciel de messagerie JMS	48
Résumé de l'intégration de l'interface RPM-SIM	49
Chapitre 5 – Flux de données de l'interface d'intégration.....	51
Présentation	51
Flux de données SIM inter-systèmes	51
De SIM à WMS (Warehouse Management System).....	52
De WMS à SIM.....	52
D'un système de point de vente (RPOS par exemple) à SIM	52
De SIM à un système de point de vente (RPOS par exemple).....	52
Du système de commercialisation à SIM	52

De SIM au système de commercialisation	53
De SIM au système de commercialisation via le module de téléchargement de stock du système de commercialisation	54
De SIM au système de création de rapports (Crystal Enterprise par exemple)	54
De SIM à un système de gestion des prix (RPM par exemple).....	54
D'un système de gestion des prix (RPM par exemple) à SIM	54
Chapitre 6 – Conception et présentations des fonctionnalités.....	55
Présentation fonctionnelle des ajustements de stock.....	55
Récapitulatif des codes de motif et des statuts	56
Présentation fonctionnelle des commandes magasin	57
Demandes d'articles.....	57
Présentation fonctionnelle des modifications de prix	59
Présentation fonctionnelle de la réception	60
Réception des transferts.....	60
Livraison entrepôt	61
livraison directe au magasin.....	62
Présentation fonctionnelle de la création de rapports.....	64
Rapports opérationnels	64
Rapports analytiques (et ad hoc)	64
Suppositions	64
Présentation fonctionnelle des retours et des demandes de retour	65
Retours	65
Demandes de retour.....	65
Présentation fonctionnelle de l'ordonnancement.....	66
Présentation fonctionnelle du réapprovisionnement des rayons	67
Récapitulatif du calcul de réapprovisionnement	67
Présentation fonctionnelle des comptages.....	69
Comptages ad hoc	69
Comptages de type Unité	70
Comptages de type Unité et valeur.....	70
Comptages Unité et valeur Tous les rayons	71
Comptages de type Ligne problématique	72
Présentation fonctionnelle de l'étiquetage	73
Présentation fonctionnelle du transfert sortant.....	74
Création de transferts entre magasins.....	74
Demandes de transfert.....	75
Situations fonctionnelles de processus RIB relatifs aux transferts.....	75
Présentation fonctionnelle des pertes	77

Chapitre 7 – Traitements par lots JAVA.....	79
Présentation du traitement par lots	79
Exécution d'un traitement par lots Java.....	79
Récapitulatif des fichiers exécutables associés aux paquetages et aux classes Java	80
Gestionnaire et ligne de commande	80
Normes de traitement par lots de valeur renvoyée.....	80
Descriptions fonctionnelles et dépendances.....	81
Détails supplémentaires concernant ThirdPartyStockCountParser	87
Processus à plusieurs tâches légères et processus multiples	88
Redémarrage et reprise.....	88
Processus spécifique à DexnexFileParser	88
Annexe A – Spécifications de mise en page du fichier de comptage .	89
Fichier de configuration rmsupload.cfg	89
Spécifications du fichier plat des résultats de comptage.....	89
Annexe B – Spécifications de mise en page du fichier de traitement par lots	91
Fichier plat utilisé lors du processus de traitement par lots ResaFileParser	91
Fichier plat utilisé lors du processus de traitement par lots DexnexFileParser.....	97
Structure du fichier : livraison 894.....	97
Fichier plat utilisé lors du processus de traitement par lots ThirdPartyStockCountParser	101
Définition de la mise en page des fichiers dans RGIS	101
Données des fichiers d'exemple RGIS.....	101

Chapitre 1 – Introduction

Ce guide des opérations explique les processus « d'arrière-plan » dans Retek Store Inventory Management (SIM). Le logiciel SIM est conçu comme une application autonome pouvant être personnalisée afin de fonctionner avec tout système de commercialisation.

Présentation

SIM permet au personnel d'un magasin de gérer les ventes et les services et de personnaliser les interactions client en offrant aux utilisateurs la possibilité d'effectuer les tâches administratives courantes dans la surface de vente du magasin. Vous pouvez ainsi bénéficier de nombreux avantages, forte amélioration du taux de conversion client, optimisation du service clientèle, diminution des coûts de transport d'inventaire et des démarques. SIM fournit les informations et fonctions dont les employés des magasins ont besoin pour maintenir des niveaux d'inventaire optimaux et concrétiser des ventes.

La solution SIM présente les avantages suivants :

- Elle améliore les niveaux d'inventaire permanents en permettant la gestion des stocks dans le magasin grâce à des périphériques de poche et des ordinateurs de magasin.
- Elle réduit le temps de traitement de la réception et de l'enregistrement des marchandises entrantes.
- Elle permet de réceptionner, de transférer et d'effectuer le suivi des marchandises de façon exacte, efficace et simple.
- Elle réduit les coûts de technologie en centralisant les besoins en équipement.
- Elle guide les utilisateurs lors des transactions nécessaires.
- Elle permet de personnaliser le produit via une plate-forme de technologie étendue. Les modifications du détaillant sont isolées lors des mises à niveau du produit, ce qui diminue le coût total de propriété.

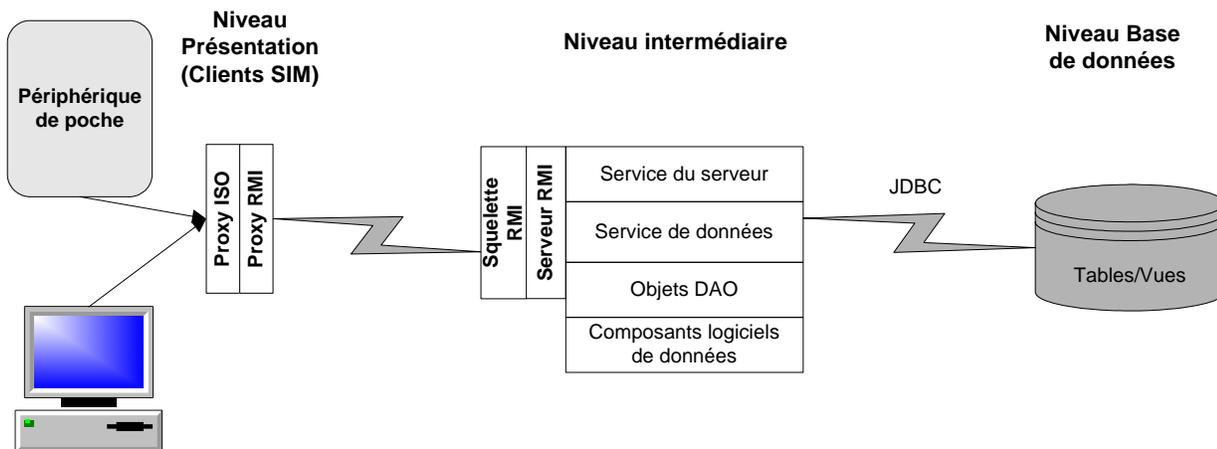
Présentation de l'architecture technique

La plate-forme informatique distribuée robuste de SIM améliore les performances et permet la hiérarchisation.

L'architecture multi-niveaux de SIM permet l'encapsulation de logique applicative, évitant au client de faire face à la complexité du système en arrière-plan. La séparation de la présentation, de la logique applicative et des données clarifie le logiciel et facilite sa gestion et sa modification. Chaque niveau de l'architecture n'a pas à se préoccuper des tâches de fonctionnement internes des autres niveaux.

L'un des avantages les plus notables de SIM est sa topologie distribuée flexible. SIM offre une transparence totale des emplacements, l'emplacement des données et des services étant basé sur les besoins du détaillant et non sur des limitations techniques. Le serveur n'est pas déployé dans le magasin. Les clients de l'application se connectent et communiquent avec le serveur quasiment en temps réel.

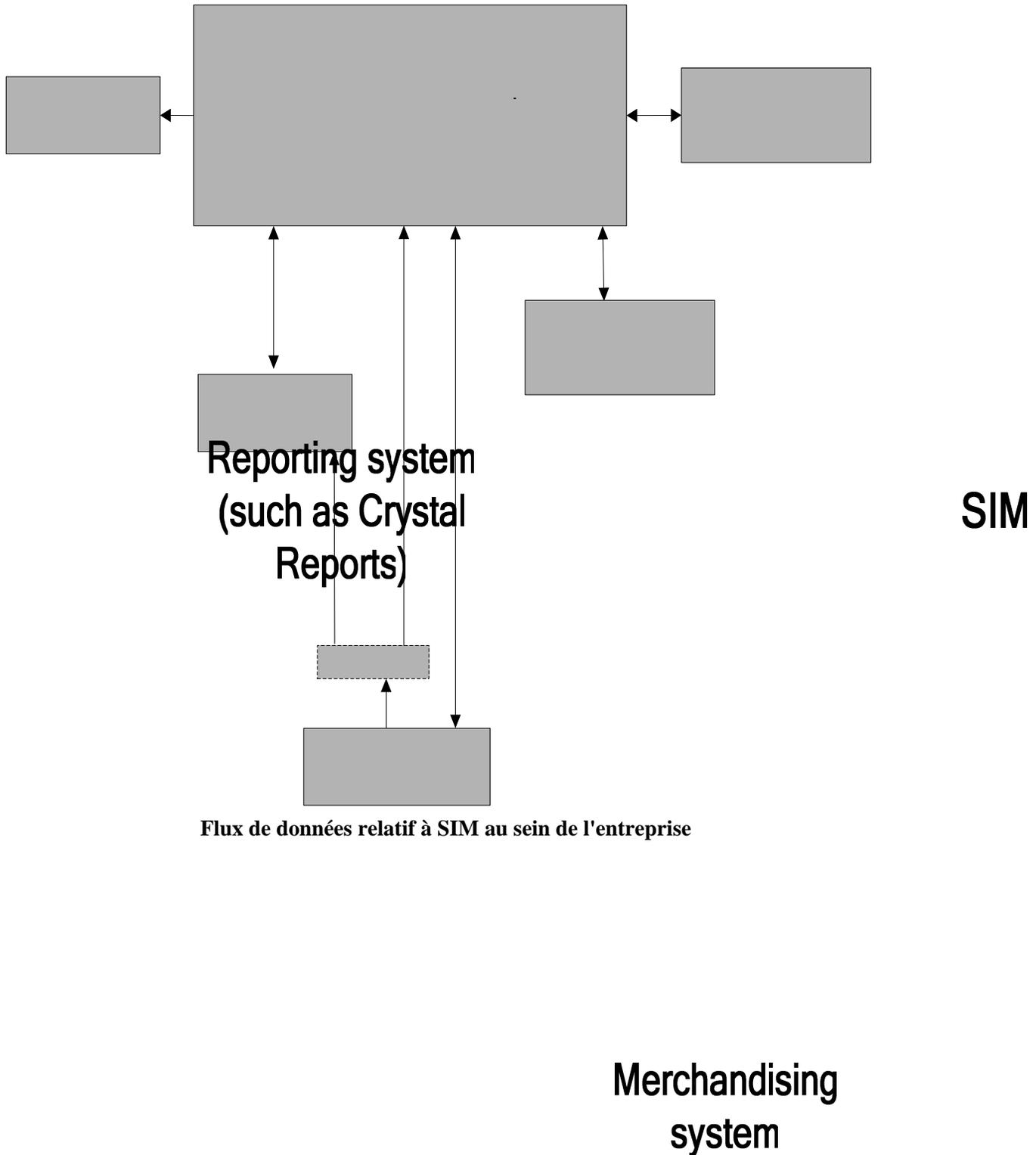
Le diagramme ci-après propose une vue d'ensemble des niveaux de l'architecture. Pour obtenir une description détaillée de ce diagramme, reportez-vous au chapitre 3, « Architecture technique ».



Architecture technique du logiciel SIM

Points d'intégration entre SIM et le commerce de détail

Le diagramme de vue d'ensemble ci-après présente le flux général des données entre les systèmes et produits de l'entreprise. Pour obtenir une description détaillée de ce diagramme, reportez-vous au chapitre 5, « Flux de données de l'interface d'intégration ».



Public auquel s'adresse ce guide

Toute personne souhaitant mieux comprendre le fonctionnement interne du système SIM trouvera des informations intéressantes dans ce guide. Il existe trois types génériques de public à qui s'adresse ce guide :

- Les analystes système et les responsables du système :
 - qui recherchent des informations sur les processus SIM internes ou sur la relation de ces processus avec les systèmes au sein de l'entreprise.
 - qui utilisent SIM régulièrement.
- Les intégrateurs et l'équipe en charge de l'implémentation de SIM au sein de l'entreprise.
- Les analystes d'entreprise à la recherche d'informations sur les processus et les interfaces afin de valider le soutien aux entreprises au sein de SIM et d'autres systèmes de l'entreprise.

Javadoc pour SIM

Javadoc est un outil de Sun Microsystems permettant de générer la documentation API au format HTML. Retek offre de la documentation Javadoc générée à partir d'un code SIM. Cliquez sur le fichier HTML « index » dans le dossier Javadoc applicable afin d'ouvrir Javadoc.

Où trouver des informations supplémentaires

- Documentation frontale de SIM (le guide de l'utilisateur SIM par exemple)
- Guide d'installation SIM
- Documentation du système RPOS de Retek (système de point de vente)
- Documentation du logiciel ISO (Integrated Store Operations), les notes de mise à jour ISO par exemple
- Documentation du logiciel RWMS (Retek Warehouse Management System)
- Documentation du produit RMS (Retek Merchandising System)
- Guide d'intégration Retek (Retek Integration Guide) et autre documentation relative à RIB

Chapitre 2 – Configuration du système d'arrière-plan

Ce chapitre du guide des opérations s'adresse aux administrateurs en charge de l'assistance et de la surveillance du système.

Ce chapitre ne contient pas de procédures. Son objectif est de présenter de façon générale les paramètres système clés, les paramètres de journalisation et la gestion des exceptions.

Environnements pris en charge

Pour connaître la configuration requise pour le client, le(s) serveur(s) et la base de données SIM, reportez-vous au guide d'installation du logiciel SIM. Notez que SIM est compatible avec Crystal Enterprise 10.0.

Produits Retek pris en charge

Cette version de SIM est compatible avec les logiciels suivants :

- RPOS 11.0
- RMS 11.0.2
- RIB 11.0.2
- RWMS 10.3.5

Fichiers de configuration

Les configurations clés de SIM définies par le client sont décrites dans cette section. Les paramètres système contenus dans les fichiers sont également décrits. De nombreux paramètres que les détaillants n'ont *pas* à modifier n'ont pas été décrits.

Notez que dans ces fichiers (et donc dans certains exemples ci-après issus de ces fichiers), le signe # situé devant une valeur indique que le texte qui suit n'est pas un paramètre mais un commentaire.

Certains paramètres des fichiers sont configurables. Ainsi, lorsque les détaillants installent SIM dans un environnement, ils doivent mettre à jour ces valeurs avec leurs propres paramètres.

Fichier de configuration JNDI (jndi_providers.xml)

SIM utilise ce fichier pour son intégration RSL à l'application RPM (Retek Price Management). Pour plus d'informations sur cette intégration, reportez-vous au « Chapitre 4 - SIM, le RIB, JMS et RSL ». Le fichier jndi providers contient des informations iiop destinées à l'application Retek que la plateforme ISO (Integrated Store Operations) appelle.

Fichier de configuration JDBC (jdbc.cfg)

Ce fichier décrit comment le système utilise la couche de persistance. Vous trouverez ci-après les valeurs clés relatives à SIM de ce fichier. Notez que certaines valeurs de ce fichier peuvent ne s'adresser qu'à l'équipe de développement ou être liées à un autre produit (un produit de PDV par exemple).

Configuration Oracle

La configuration dans ce fichier indique au système la base de données dans laquelle se trouvent les tables SIM. Notez qu'un détaillant utilisant une base de données spécifique insère des commentaires pour tous les paramètres de la base de données dans le fichier de configuration. La valeur du pilote ne doit normalement *pas* être modifiée. Le détaillant définit les noms de l'ordinateur, de la base de données et de l'utilisateur, ainsi que le mot de passe. La taille du pool dépend du nombre de connexions à la base de données disponibles que le détaillant souhaite garder disponible dans le pool. L'application règle automatiquement la taille du pool en fonction des besoins. Le paramètre JDBC_VERBOSE n'existe qu'à des fins de débogage. Si une connexion à la base de données n'est pas utilisée pendant un certain temps, elle s'arrête automatiquement. Le paramètre de verrouillage décrit la durée pendant laquelle le système essaie de fournir à l'utilisateur un verrouillage avant d'abandonner et de renvoyer un message indiquant que le verrouillage n'a pas pu être obtenu.

Exemple :

```
# Configuration Oracle
DATABASE=ORACLE
DRIVER=oracle.jdbc.driver.OracleDriver
URL=jdbc:oracle:thin:@<SIMDBMachineName>:1521:<SIMDBName>
POOL_INITIAL_SIZE=5
USER_NAME=<NomUtilSIM>
PASSWORD=<motdepasse>
JDBC_VERBOSE=false
#Temps de conservation des connexions non utilisées, exprimé en secondes
(3600 = 1 heure)
CONNECTION_EXPIRATION=3600
#Temps d'attente pour les verrouillages de bases de données, exprimé en
secondes
LOCK_WAIT=5
```

Configuration Oracle RMS

Cette section n'est valide que si le détaillant utilise RMS. Ces paramètres sont utilisés par le programme d'insertion de données. Exemple :

```
# Configuration Oracle RMS
RMS_DATABASE=ORACLE
RMS_DRIVER=oracle.jdbc.driver.OracleDriver
RMS_URL=jdbc:oracle:thin:@<RMSDBMachineName>:1521:<RMSDBName>
RMS_POOL_INITIAL_SIZE=5
RMS_USER_NAME=<NomUtilRMS>
RMS_PASSWORD=<motdepasse>
RMS_JDBC_VERBOSE=false
#Temps de conservation des connexions non utilisées, exprimé en secondes
(3600 = 1 heure)
RMS_CONNECTION_EXPIRATION=3600
```

Fichier de configuration DAO (dao.cfg)

Implémentations d'objets d'accès aux données (DAO) spécifiques

Ces paramètres sont relatifs aux informations d'accès aux données. Les valeurs permettent au détaillant de personnaliser la classe utilisée pour accéder aux données, pour un service spécifique. La grande majorité des paramètres de cette section sont associés à Oracle. Les valeurs ne doivent pas être modifiées si le système a été acheté avec RMS, qui doit utiliser Oracle. Un détaillant travaillant sur commande peut cependant écrire une implémentation qui fonctionnera pour toutes les bases de données utilisées et modifier les valeurs de cette section en conséquence.

Exemple :

```
#Implémentation de DAO spécifiques
EMPLOYEE_DAO=com.retek.iso.cs.dataaccess.rss.ldap.dao.EmployeeLdapDAO
LOCALITEM_DAO=com.retek.iso.cs.dataaccess.rss.oracle.dao.LocalItemRSSOracle
DAO
ALLOCATION_DAO=com.retek.iso.cs.dataaccess.rss.oracle.dao.AllocationRSSOracle
DAO
PACKITEM_DAO=com.retek.iso.cs.dataaccess.rss.oracle.dao.PackItemRSSOracleDAO
ITEMMASTER_DAO=com.retek.iso.cs.dataaccess.rss.oracle.dao.ItemMasterRSSOracle
DAO
SOURCE_DAO=com.retek.iso.cs.dataaccess.rss.oracle.dao.SourceRSSOracleDAO
MDSE_HIERARCHY_DAO=com.retek.iso.cs.dataaccess.rss.oracle.dao.MdseHierarchy
RSSOracleDAO
#LOCATION_DAO=com.retek.iso.cs.dataaccess.rss.oracle.dao.StoreRSSOracleDAO
```

Fichier de configuration réseau (network.cfg)

La connectivité entre le client SIM et le niveau intermédiaire s'effectue via un service RNS (Remote Naming Service), qui fournit au client SIM l'adresse IP et le port, nécessaires pour accéder au conteneur SIM. Ce fichier de configuration comprend des paramètres relatifs au service de nommage (naming service) de l'application ISO (Integrated Store Operations). Le client définit l'adresse IP du serveur sur lequel est exécuté le code. Le numéro de port (40000 dans l'exemple ci-après) peut être défini sur la valeur par défaut ou être modifié.



Remarque : si le détaillant modifie le numéro de port, la même valeur doit être utilisée dans les fichiers .bat et .sh relatifs au RNS. Les valeurs doivent toujours être synchronisées.

La valeur du paramètre `MAXIMUM_CONNECT_ATTEMPTS` indique au système le nombre de fois où il doit essayer de se connecter avant de passer en mode hors ligne, ce qui empêche l'utilisateur de continuer à utiliser le système. Le paramètre `VERSION` ne devrait pas avoir à être modifié.

```
# Voici une liste d'adresses, délimitées par des virgules, du service de
nommage.
# Exemple : 10.9.1.47:3000,10.9.1.47:4000,10.9.1.47:5000
#MASTER_NODE=<AdresseIpServeurNommage>:40000
MASTER_NODE=10.6.1.162:40000

# Voici l'adresse du service de nommage (naming service) pour un
# serveur d'application spécifique.
# Chaque serveur d'application possède une adresse unique.
# L'application Node Monitor se lie à cet élément.
#APPLICATION_NODE=<AdresseIpServeurApp>:40001
APPLICATION_NODE=10.6.1.162:40001

# Nombre maximal de tentatives de liaison à un service de nommage avant
# l'envoi d'une exception de coupure "Naming service introuvable".
MAX_CONNECT_ATTEMPTS=2

# Numéro de contrôle de version du service de nommage que tout
# client doit rechercher. Cela permet l'existence simultanée
# de plusieurs versions.
VERSION=1
```

Fichier de configuration LDAP (ldap.cfg)

La configuration de la sécurité s'effectue dans le fichier `ldap.cfg`. Les entrées de ce fichier indiquent notamment à SIM l'ordinateur et le port lui permettant de trouver le serveur LDAP. Vous trouverez ci-après les valeurs clés de ce fichier. Notez que certaines valeurs de ce fichier peuvent ne s'adresser qu'à l'équipe de développement ou être liées à un autre produit (un produit de PDV par exemple).

URL et numéro de port du serveur LDAP

Si plusieurs serveurs LDAP sont utilisés, cette entrée peut contenir le serveur principal, le serveur secondaire, etc. Le détaillant définit le nom du serveur ou l'adresse IP.

Exemple :

```
#URL et numéro de port du serveur ldap
#PRIMARY_LDAP_URL=ldap://<NomServeurLDAP ou AdresseIP>:389
PRIMARY_LDAP_URL =ldap://10.6.1.21:389
BACKUP_LDAP_URL_1=ldap://10.6.1.21:389
BACKUP_LDAP_URL_2=ldap://10.6.1.21:389
```

Détails du schéma LDAP

Les valeurs de cette entrée doivent correspondre aux entrées du serveur LDAP. DN est l'abréviation de domain name (nom de domaine). CN est l'abréviation de company name (nom de l'entreprise).

Exemple :

```
#Détails du schéma LDAP
BASE_DN=dc=rettek,dc=com
#APPLICATION_LOGIN=cn=<NomConnexionUtilLDAP>,dc=rettek,dc=com
APPLICATION_LOGIN=cn=Retek,dc=rettek,dc=com
#APPLICATION_PASSWORD=<MotdepasseConnexionUtilLDAP>
APPLICATION_PASSWORD=rettek
```

Contexte des données utilisateur

Les valeurs de cette entrée doivent correspondre aux entrées du serveur LDAP. Notez que la valeur de retekAppName est 'rss'. La valeur du paramètre SUB_CONTEXT permet l'utilisation de plusieurs applications au sein d'une même entreprise.

Exemple :

```
#Contexte des données utilisateur
SUB_CONTEXT=rettekAppName=rss,cn=Retek
RETEK_APP_NAME=rss
#CN=<NomConnexionUtilLDAP>
CN=Retek
```

Fichier de configuration Client master (clientmaster.cfg)

Ce fichier contient des informations relatives au client SIM. Notez que certaines valeurs de ce fichier peuvent ne s'adresser qu'à l'équipe de développement ou être liées à un autre produit (un produit de PDV par exemple).

rmsupload.cfg

Pour consulter la description de ce fichier, reportez-vous à l'«Annexe A – Spécifications de mise en page du fichier de comptage».

Classe initiale exécutée

Ces paramètres sont associés au traitement système initial. Lors d'une utilisation normale de l'application, ces paramètres ne doivent *pas* être modifiés.

Exemple :

```
# La classe exécutée est toujours la classe initiale. Cette classe
pourrait
# être utilisée pour mettre à jour ce fichier.

STARTUP_BOOT_STRAP=com.retek.iso.cr.appmgr.bootstrap.InitialBootStrap

# nom de la classe, délimité par des virgules, qui doit être exécutée
# avant le chargement des applications.

BOOT_STRAP=com.retek.iso.cr.appmgr.bootstrap.ClientLockingBootStrap,com.retek.iso.cr.appmgr.bootstrap.ClientServicesBootStrap,com.retek.iso.cs.util.StartupBrowserBootStrap,com.retek.iso.cs.swing.GUIBootStrap
```

Journalisation

Ce paramètre contient des informations sur la journalisation relatives au client SIM. Le détaillant peut spécifier le nom du fichier journal. Le système utilise par défaut le répertoire spécifié et un même client peut avoir différents fichiers journaux. La valeur du paramètre LOGGING_PAUSE indique au système le temps d'attente avant d'écrire des données dans le fichier journal.

Exemple :

```
#journalisation

LOGGING_IMPL=com.retek.iso.cr.logging.LoggingFileServices

# en l'absence de commentaire sur cette ligne, ce nom est utilisé comme
# nom de fichier journal client à la place du nom contenant l'horodatage

#LOGGING_FILE_NAME=..\log\sim_client.log

LOGGING_LEVEL=4

LOGGING_PAUSE=5000

LOGGING_SYSTEM_OUT=true

LOGGING_SYSTEM_ERR=true
```

Paramètres relatifs à l'aide en ligne

Les valeurs définies dans cette section déterminent la façon dont le système gère l'aide en ligne. Dans le cadre d'une utilisation normale de l'application, ces valeurs ne devraient *pas* avoir à être modifiées.

Exemple :

```
# Détermine si l'application autorise l'utilisateur final à activer la
#..fonction 'Etablir le mode d'aide' ENABLE_BUILD_HELP_MODE=false

# Définit la première page du système d'aide comme répertoire de base.
# L'application essaie d'ajouter les paramètres régionaux de l'utilisateur
# après cette valeur de façon à ce que la documentation s'affiche
# dans sa langue (si la documentation existe.) Si l'application
# ne parvient pas à trouver la documentation dans la langue de
# l'utilisateur, elle utilise le répertoire de base tel qu'il est
# (sans ajouter les valeurs des paramètres régionaux).
# Cette valeur est associée à la valeur HELP_ENTRY_PAGE.
# Cette valeur peut être :
# 1. un chemin d'accès relatif. Dans ce cas, le chemin est considéré comme
# relatif à partir du répertoire files/prod ou files/training.
# Ainsi, une valeur "help" correspond en fait à
# "<rss-dir>files/prod/help-<user-locale>" ou
# "<rss-dir>files/prod/help."
# 2. Un chemin d'accès absolu. Ex : "y:\\rss-help"
# 3. une adresse URL complète. Incluez la section du protocole,
# sinon le système risque de ne pas fonctionner. Ex :
# "http://ourhelpmachine"
HELP_BASE_DIR=OnlineHelp/WebHelp

# Voir les commentaires sur HELP_BASE_DIR pour plus de détails sur # les
# paramètres régionaux de l'utilisateur.
# Cette valeur indique à l'application l'emplacement de la page
# d'entrée pour la documentation.
HELP_ENTRY_PAGE=iso_im.htm

# Cette valeur contient le fichier exécutable permettant d'afficher l'aide.
# Paramètres disponibles : {0} - chemin d'accès absolu de l'URL qui
# doit afficher la documentation de l'aide.
# (1) - répertoire de base absolu de l'application.
#Cela peut permettre de trouver les fichiers suivants.
#HELP_EXECUTABLE=rundll32 url.dll,FileProtocolHandler {0}
HELP_EXECUTABLE=cmd /c start {0}
#HELP_EXECUTABLE=wscript /nologo {1}OnlineHelp/IEStarter.js {0}
```

Liste des classes ResourceBundles

Les classes ResourceBundles sont des fichiers Java relatifs à l'internationalisation. Cette entrée détermine les classes ResourceBundles par défaut du système. Les versions spécifiques à des langues sont gérées automatiquement en fonction des paramètres régionaux de l'utilisateur. Ces valeurs ne doivent *pas* être modifiées même lorsque le système est proposé dans plusieurs langues.

Exemple :

```
# Liste, délimitée par des virgules, de classes ResourceBundles
MESSAGE_BUNDLE=com.retek.iso.cs.util.MessageBundle,com.retek.iso.cs.util.La
belResourceBundle,com.retek.iso.cs.util.ButtonResourceBundle,com.retek.iso.
cs.util.RSSRuleMessageBundle,com.retek.iso.cs.util.RendererResourceBundle,c
om.retek.iso.cs.util.EMailTextResourceBundle,com.retek.iso.cs.util.RSSRibR
sourceBundle
```

Apparence du client

Ces paramètres permettent au détaillant de personnaliser l'apparence de l'application sur l'ordinateur. Cependant, Retek ne recommande *pas* de modifier ces valeurs.

Exemple :

```
# Essai d'utilisation de l'une des apparences suivantes, de LF1 à
# LFn.
LOOK_FEEL_KEY=LF1,LF2,LF3,LF4,LF5
LF1=com.retek.iso.cr.swing.plaf.metal.ConfigurableMetalLookAndFeel
LF2=com.apple.mrj.swing.MacLookAndFeel
LF3=com.sun.java.swing.plaf.windows.WindowsLookAndFeel
LF4=javax.swing.plaf.metal.MetalLookAndFeel
LF5=com.sun.java.swing.plaf.motif.MotifLookAndFeel

# Emplacement de la barre d'outils et de la barre d'application
# 0 - En bas à droite
# 1 - En bas à gauche
# 2 - En haut à droite
# 3 - En haut à gauche
USER_PREFERENCES.ANCHOR=0
# Affiche une page d'accueil
SHOW_SPLASH=FALSE
# Temps d'attente, en millisecondes, avant qu'un bouton ne soit activé
# sur un applet qui vient de s'afficher
# Valeur conseillée : 200 (deux dixièmes de seconde)
BUTTON_DELAY=0
# Définit si le texte du bouton doit s'afficher au format HTML. Utile
# pour le renvoi à la ligne automatique.
FORMAT_BUTTONS_WITH_HTML=false
```

Port permettant de verrouiller l'application client

SIM n'utilise *pas* directement cette valeur. Le détaillant peut changer la valeur du port si le port spécifié est utilisé à d'autres fins sur son système.

Exemple :

```
# Port permettant de verrouiller l'application client. Ce port
# est vérifié à chaque lancement du client pour s'assurer qu'une seule
# instance de cette application est exécutée.

CLIENT_LOCK_PORT=51802
```

Services client chargés par amorces

Ces paramètres sont associés au traitement système initial des services client. Dans le cadre d'une utilisation normale de l'application, ces paramètres ne devraient *pas* avoir à être modifiés.

Exemple :

```
#Services client chargés par amorces

SERVICES_LIST=CONFIG_SRVC,EMPLOYEE_SRVC,AUTH_SRVC,LOCALITEM_SRVC,
MDSEHIERARCHY_SRVC,MINISTORE_SRVC,PRINTING_SRVC,REGISTER_SRVC,SOU
RCE_SRVC,STOCKCOUNT_SRVC,STOCKCOUNTEVENT_SRVC,STOCKEVENT_SRVC,STO
RE_SRVC,TXN_NUMBER_SRVC,TXN_POSTER_SRVC,ACTIVITYLOCKING_SRVC,PROC
ESSMEASUREAUDIT_SRVC,SEQUENCING_SRVC,REPLENISHMENT_SRVC,PRODUCTGR
OUP_SRVC,ADHOCOUNTADMIN_SRVC,STOREORDER_SRVC,PRICECHANGE_SRVC,ST
OCKCOUNTEVENTLOCATION_SRVC,REPORTING_SRVC
```

Reporting.cfg

URL de création de rapports par défaut de l'entreprise

Ces paramètres facilitent l'accès au site Web hébergeant Crystal Enterprise afin que les utilisateurs de SIM puissent employer la fonctionnalité Crystal Enterprise. L'adresse de l'outil de rapports dit au système d'envoyer les utilisateurs à la page d'accueil. La requête de l'outil de rapports est un paramètre interne lié à l'emplacement du serveur Web du côté Crystal Enterprise. Les valeurs susceptibles de changer sont le nom de l'ordinateur, du port, de l'utilisateur, ainsi que le mot de passe.

Exemple :

```
# URL de création de rapports par défaut de l'entreprise
REPORTING_TOOL_ADDRESS=http://msppc003648/crystal/enterprise10/ePortfolio
REPORTING_TOOL_REQUEST_URL=http://msppc003648:8080/retекCrystal/integration
/request_report.jsp?authType=secEnterprise&username=administrator&password=
&apsname=msppc003848
```

Modèles des types de rapports

Ces valeurs correspondent aux rapports fonctionnels 'prêts à l'emploi' qui sont livrés avec SIM. Chaque rapport est lié à une zone fonctionnelle basée sur les données que le système SIM est capable de générer.

Exemple :

```
# Valeurs par défaut de l'entreprise pour chaque type de rapport
REPORT_FORMAT_DIRECTDELIVERY=DirectDeliveryReport
REPORT_FORMAT_ITEMREQUEST=ItemRequestReport
REPORT_FORMAT_PICKLIST=PickListReport
REPORT_FORMAT_RETURN=ReturnReport
REPORT_FORMAT_STOCKCOUNT=StockCountReport
REPORT_FORMAT_STOCKRECOUNT=StockCountReCountReport
REPORT_FORMAT_STOREORDER=StoreOrderReport
REPORT_FORMAT_TRANSFER=TransferReport
REPORT_FORMAT_WAREHOUSEDELIVERY=WarehouseDeliveryReport
```

Emplacement des modèles

Cette valeur correspond au répertoire du serveur dans lequel le détaillant stocke ses modèles d'étiquetage. Pour chaque type d'imprimante, il existe un type de sous-répertoire concordant avec l'ID de type de l'imprimante définie dans la base de données.

Exemple :

```
# Répertoire des modèles de formats d'étiquette
TICKET_TEMPLATE_DIR=../files/prod/template
```

Paramètre du pilote d'imprimante

Ce paramètre s'applique aux imprimantes d'étiquettes Monarch. Pour les autres types d'imprimantes d'étiquettes, les détaillants doivent changer cette valeur.

Exemple :

```
# Pilotes d'imprimantes pour les imprimantes non PostScript
PRINTER_DRIVER_FOR_TYPE_2=com.retek.iso.cs.report.CMSRawTicketPrinterDriver
```

Wireless_services.cfg

Pour faciliter la connexion des périphériques sans fil, les détaillants doivent connecter le serveur sans fil à un port spécifique. Ils doivent en outre attribuer un numéro de port qui n'est pas encore utilisé.

Exemple :

```
# Paramètres sans fil
PORT=40002
```

Informations de journalisation

Pour plus d'informations sur la journalisation relative au traitement par lots DexnexFileParser, reportez-vous au "Chapitre 7 - Traitements par lots JAVA".

Emplacement par défaut des fichiers journaux client et serveur

Unix

Les fichiers journaux relatifs au client se trouvent dans le répertoire suivant :

```
clientUnix\rettek\sim\files\prod\log
```

Les fichiers journaux relatifs au serveur se trouvent dans le répertoire suivant :

```
serverUnix\rettek\sim\files\prod\log
```

Windows

Les fichiers journaux relatifs au client se trouvent dans le répertoire suivant :

```
clientWindows\rettek\sim\prod\log
```

Les fichiers journaux relatifs au serveur se trouvent dans le répertoire suivant :

```
serverWindows\rettek\sim\files\prod\log
```

Niveaux de journalisation établis dans les fichiers de configuration (.cfg)

Les niveaux de journalisation peuvent être établis dans des fichiers de configuration (par exemple, stockcount.cfg, store.cfg et activitylocking.cfg, etc). Ce niveau de journalisation est utile lors du dépannage de certaines parties de l'application. Par exemple, si l'application rencontre des problèmes dans un certain domaine, le responsable du fichier journal peut définir un niveau élevé de granularité. Le fichier stockcount.cfg contient par exemple l'entrée suivante :

```
# Journalisation
LOGGING_IMPL=com.retek.iso.cr.logging.LoggingFileServices
LOGGING_FILE_NAME=../log/sim_stockcount.log
LOGGING_LEVEL=4
LOGGING_PAUSE=5000
LOGGING_SYSTEM_OUT=true
LOGGING_SYSTEM_ERR=true
```

Gestion des exceptions

Dans le système SIM, les principaux types d'exceptions sont les suivants :

- `com.retek.iso.cr.rules.BusinessRuleException`
Cette exception indique qu'une règle commerciale a été enfreinte.
- `com.retek.iso.cr.appmgr.ApplicationException`
Cette exception permet de regrouper les exceptions de bas niveau du système qui surviennent dans le logiciel. Cette classe doit être utilisée pour indiquer ce dont l'application a besoin pour gérer cette exception avec une logique applicative, permettant ainsi aux exceptions de bas niveau qui surviennent d'échouer et d'être gérées différemment.
- `com.retek.iso.cs.dataaccess.DAOException`
Une exception `DAOException` est une exception signalée par un objet d'accès aux données (DAO). Lorsqu'un code détecte une exception dans un DAO, il doit renvoyer une exception `DAOException`. Cette classe permet à l'exception d'origine d'être toujours accessible.
- `com.retek.iso.cr.activitylocking.ActivityLockingException`
Cette classe d'exception prolonge l'exception `BusinessRuleException` et fournit une API afin d'indiquer la chaîne identifiant le détenteur actuel du verrouillage.

Options de machine virtuelle Java (JVM)

Tout modificateur d'instruction JVM (c'est-à-dire pouvant être spécifié dans le programme 'Java') peut être spécifié dans le fichier RSSMainContainer.xml.

Par exemple, si des problèmes de manque de mémoire surviennent à cause du nombre de données utilisées lors de l'exécution, le paramètre JVM peut être modifié dans la section du fichier RSSMainContainer.xml que vous trouverez ci-après. Notez que dans cet exemple, 128m signifie 128 méga-octets. Il s'agit de la valeur à augmenter.

Exemple :

```
<jvmLineArgs length="2">  
    <java.lang.String>-Xms4m</java.lang.String>  
    <java.lang.String>-Xmx128m</java.lang.String>  
</jvmLineArgs>
```


Chapitre 3 – Architecture technique

Ce chapitre décrit l'architecture générale du logiciel SIM, offrant une vue d'ensemble de la structure générale du système, y compris des différentes couches de code Java. Ces informations sont notamment utiles dans les situations suivantes :

- lorsque le détaillant souhaite profiter des possibilités extensibles de SIM et écrire son propre code pour l'intégrer au système SIM.
- lorsque le détaillant souhaite implémenter le système pour différentes bases de données (Oracle, DB2, etc.).

Pour les utilisateurs ayant une connaissance réduite de la terminologie Java, une description de normes et de termes Java relatifs à SIM est fournie en référence à la fin de ce chapitre.

Présentation

SIM et ISO (Integrated Store Operations)

ISO est un groupe d'applications d'opérations en magasin qui utilisent et partagent une logique applicative, des utilitaires de configuration et une infrastructure technologique. Il est prouvé que les applications simplifient les opérations en magasin en améliorant le service clientèle et en diminuant les coûts. SIM est une application de la plate-forme ISO.

Avantages de l'architecture

La plate-forme informatique distribuée robuste de SIM améliore les performances et permet la hiérarchisation.

L'architecture multi-niveaux de SIM permet l'encapsulation de logique applicative, évitant au client de faire face à la complexité du système en arrière-plan. Chaque niveau de l'architecture n'a pas à se préoccuper des tâches de fonctionnement internes des autres niveaux.

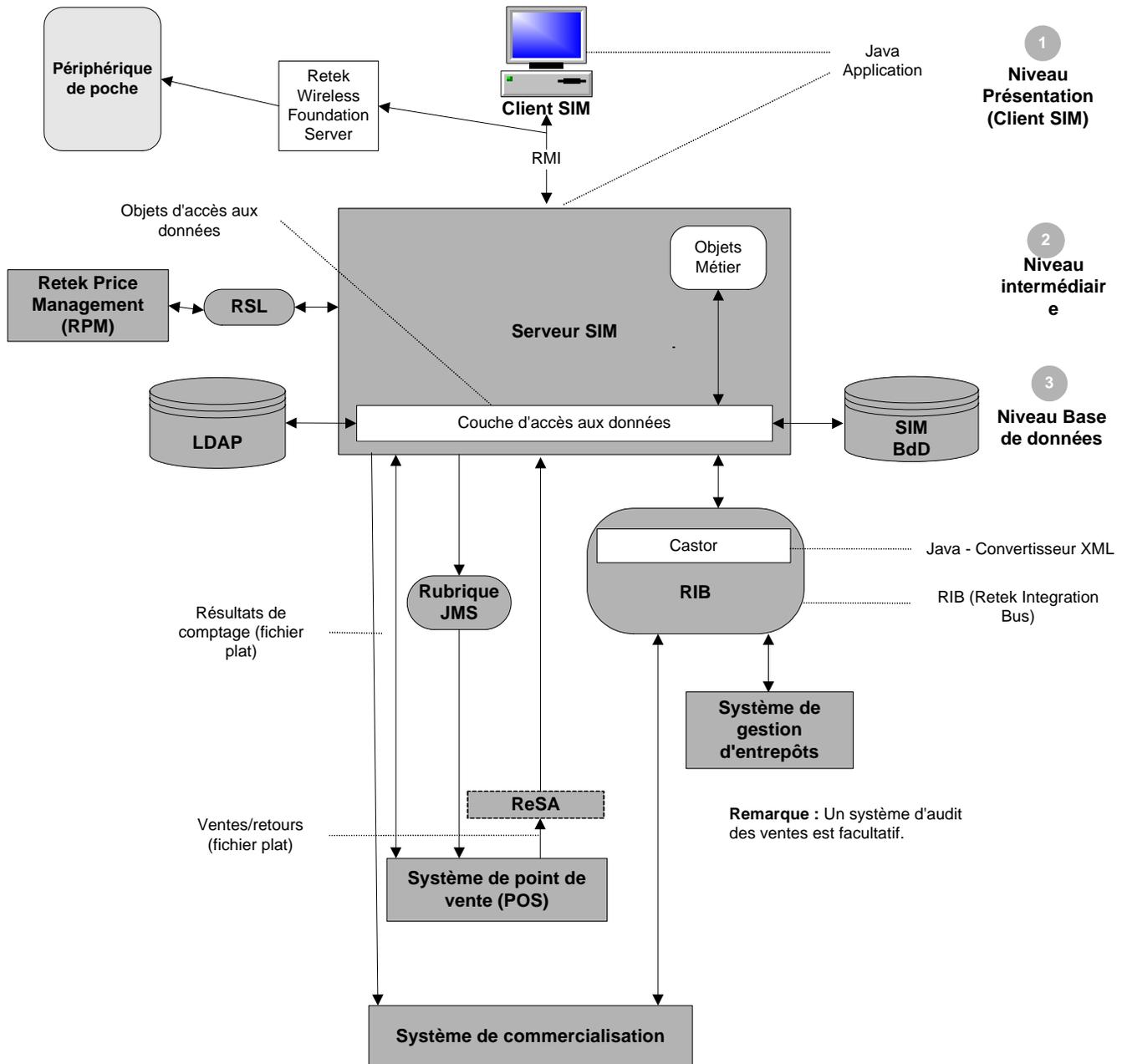
La liste suivante est un résumé des avantages que présente l'architecture multi-niveaux utilisée par SIM.

- **Hiérarchisation** : Du matériel et des logiciels peuvent être ajoutés afin de répondre aux besoins du détaillant pour chaque niveau.
- **Facilité de maintenance** : La séparation de la présentation, de la logique applicative et des données clarifie le logiciel et facilite sa gestion et sa modification.
- **Indépendance de la plate-forme** : Le code est écrit une seule fois mais peut être exécuté sous tout système pouvant utiliser Java.
- **Rentabilité** : La technologie open source utilisée a fait ses preuves sur le marché, tandis que la conception orientée objet améliore les possibilités de réutilisation, permettant un développement et un déploiement plus rapides.
- **Facilité d'intégration** : La réutilisation d'objets-métier et de fonctions accélère l'intégration aux sous-systèmes d'entreprise. L'architecture multi-niveaux est devenue une norme du secteur.
- **Grande disponibilité** : La couche intermédiaire est conçue pour un environnement à grappes ou un serveur lame peu coûteux.
- **Longévité** : Une architecture multi-niveaux physiquement distribuée prolonge la durée de vie du système.
- **Flexibilité** : Le système alloue les ressources de façon dynamique, en fonction de la charge.

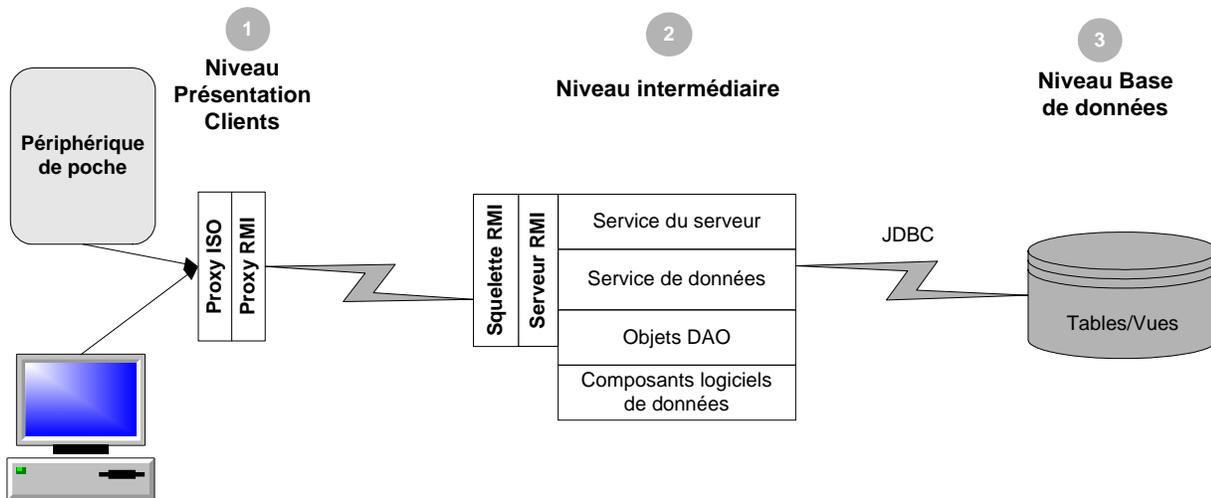
Diagrammes et description de l'architecture technique de SIM

Cette section vous offre une vue d'ensemble de l'architecture technique de SIM. Les diagrammes suivants illustrent les principaux éléments d'une implémentation SIM traditionnelle à trois niveaux. Les numéros des éléments dans les deux diagrammes renvoient à des descriptions ci-après.

Pour plus d'informations sur l'intégration technique, reportez-vous au « Chapitre 4 - SIM, le RIB, JMS et RSL ». Consultez également le guide des opérations RPOS.



Architecture technique de SIM au sein de l'entreprise



Architecture technique du logiciel SIM

Niveau Présentation

- 1 SIM peut être déployé sur une large gamme de clients, par exemple un ordinateur de bureau ou un périphérique de poche. L'interface utilisateur présente les données à l'utilisateur et reçoit directement des données provenant de l'utilisateur via le système « frontal ». Le niveau Présentation interagit uniquement avec le niveau intermédiaire (et non avec le niveau Base de données). Pour optimiser les performances, le système frontal de SIM facilite un traitement côté client robuste.

Le côté ordinateur de SIM est conçu sur une architecture de client lourd développée à l'aide de Swing, boîte à outils utilisée pour créer des interfaces graphiques riches dans des applications Java.

L'élément d'infrastructure de communication de poche, appelé Retek Wireless Foundation Server, permet aux périphériques de poche de communiquer. Les périphériques de poche communiquent avec le serveur Retek Wireless Foundation Server, qui à son tour agit comme client du niveau intermédiaire de SIM.

Niveau intermédiaire

- 2 En tant que lien entre le client SIM et la base de données, le niveau intermédiaire gère quasiment l'intégralité du traitement de la logique applicative qui survient dans l'architecture multi-niveaux de SIM. Le niveau intermédiaire est composé de services, dont la majorité est associée aux fonctions commerciales. Par exemple, un service d'articles s'occupe des articles, etc. Dans SIM, les objets-métier sont des composants logiciels (c'est-à-dire des classes Java qui ont un ou plusieurs attributs et méthodes set / get correspondantes) qui représentent une entité fonctionnelle. En d'autres termes, les objets-métier peuvent être perçus comme des conteneurs de données qui seuls n'ont quasiment pas de fonctionnalités commerciales.

Bien que les clients sur ordinateur et sur périphérique de poche utilisent différemment la fonctionnalité du niveau intermédiaire, celui-ci est le même pour les deux clients. Par exemple, le périphérique de poche, utilisé « à la volée », effectue des soumissions fréquentes dans la base de données, tandis que l'ordinateur de bureau en effectue moins souvent. L'application est flexible car elle accepte les différents styles de traitement issus du client.

Le niveau intermédiaire est conçu pour fonctionner comme un objet « sans état », c'est-à-dire qu'il reçoit toutes les instructions nécessaires pour accéder à la base de données depuis le client. De plus, SIM possède des capacités de basculement, la très grande majorité des données transférées à l'un des conteneurs de la base de données n'étant jamais mise en mémoire cache ni stockée au niveau intermédiaire entre les interactions des clients. Cela signifie que si le serveur d'un niveau intermédiaire spécifique s'arrête, le traitement peut basculer sur un autre serveur SIM afin que le traitement continue.

De plus, si la charge le justifie, SIM peut être étendu de façon verticale en ajoutant des serveurs d'applications supplémentaires. Les serveurs SIM s'exécutant sur plusieurs serveurs d'applications dans un système sans état, la plate-forme n'a pas besoin de configurations distinctes pour chaque serveur d'applications. Ainsi, les détaillants SIM ne se rendent jamais compte que des serveurs d'applications ont été ajoutés pour répondre à la charge. Les serveurs d'applications SIM comportent plusieurs conteneurs, chacun d'entre eux étant lié à une seule machine virtuelle Java (JVM). Chaque conteneur peut avoir plusieurs instances d'un composant et chaque composant correspond à un service SIM spécifique. En introduisant plusieurs instances d'un conteneur ou d'un service, les détaillants SIM peuvent répartir le traitement entre plusieurs conteneurs de façon plus efficace. Ils étendent ainsi la plate-forme de façon horizontale. Au fur et à mesure que la charge de requêtes d'un service augmente, des instances de ce service sont automatiquement créées afin de faire face à la charge accrue.

Afin de contacter la base de données SIM de manière efficace et sûre, le niveau intermédiaire comprend les composants principaux suivants : Les services du serveur contiennent la logique applicative appropriée.

- Les objets DAO ne sont associés à aucune base de données, permettant ainsi l'indépendance de la base de données.
- Les services de données « communiquent » avec les composants logiciels de données appropriés et créent les objets nécessaires.
- Les composants logiciels de données qui créent le code SQL nécessaire sont utilisés pour extraire des données de la base de données. Notez qu'une relation de type un-à-un existe entre les composants logiciels de données et les tables ou vues de la base de données.

Objets d'accès aux données (DAO)

Les DAO sont des classes qui analysent le mécanisme de persistance utilisé pour rendre persistants les objets-métier. Les objets DAO fournissent le mécanisme permettant à SIM d'être associé à un autre moteur de persistance. Le scénario idéal dans ce cas est lorsque seul le DAO a besoin d'être modifié suite à ce changement. Le reste du système SIM n'est pas modifié.

Connectivité JDBC

Le niveau intermédiaire communique avec la base de données via le protocole standard JDBC (Java database connectivity). Pour que le client SIM puisse extraire de la base de données les informations souhaitées, une connexion JDBC doit être établie entre le niveau intermédiaire et la base de données. Le protocole JDBC facilite la communication entre une application Java et une base de données relationnelle. JDBC est en fait un ensemble d'interfaces API (Application Programming Interface) qui offre une méthode, indépendante de la base de données, permettant d'extraire des données du système SIM ou d'en insérer. Pour insérer ou extraire ces données, le code SQL se trouve également à ce niveau, facilitant ainsi les opérations de création, lecture, mise à jour et suppression. Pour obtenir une définition de JDBC, reportez-vous à la section « Normes et termes Java relatifs à SIM » de ce chapitre.

Niveau Base de données



Remarque : certaines tables et colonnes du modèle de données SIM sont spécifiques à SIM et d'autres sont nommées d'après le modèle de données de l'association ARTS (Association for Retail Technology Standards). Notez cependant que SIM utilise la norme ARTS mais n'y est pas totalement conforme.

- 3 Le niveau Base de données constitue la plate-forme de stockage de l'application. Elle contient les données physiques (utilisateur et système) utilisées dans toute l'application. La base de données héberge les tables, vues et enregistrements qui sont extraits par le conteneur puis transmis au client. Notez que la base de données ne génère pas de code PL/SQL supplémentaire qui lui est transmis via JDBC. L'objectif de ce niveau n'est pas de manipuler ni de fournir des données, mais uniquement de prendre en charge le stockage et l'extraction des informations. Ce niveau répond aux requêtes, il ne les crée pas.

SIM prend en charge plusieurs types de bases de données et de mécanismes de persistance (par exemple, Oracle et DB2).

Topologie distribuée

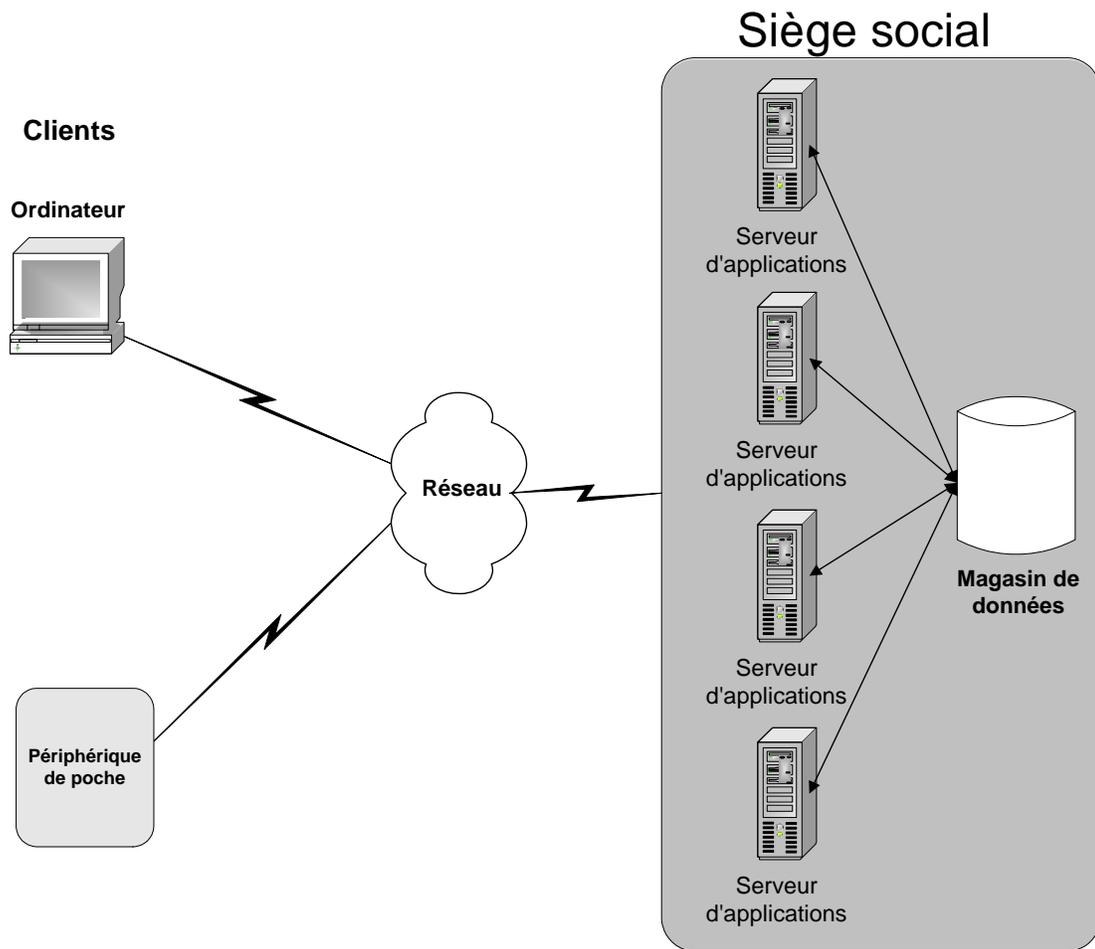
L'un des avantages les plus notables de SIM est sa topologie distribuée flexible. SIM offre une transparence totale des emplacements, l'emplacement des données et des services étant basé sur les besoins du détaillant et non sur des limitations techniques. La connexion du serveur client de SIM utilise le protocole RMI (Remote Method Invocation). L'utilisation du protocole RMI permet à l'application SIM de tirer parti des objets distribués. Ceci signifie que le serveur peut être distribué de façon géographique, tout en se trouvant à un emplacement centralisé. Le serveur n'étant pas obligé de se trouver au même endroit que les clients en magasin, les clients se connectent au serveur « à distance ».

Le proxy RMI de SIM se trouve du côté client dans la relation client/serveur. Le proxy RMI ne contient pas de logique applicative mais il contient juste assez de code pour « transmettre » des données de façon efficace. En ce qui concerne les clients, l'intégralité du traitement s'effectue localement car ils interagissent uniquement avec le proxy, qui est local. L'interface graphique utilise les classes d'assistant pour « communiquer » avec le proxy.

Par exemple, si le proxy, à la demande du client, doit effectuer la méthode « Mise à jour de l'expédition », le proxy RMI semble disposer de cette fonctionnalité, même si, en réalité, il ne sert que de passage vers le squelette du RMI, qui se trouve côté serveur. Le squelette « communique » avec le serveur afin d'effectuer le traitement de logique applicative. Le serveur effectue la logique applicative « Mise à jour de l'expédition ».

La connectivité entre le client SIM et le niveau intermédiaire s'effectue via un service RNS (Remote Naming Service), qui fournit au client SIM l'adresse IP et le port nécessaires. Le service RNS, ou couche intermédiaire de la plate-forme SIM, sert essentiellement de pont de communication entre les clients et le serveur d'applications.

Le diagramme suivant illustre le déploiement de SIM.



Déploiement de SIM

Paquetages contenant des services et des objets-métier clés

Les paquetages suivants, à afficher dans Javadoc, contiennent des classes qui font partie intégrante du traitement commercial du système. Ces classes contiennent à la fois les méthodes de logique applicative et les méthodes de recherche. Elles sont situées au niveau intermédiaire.

- `com.retek.iso.cr.group`
- `com.retek.iso.cr.localitem`
- `com.retek.iso.cr.mdsehierarchy`
- `com.retek.iso.cr.print`
- `com.retek.iso.cr.replenishment`
- `com.retek.iso.cr.rules`
- `com.retek.iso.cr.sequencing`
- `com.retek.iso.cr.stock.activity`
- `com.retek.iso.cr.stock.adhoc`
- `com.retek.iso.cr.stock.count`
- `com.retek.iso.cr.stock.countgroup`
- `com.retek.iso.cr.stockevent`
- `com.retek.iso.cr.store`
- `com.retek.iso.cr.swing`
- `com.retek.iso.cr.pricechange`
- `com.retek.iso.cr.storeorder`
- `com.retek.iso.cs.activitylocking`
- `com.retek.iso.cs.group`
- `com.retek.iso.cs.localitem`
- `com.retek.iso.cs.mdsehierarchy`
- `com.retek.iso.cs.print`
- `com.retek.iso.cs.replenishment`
- `com.retek.iso.cs.rules`
- `com.retek.iso.cs.sequencing`
- `com.retek.iso.cs.source`
- `com.retek.iso.cs.stock.count`
- `com.retek.iso.cs.stock.countgroup`
- `com.retek.iso.cs.stock.event`
- `com.retek.iso.cs.stockevent`
- `com.retek.iso.cs.store`
- `com.retek.iso.cs.swing`
- `com.retek.iso.cs.pricechange`
- `com.retek.iso.cs.storeorder`

Présentation du verrouillage des activités

Le verrouillage des activités a été conçu pour être contrôlé depuis SIM. L'exemple suivant illustre la logique derrière le verrouillage des activités.

Un utilisateur est impliqué dans une livraison entrepôt qui concerne des conteneurs comportant plusieurs articles. Le traitement en va-et-vient entre l'écran et le serveur est alors très important. Depuis l'interface utilisateur, un appel vers le verrouillage des activités est effectué. Il indique au système que l'utilisateur travaille actuellement sur la livraison entrepôt. Si un autre utilisateur détient le verrouillage, le système demande à l'utilisateur s'il souhaite déverrouiller l'élément et en prendre la propriété. Une réponse positive signifie que l'ancien propriétaire du verrouillage a laissé celui-ci en suspens sans raison valable (il est par exemple en pause déjeuner). Une réponse négative signifie que l'ancien propriétaire du verrouillage a toujours besoin de ce verrouillage pour terminer le traitement.

Services de support technique

Les services techniques maintiennent l'application en place en fournissant des services communs à l'application. Ces services ne sont pas nécessairement entraînés par des besoins commerciaux. Afin d'améliorer la maintenance du code, un certain nombre de services techniques de base sont fournis dans SIM. Ces services sont décrits ci-dessous.

Service de transaction

La couche de service de transaction de SIM est chargée de fournir un support de transaction de soumission en deux phases. Grâce à ce traitement, les changements de couches de persistance sont soit tous soumis soit tous annulés.

Service de journalisation

Ce service fournit au système une méthode standard de journalisation des informations dans un fichier texte plat.

Service d'internationalisation

Ce service utilise les classes Java avec une extension .java pour permettre la configuration des messages à l'écran, tels que les libellés ou les messages d'erreur. Pour changer la langue des écrans de l'interface SIM, le détaillant peut modifier ces fichiers .java et les recompiler, sans affecter la fonctionnalité commerciale de l'application. Notez que bien que ce service prenne en charge un certain nombre de langues, l'écran se lit toujours de gauche à droite et de haut en bas. SIM utilise la prise en charge standard des ResourceBundles de Java.

Notez que les paramètres régionaux d'un utilisateur sont spécifiés dans sa configuration LDAP. La langue préférée d'un utilisateur est donc associée à son nom d'utilisateur et *non* à l'ordinateur.

Service de sécurité

Le service de sécurité fournit la fonctionnalité de base d'autorisation et d'authentification lors de la connexion de l'utilisateur. L'association de l'utilisateur à des rôles de sécurité permet de contrôler l'accès de l'utilisateur aux parties fonctionnelles de l'application. Le service de sécurité valide l'identité d'un utilisateur à l'aide d'un magasin de sécurité. Il extrait les membres et les autorisations du rôle de cet utilisateur en cas de connexion réussie. L'implémentation physique des informations de sécurité pour chaque utilisateur, rôle, autorisation fonctionnelle et de champs peut être configurée indépendamment à l'emplacement du serveur LDAP.

Sécurité et protocole LDAP (Light Directory Access Protocol)

La norme LDAP définit un protocole réseau permettant d'accéder aux informations d'un répertoire.

LDAP est le mécanisme de persistance utilisé par SIM pour stocker les informations d'authentification et d'autorisation. SIM lit les informations utilisateur standard depuis un serveur LDAP. Le navigateur LDAP permet de configurer les utilisateurs dans le système.

La sécurité dans SIM comprend les deux domaines suivants :

- Authentification : détermine si l'utilisateur est autorisé à utiliser le système.
- Autorisation : détermine les droits ou rôles de l'utilisateur et indique les magasins ou zones où l'utilisateur peut opérer.

Un serveur LDAP peut disposer de plusieurs schémas (l'un pointant vers SIM, l'autre vers POS, etc). Le fichier de configuration définit l'association entre les schémas et les serveurs LDAP. Reportez-vous au chapitre 2, « Configuration du système d'arrière-plan ».

La configuration du protocole LDAP s'effectue dans le navigateur LDAP. Le détaillant définit les utilisateurs auxquels sont attribués des rôles (desquels dépendent les droits d'activité) et auxquels sont associés des magasins. Les rôles apparaissent dans l'interface utilisateur de l'application SIM. Notez que les magasins configurés avec LDAP doivent exister dans la table de magasins correspondante de la base de données SIM.

Un magasin et des rôles valides doivent être attribués aux utilisateurs. Le système valide un nom d'utilisateur et un mot de passe saisis en fonction de ce qui a été établi. Une fois que le système a accepté l'utilisateur et vérifié ses rôles, il limite les éléments affichés en fonction du rôle de cet utilisateur. Notez qu'un « superutilisateur » possède tous les droits.

Ressources LDAP supplémentaires

- <http://www.openldap.org/>
Cette adresse ouvre la page principale (en anglais) de OpenLDAP. Ce site contient une introduction et propose des téléchargements et de la documentation.

Normes et termes Java relatifs à SIM

SIM est déployé à l'aide des technologies, méthodes, versions et/ou modèles de conception définis dans cette section.

Objet d'accès aux données (DAO)

Ce modèle de conception isole l'accès aux données et la logique de persistance. Le reste du composant peut ainsi ignorer les détails de persistance (par exemple la version ou le type de base de données).

Instancier

Dans le contexte de SIM, « instancier » signifie créer une instance de classe.

Java Development Kit (JDK), version 1.4.1

Outils de développement Java standard de Sun Microsystems.

Rubrique JMS (Java Messaging Service)

Une rubrique JMS fait partie de la couche intermédiaire de Retek, orientée message. SIM utilise une vue JMS du RIB. La rubrique peut être perçue comme la diffusion d'un message depuis le RIB. SIM (et éventuellement d'autres abonnés) peut s'abonner à cette diffusion.

JDBC

JDBC est un moyen pour les applications disposant d'une architecture Java, telles que SIM, d'exécuter des instructions SQL sur une base de données compatible SQL, telle qu'Oracle. Faisant partie de la spécification J2EE de Sun, la majorité des fournisseurs de bases de données implémente cette spécification.

Conventions de nommage dans Java

- Paquetages : le préfixe d'un nom unique de paquetage est en lettres minuscules.
- Classes : ces noms descriptifs ne sont pas abrégés et comportent à la fois des lettres majuscules et minuscules.
- Interfaces : ces noms descriptifs ne sont pas abrégés et comportent à la fois des lettres majuscules et minuscules. La première lettre de chaque mot interne est en majuscules.
- Méthodes : le premier mot des méthodes est un verbe en minuscules. La première lettre de chaque mot interne est en majuscules.

Persistance

Protocole permettant de transférer l'état d'un objet-métier entre des variables et une base de données sous-jacente.

Connexions persistantes

Etat de la connexion entre une application et la base de données.

Interface distante

Interface côté client d'un service. Cette interface définit les méthodes côté serveur disponibles au niveau client.

RMI (Remote Method Invocation)

Le protocole RMI offre un modèle simple et direct permettant d'effectuer des calculs distribués avec des objets Java. Le protocole RMI invoque une méthode d'interface distante sur un objet distant. Important : la syntaxe d'une invocation de méthode sur un objet distant est identique à la syntaxe d'une invocation de méthode sur un objet local.

Squelette

Le squelette comprend comment communiquer avec le proxy par l'intermédiaire du lien RMI. Le squelette :

- « communique » avec le proxy ;
- lit les paramètres de l'appel de méthode depuis le lien ;
- appelle l'objet d'implémentation du service distant, accepte la valeur retournée, puis renvoie cette valeur au proxy.

Proxy

Le proxy contient des informations qui permettent la connexion à un objet distant, comprenant l'implémentation des méthodes. Le proxy implémente le même ensemble d'interfaces distantes que la classe de l'objet distant.

Chapitre 4 – SIM, le RIB, JMS et RSL

Ce chapitre, qui décrit les méthodes d'intégration de Sim, est divisé en deux parties :

- Intégration RIB (Retek Integration Bus)
- Intégration RSL (Retek Service Layer)

Chaque partie contient des informations sur l'architecture de la méthode d'intégration et les données transmises. Pour obtenir une description supplémentaire du flux de données, reportez-vous au chapitre 5, « Flux de données de l'interface d'intégration ».

Présentation de SIM et du RIB

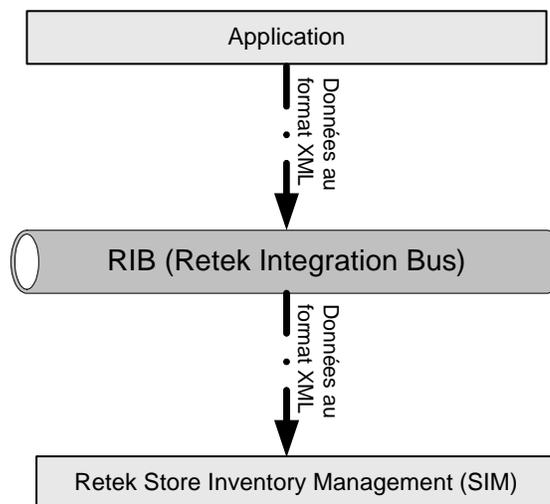
Les diagrammes et les explications de ce chapitre présentent rapidement les processus de publication et d'abonnement.

Retek utilise un paradigme de messagerie de type publication-abonnement avec une garantie quant à la livraison d'un message. Dans ce système de messagerie, un adaptateur publie un message sur le bus d'intégration. Ce message est ensuite transmis à un ou plusieurs abonnés. L'adaptateur de publication ne connaît pas le nombre d'abonnés qui attendent ce message, le type d'adaptateur des abonnés, leur état actuel (en cours d'exécution ou non) ni leur situation géographique et ne souhaite pas vraiment le savoir. La livraison du message à tous les adaptateurs abonnés est la responsabilité du bus d'intégration.

Pour plus d'informations, consultez le dernier guide d'intégration Retek (Retek Integration Guide) et la documentation relative au RIB.

Format des messages XML

Comme décrit dans le diagramme suivant, les messages auxquels s'abonne SIM sont au format XML. La structure de leurs données est définie par les DTD (Document Type Definitions) ou les documents de schéma XML.



Données du RIB au format XML

Processus d'abonnement aux messages

Diagramme du processus d'abonnement aux messages

Dans l'ensemble, tous les processus d'abonnement aux messages de SIM suivent le même modèle de base. Le diagramme et les explications numérotées ci-après présentent brièvement ce processus d'abonnement. Notez que l'explication utilise comme exemple l'abonnement de SIM à un message simple Diff comprenant un enregistrement Diff.

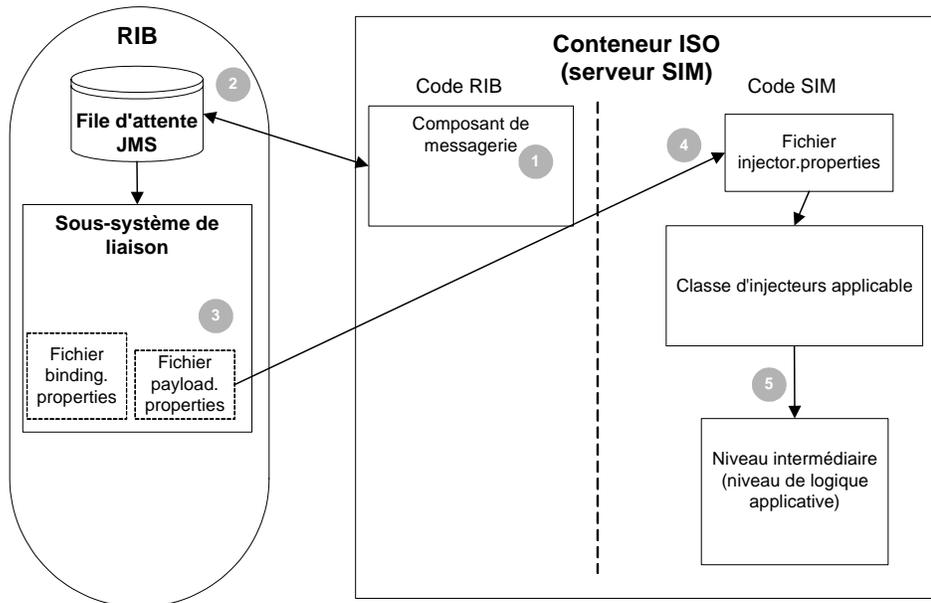


Diagramme du processus d'abonnement aux messages de SIM

Description du processus d'abonnement aux messages

- 1 Le composant de messagerie est issu de classes Java SIM et est déployé dans le conteneur ISO (le serveur SIM). Cependant, le code pour un composant de messagerie est la « propriété » du RIB.

Le code de composant de messagerie est le même pour toutes les familles et tous les types de messages. Lorsque le composant du message est déployé, il contient uniquement les entrées et les fichiers de configuration. Le fichier RIBContainer.xml détermine l'élément exécuté dans le conteneur. Ce fichier comprend toutes les instances d'informations qui incluent ce composant de messagerie et le nom du fichier de configuration. Un récapitulatif de ces informations est fourni plus loin dans ce chapitre.

- 2 Dans le RIB, un message est remis à une rubrique JMS à laquelle SIM est abonné. Lorsque la rubrique JMS (par exemple EtDiffsFromRMS) apprend qu'il existe un message (relatif à diff) au format XML, le composant du message basé sur Java (DiffsMessagingComponent par exemple) est notifié afin de récupérer le message. Le service JMS (Java Messaging Service) du RIB continue le processus. Pour obtenir une définition de la rubrique JMS, reportez-vous à la section « Normes et termes Java relatifs à SIM » du chapitre 3, « Architecture technique ».

- 3 Faisant référence à la fois aux fichiers `binding.properties` et `payload.properties`, la liaison utilise un utilitaire d'analyse syntaxique XML afin d'analyser le XML et de transformer les trois parties du message RIB (la famille, le type et la charge utile) en charge utile (objet) Java que le niveau intermédiaire de SIM peut interpréter. Le terme « liaison » désigne un sous-système de traitement de messages utilisé pour les environnements ISO et J2EE de Retek. Ce sous-système comprend un ensemble de classes utilisées pour traiter ou créer des messages RIB. La liaison est chargée de traduire la chaîne XML de charge utile du message RIB en objet « charge utile », puis d'exécuter le code spécifique d'application afin de traiter la charge utile. Autrement dit, la liaison transforme le message XML en objet Java.



Remarque : dans des conditions d'exécution normales, les fichiers de propriétés relatifs à l'intégration et les fichiers de configuration décrits auparavant n'ont pas besoin d'être modifiés.

- 4 En fonction de la famille et du type de message, un fichier `injector.properties` de SIM sait quelle classe java d'abonnement (appelée « injecteurs » dans le code) doit être instanciée. La classe d'injecteur implémente une interface d'injecteur de messagerie de l'application. Notez qu'un injecteur peut gérer plusieurs messages `.dtd`. Pour obtenir une définition du verbe « instancier », reportez-vous à la section « Normes et termes Java relatifs à SIM » du chapitre 3, « Architecture technique ».
- 5 L'abonné « injecte » les données dans le niveau intermédiaire de l'application, configuré pour corriger et/ou valider les données. Lorsque SIM reçoit des données du système de commercialisation via le RIB, il se comporte comme si le RIB était un client du système. Le RIB devient une interface système de SIM (c'est-à-dire que le RIB devient une interface sans interface graphique). SIM traite les données du RIB de la même façon qu'il le ferait pour des données entrées par un utilisateur dans l'interface graphique.

Messages d'erreur RIB relatifs à l'abonnement

Pour chaque message auquel s'abonne SIM, l'injecteur gère le message de son point d'entrée dans le système SIM à son arrivée à la base de données. Si un message échoue lors du processus relatif à l'injecteur et qu'une exception est renvoyée, différents messages peuvent apparaître, notamment :

- CREATE_FAILED
- UNMARSHALLING_ERROR
- COMMAND_FACTORY_CANNOT_INSTANTIATE_PAYLOAD
- MODIFY_FAILED

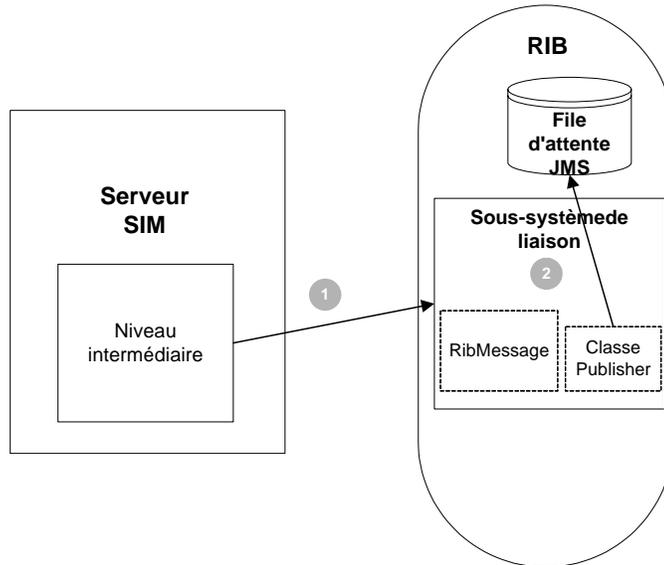
RIB Error Hospital

Le programme RIB Error Hospital est un ensemble de classes Java et de tables de bases de données situé sur le serveur SIM mais dont le RIB est « propriétaire ». Il est conçu pour isoler et déclencher de nouveau le traitement de messages ayant provoqué une erreur au cours du traitement initial. Le but est d'arrêter le traitement des messages provoquant des erreurs tout en permettant de continuer à traiter les « bons » messages. Le programme RIB Error Hospital fait référence aux tables de SIM (par exemple RIB_MESSAGE, RIB_MESSAGE_FAILURE ou RIB_MESSAGE_ROUTING_INFO). Pour plus d'informations sur le programme RIB Error Hospital, consultez les derniers guides RIB Technical Architecture Guide et RIB Operations Guide ou l'aide en ligne RIB Hospital Administration.

Processus de publication des messages

Dans l'ensemble, tous les processus de publication des messages de SIM suivent le même modèle de base. Le diagramme et les explications numérotées ci-après présentent rapidement ce processus de publication.

Diagramme du processus de publication des messages



Processus de publication des messages SIM

Description du processus de publication des messages

- 1 Suite à un événement dans SIM (à savoir une insertion, une mise à jour ou une suppression), le niveau intermédiaire de SIM renseigne un objet de charge utile et appelle le sous-système de liaison.
- 2 Le sous-système de liaison convertit l'objet de charge utile en chaîne XML. L'objet du sous-système de liaison est placé dans une enveloppe RIB appelée RibMessage. Les données de RibMessage deviennent par la suite un message dans le RIB. Une classe Publisher du sous-système de liaison est appelée pour écrire les données dans la file d'attente JMS du RIB. A une fréquence régulière (configurable), le RIB scrute la file d'attente JMS à la recherche d'un message existant. Un message publiable qui apparaît dans la file d'attente de JMS est traité.

Tableau de mappage des abonnements

Le tableau suivant répertorie la famille et le type de message, le DTD (Document Type Definition) décrivant le message XML et les classes d'abonnement permettant d'entrer des données dans le niveau intermédiaire de l'application. Ces classes sont décrites dans le code en tant qu'« injecteurs ». Pour plus d'informations, consultez le dernier guide d'intégration Retek (Retek Integration Guide) et la documentation relative au RIB.

Famille	Type	DTD/Charge utile	Classe d'abonnement (« injecteur »)
ASNIN	ASNOUTCRE	ASNInDesc	ASNInCreateInjector
ASNIN	ASNINDEL	ASNInRef	ASNInRemoveInjector
ASNIN	ASNINMOD	ASNInDesc	ASNInModifyInjector
ASNOUT	ASNOUTCRE	ASNOutDesc	ASNOutCreateInjector
CLRPRCCHG	CLRPRCCHGCRE	ClrPrcChgDesc	ClrPrcChgCreateInjector
CLRPRCCHG	CLRPRCCHGMOD	ClrPrcChgDesc	ClrPrcChgModifyInjector
CLRPRCCHG	CLRPRCCHGDEL	ClrPrcChgRef	ClrPrcChgRemoveInjector
DIFFS	DIFFCRE	DiffDesc	DifferentiatorCreateInjector
DIFFS	DIFFDEL	DiffRef	DifferentiatorRemoveInjector
DIFFS	DIFFMOD	DiffDesc	DifferentiatorModifyInjector
ITEMS	ITEMBOMCRE	ItemBOMDesc	ItemBOMCreateInjector
ITEMS	ITEMBOMDEL	ItemBOMRef	ItemBOMRemoveInjector
ITEMS	ITEMBOMMOD	ItemBOMDesc	ItemBOMModifyInjector
ITEMS	ITEMCRE	ItemDesc	ItemCreateInjector
ITEMS	ITEMDEL	ItemRef	ItemRemoveInjector
ITEMS	ITEMHDRMOD	ItemHdrDesc	ItemModifyInjector
ITEMS	ITEMSUPCRE	ItemSupCtyDesc	ItemSupCreateInjector
ITEMS	ITEMSUPCTYCRE	ItemSupCtyRef	ItemSupCtyCreateInjector
ITEMS	ITEMSUPCTYDEL	ItemSupCtyRef	ItemSupCtyRemoveInjector
ITEMS	ITEMSUPCTYMOD	ItemSupCtyDesc	ItemSupCtyModifyInjector
ITEMS	ITEMSUPDEL	ItemSupRef	ItemSupRemoveInjector
ITEMS	ITEMSUPMOD	ItemSupDesc	ItemSupModifyInjector
ITEMS	ITEMUPCCRE	ItemUPCDesc	ItemUPCCreateInjector
ITEMS	ITEMUPCDEL	ItemUPCRef	ItemUPCRemoveInjector
ITEMS	ITEMUPCMOD	ItemUPCDesc	ItemUPCModifyInjector
ITEMZONEPRC	ITEMZONEPRCCRE	ItemZonePrcDesc	ItemZonePrcCreateInjector

Famille	Type	DTD/Charge utile	Classe d'abonnement (« injecteur »)
ITEMZONEPRC	ITEMZONEPRCDEL	ItemZonePrcRef	ItemZonePrcRemoveInjector
ITEMZONEPRC	ITEMZONEPRCMOD	ItemZonePrcDesc	ItemZonePrcModifyInjector
ORDER	POCRE	PODesc	PurchaseOrderCreateInjector
ORDER	PODEL	PORef	PurchaseOrderRemoveInjector
ORDER	PODTLCRE	PODesc	PurchaseOrderDetailCreateInjector
ORDER	PODTLDEL	PORef	PurchaseOrderDetailRemoveInjector
ORDER	PODTLMOD	PODesc	PurchaseOrderDetailModifyInjector
ORDER	POHDRMOD	PODesc	PurchaseOrderModifyInjector
PRCCHGCONF	PRCCHGCONFCRE	PrcChgConfDesc	PrcChgConfCreateInjector
PRCZONEGRP	PRCZONEGRPCRE	PrcZoneGrpDesc	PrcZoneGrpCreateInjector
PRCZONEGRP	PRCZONEGRPDEL	PrcZoneGrpRef	PrcZoneGrpRemoveInjector
PRCZONEGRP	PRCZONEGRPMOD	PrcZoneGrpDesc	PrcZoneGrpModifyInjector
PRMPRCCHG	PRMPRCCHGCRE	PrmPrcChgDesc	PrmPrcChgCreateInjector
PRMPRCCHG	PRMPRCCHGMOD	PrmPrcChgDesc	PrmPrcChgModifyInjector
PRMPRCCHG	PRMPRCCHGDEL	PrmPrcChgRef	PrmPrcChgRemoveInjector
REGPRCCHG	REGPRCCHGCRE	RegPrcChgDesc	RegPrcChgCreateInjector
REGPRCCHG	REGPRCCHGMOD	RegPrcChgDesc	RegPrcChgModifyInjector
REGPRCCHG	REGPRCCHGDEL	RegPrcChgRef	RegPrcChgRemoveInjector
RCVUNITADJMOD	RCVUNITADJDTL	RcvUnitAdjDesc	RcvUnitAdjModInjector
RTVREQ	RTVREQCRE	RTVReqDesc	RTVReqCreateInjector
RTVREQ	RTVREQMOD	RTVReqDesc	RTVReqModifyInjector
RTVREQ	RTVREQDEL	RTVReqRef	RTVReqRemoveInjector
RTVREQ	RTVREQDTLCRE	RTVReqDesc	RTVReqDetailCreateInjector
RTVREQ	RTVREQDTLDEL	RTVReqRef	RTVReqDetailRemoveInjector
RTVREQ	RTVREQDTLMOD	RTVReqDesc	RTVReqDetailModifyInjector
SEEDDATA	DIFFTYPEPCRE	DiffTypeDesc	DifferentiatorTypeCreateInjector
SEEDDATA	DIFFTYPEPEDEL	DiffTypeRef	DifferentiatorTypeRemoveInjector
SEEDDATA	DIFFTYPEPEMOD	DiffTypeDesc	DifferentiatorTypeModifyInjector
STOCKORDER	SOCRE	SODesc	StockOrderCreateInjector
STOCKORDER	SODTLCRE	SODesc	StockOrderCreateInjector
STOCKORDER	SODTLDEL	SORef	StockOrderRemoveInjector

Retek Store Inventory Management

Famille	Type	DTD/Charge utile	Classe d'abonnement (« injecteur »)
STOCKORDER	SODTLMOD	SODesc	StockOrderModifyInjector
STOCKORDER	SOHDRDEL	SORef	StockOrderRemoveInjector
STOCKORDER	SOHDRMOD	SODesc	StockOrderModifyInjector
STORES	STORECRE	StoresDesc	StoreCreateInjector
STORES	STOREDEL	StoresRef	StoreRemoveInjector
STORES	STOREMOD	StoresDesc	StoreModifyInjector
VENDOR	VENDORADDRCRE	VendorAddrDesc	SupplierAddrCreateInjector
VENDOR	VENDORADDRDEL	VendorAddrRef	SupplierAddrRemoveInjector
VENDOR	VENDORADDRMOD	VendorAddrDesc	SupplierAddrModifyInjector
VENDOR	VENDORCRE	VendorDesc	SupplierCreateInjector
VENDOR	VENDORDEL	VendorRef	SupplierRemoveInjector
VENDOR	VENDORHDRMOD	VendorHdrDesc	SupplierModifyInjector
WH	WHCRE	WHDesc	WareHouseCreateInjector
WH	WHDEL	WHRef	WareHouseRemoveInjector
WH	WHMOD	WHDesc	WareHouseModifyInjector

Tableau de mappage des publications

Ce tableau illustre les relations entre la famille et le type de message et l'objet DTD/charge utile créé par l'application. Pour plus d'informations, consultez le dernier guide d'intégration Retek (Retek Integration Guide) et la documentation relative au RIB.

Famille	Type	DTD/Charge utile
ASNOUT	ASNOUTCRE	ASNOutDesc
DSDRECEIPT	DSDRECEIPTCRE	DSDReceiptDesc
INVADJUST	INVADJUSTCRE	InvAdjustDesc
INVREQ	INVREQCRE	InvReqDesc
PRCCHGREQ	PRCCHGREQCRE	PrcChgReqDesc
RECEIVING	RECEIPTCRE	ReceiptDesc
RECEIVING	RECEIPTMOD	ReceiptDesc
RTV	RTVCRE	RTVDesc
SOSTATUS	SOSTATUSCRE	SOStatusDesc
STKCOUNTSCH	STKCOUNTSCHCRE	StkCountSchDesc
STKCOUNTSCH	STKCOUNTSCHDEL	StkCountSchRef
STKCOUNTSCH	STKCOUNTSCHMOD	StkCountSchDesc

Descriptions fonctionnelles des messages

Le tableau suivant décrit rapidement le rôle fonctionnel des messages vis-à-vis de la fonctionnalité SIM. Ce tableau indique également si SIM publie le message sur le RIB ou s'il s'abonne à ce message auprès du RIB. Pour plus d'informations, consultez le dernier guide d'intégration Retek (Retek Integration Guide) et la documentation relative au RIB.

Zone fonctionnelle	Abonnement/ publication	Intégration aux produits	Description
AE entrant	Abonnement	RWMS, Fournisseur (externe)	Ces messages contiennent des notifications d'expédition entrante provenant des fournisseurs (commandes) et des entrepôts (transferts et allocations).
AE sortant	Publication	RMS, RWMS	SIM utilise ces messages pour communiquer à RMS et RWMS les transferts de magasin à entrepôt (retours à l'entrepôt). Ces messages lui permettent également de communiquer à RMS les transferts entre magasins.
N° de Diff	Abonnement	RMS	Ces messages sont utilisés pour communiquer les numéros de différenciateurs de RMS à SIM.
Réceptions de livraison directe au magasin	Publication	RMS	SIM utilise ces messages pour communiquer la réception d'une livraison fournisseur pour laquelle aucune commande RMS n'existait auparavant.
Articles	Abonnement	RMS	Ces messages communiqués par RMS contiennent tous les enregistrements d'articles approuvés, y compris les informations d'en-tête, l'article/fournisseur, les détails d'article/fournisseur/pays et les informations d'article/étiquette.
Article/site	Abonnement	RMS	Ces messages communiqués par RMS contiennent les données d'article/site utilisées pour ordonnancer les articles dans les sites et communiquer certains paramètres de niveau d'article/site utilisés lors de commandes magasin.

Zone fonctionnelle	Abonnement/ publication	Intégration aux produits	Description
Article/zone/prix	Abonnement	RMS	Ces messages communiqués par RMS contiennent tous les enregistrements d'article/zone/prix. Ces données sont utilisées par SIM pour qu'un magasin connaisse le prix de vente d'un article avant que celui-ci ne soit ordonnancé dans le site.
Ajustements de stock	Publication	RMS	SIM utilise ces messages pour communiquer les ajustements de stock. RMS utilise ces messages pour ajuster les stocks en conséquence.
Demande d'inventaire	Publication	RMS	SIM utilise ces messages pour communiquer la demande d'inventaire pour un article spécifique. RMS utilise ces données pour répondre à cette demande, soit en effectuant un réapprovisionnement automatique, soit en créant un transfert ou une commande unique.
Modification de prix	Abonnement	RPM	Ces messages facilitent les modifications des prix de type Permanent, Soldes et Promotion.
Commandes	Abonnement	RMS	Ces messages contiennent les commandes Direct magasin approuvées. SIM les utilise pour recevoir des livraisons directes.
Groupes de zones de prix	Abonnement	RMS	Ces messages communiqués par RMS contiennent les hiérarchies de groupe de zones de prix/zone/magasin. Ces données sont utilisées avec les enregistrements d'article/zone/prix pour déterminer le prix de vente d'un article dans un site.
Réceptions	Publication	RMS	SIM utilise ces messages pour communiquer la réception d'une commande, d'un transfert ou d'une allocation RMS.

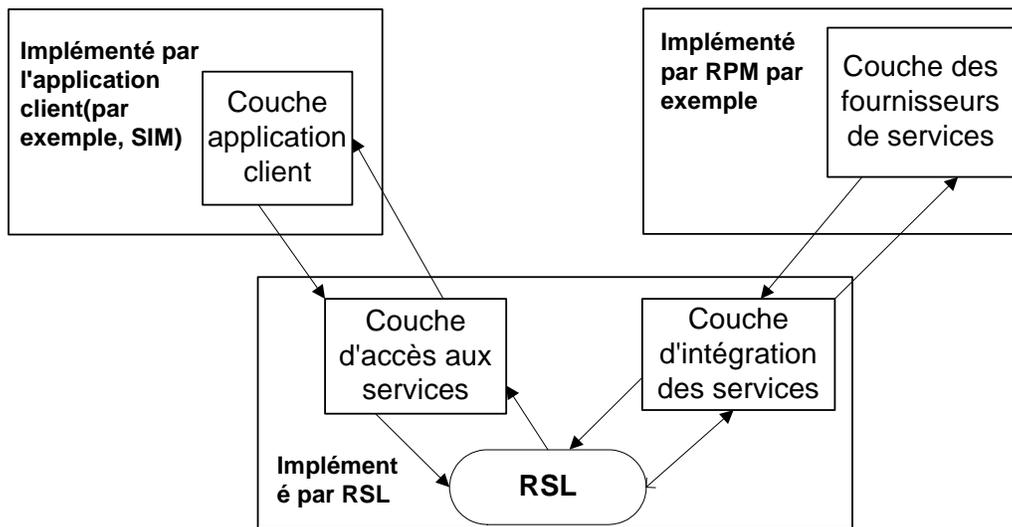
Zone fonctionnelle	Abonnement/ publication	Intégration aux produits	Description
Ajustements d'unités lors de la réception	Publication	RMS	SIM utilise ces messages pour communiquer tous les ajustements à la réception de commandes, transferts et allocations. Ces messages font partie de la famille RECEIVING (les ajustements d'unités lors de la réception utilisent uniquement le type de message RECEIPTMOD).
Retour fournisseur	Publication	RMS	SIM utilise ces messages pour communiquer l'envoi d'un retour fournisseur depuis le magasin.
Demande de RF	Abonnement	RMS	Ces messages communiqués par RMS contiennent une demande de retour d'inventaire à un fournisseur.
Données d'origine	Abonnement	RMS	Ces messages communiqués par RMS contiennent les valeurs du type de différentiateur.
Plannings de comptage	Publication	RMS	SIM utilise ces messages pour communiquer à RMS les plannings de comptage Unité et valeur. RMS utilise ce planning afin de répliquer les données d'inventaire pour le jour d'un comptage planifié.
Statut de l'ordre de réapprovisionnement	Publication	RMS	SIM utilise ces messages pour communiquer l'annulation de toutes les quantités de transfert demandées. Le système de commercialisation peut par exemple créer une demande de transfert de 90 unités depuis un magasin. Si le magasin expéditeur n'envoie que 75 unités, un message d'annulation est envoyé pour les 15 unités restantes.
Magasins	Abonnement	RMS	Ces messages communiqués par RMS contiennent les magasins configurés dans le système (RMS).
Commandes magasin	Publication	RMS	SIM utilise ces messages pour communiquer la demande d'inventaire pour un article spécifique.

Zone fonctionnelle	Abonnement/publication	Intégration aux produits	Description
Demande de transfert	Abonnement	RMS	Ces messages communiqués par RMS contiennent une demande de transfert du stock d'un magasin. Après expédition du transfert demandé, SIM utilise le message d'AE sortant pour indiquer le contenu de l'expédition. De plus, SIM utilise le message de statut de l'ordre de réapprovisionnement pour annuler toute quantité demandée non expédiée.
Entrepôts	Abonnement	RMS	Ces messages communiqués par RMS contiennent les entrepôts configurés dans le système (RMS). Les enregistrements de SIM ne concernent que les entrepôts physiques.
Fournisseur	Abonnement	RMS, externe (financier)	Ces messages communiqués par RMS contiennent les fournisseurs configurés dans le système (RMS ou système financier externe).

Présentation de SIM et RSL

RSL est un logiciel qui permet aux applications Retek d'exposer des interfaces API à d'autres applications Retek. Comme l'illustre le diagramme ci-dessous, il existe deux couches dans RSL, la « couche application client » et la « couche fournisseur de service ». RPM inclut la « couche fournisseur de service » qui est propriétaire de la logique applicative.

L'implémentation de RSL expose une méthode *synchrone* pour communiquer avec les autres applications (le traitement par le RIB est asynchrone). Tous les services RSL sont rassemblés dans une interface offerte par un Stateless Session Bean (SSB). Chaque service ressemble à un simple appel de méthode pour les applications client.



Traitement application client et fournisseur de service via RSL

Services RSL utilisés par SIM

Pour toute information spécifique sur le traitement des requêtes et réponses liées aux services ci-dessous, reportez-vous à « Message Families and Types Report » de la documentation sur l'intégration de Retek.

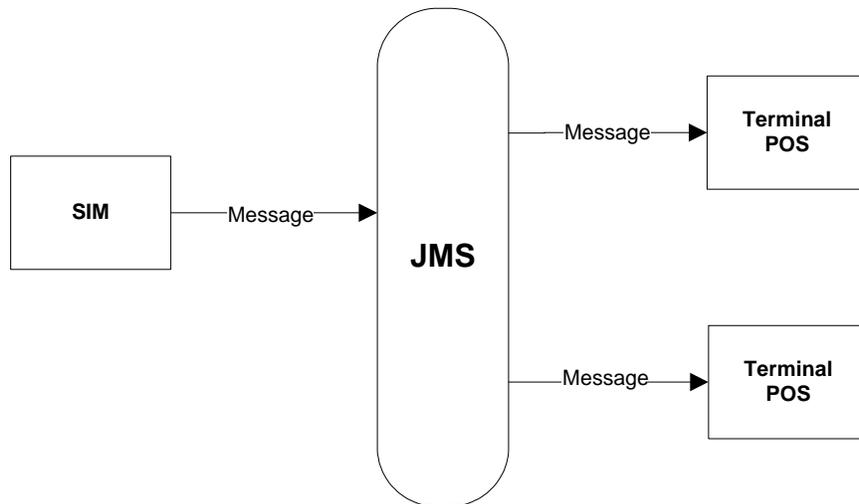
SIM n'utilise pas RSL de la même manière avec RPM et RMS. La différence majeure est la suivante :

- SIM copie les données de RPM dans son système.
- SIM ne copie *pas* les données de RMS dans son système.

Nom du service	Description
PriceInquiryService	Ce service, fourni par RPM, permet à un système de requête de demander le prix au détail effectif d'un article se trouvant dans un site spécifique, à une date donnée. RPM fournit la valeur unitaire et indique s'il s'agit d'une promotion, de soldes ou d'un prix standard.
PriceChangeService	Ce service permet de créer une modification de prix dans RPM pour Permanent, Soldes ou Promotion.
ExternalStoreOrderService	SIM appelle RMS pour la création et la demande d'une commande externe. Aux questions viennent s'ajouter les requêtes/réponses concernant la création, la modification et la suppression de commandes magasin.

SIM, RPOS et le logiciel de messagerie JMS

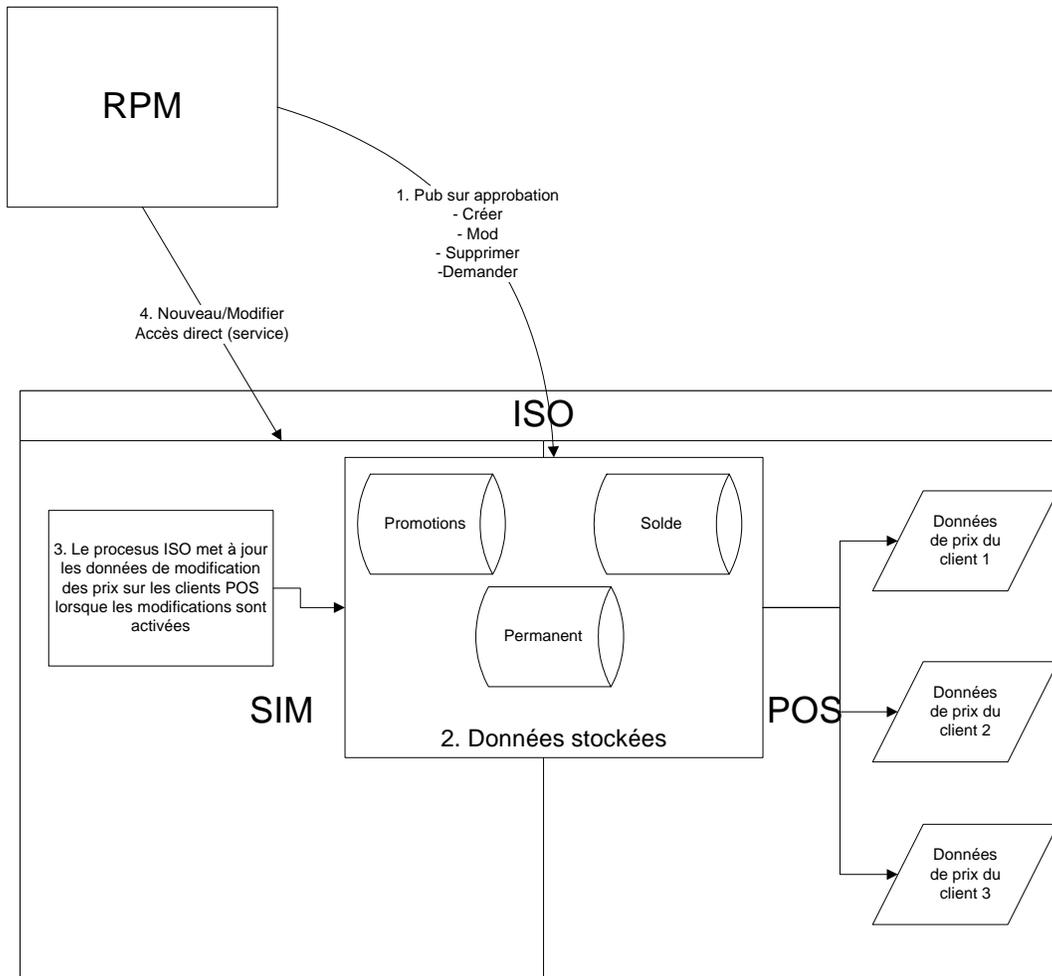
SIM utilise la fonctionnalité de messagerie pour introduire les mises à jour de prix de la journée (originaires de RPM) dans RPOS. Comme l'illustre le diagramme ci-dessous, le logiciel de messagerie permet aux clients RPOS de recevoir des messages de rubriques et files d'attente situées sur un serveur JMS. Les modifications de prix sont publiées par SIM dans la rubrique PRICE_UPDATE. Les clients RPOS reçoivent la même rubrique JMS, ainsi RPOS reçoit immédiatement les modifications de prix. Pour en savoir plus, consultez le guide des opérations RPOS.



Scénario de messagerie dans RPOS

Résumé de l'intégration de l'interface RPM-SIM

Le diagramme ci-dessous récapitule l'intégration des systèmes SIM, RPOS et RPM.



Chapitre 5 – Flux de données de l'interface d'intégration

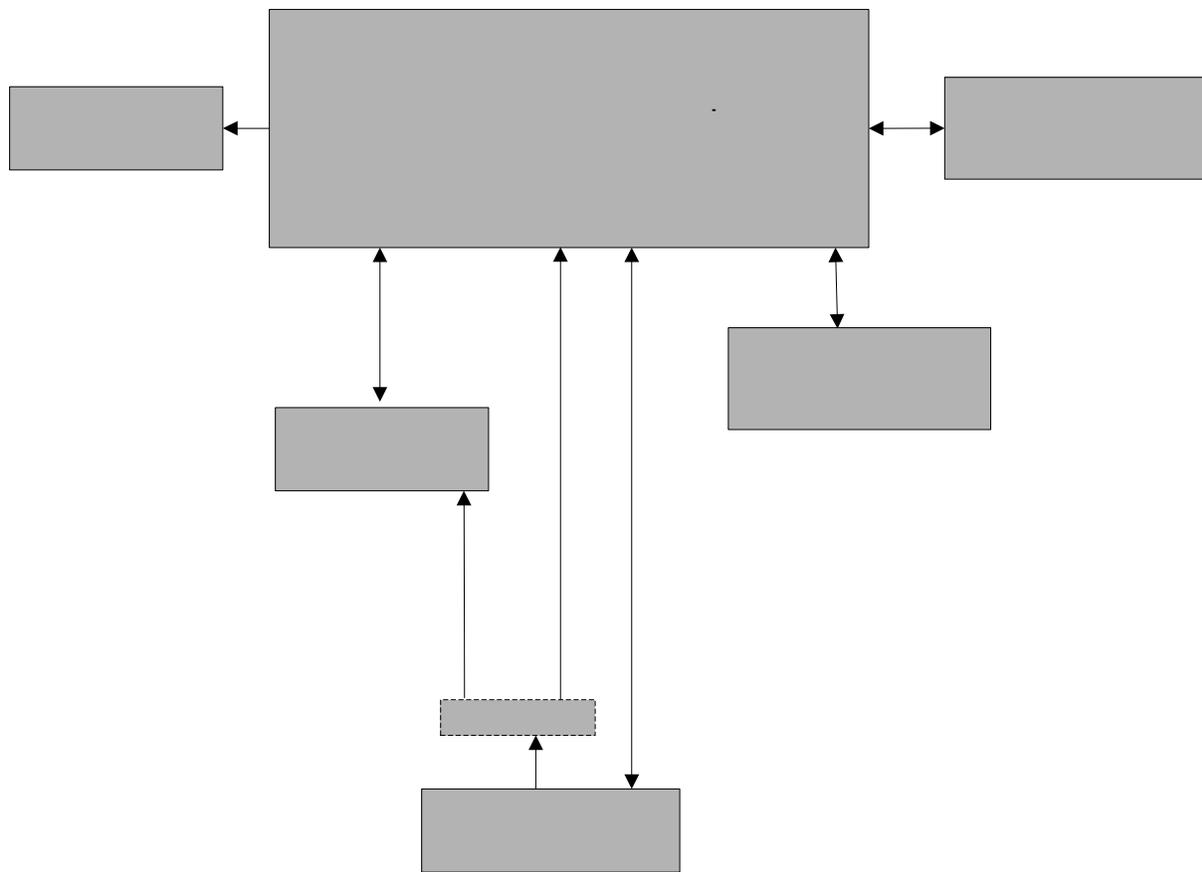
Ce chapitre contient une présentation fonctionnelle de la méthode d'intégration de SIM à d'autres systèmes, notamment d'autres systèmes Retek.

Présentation

La première section de ce chapitre présente un diagramme illustrant les différents produits et bases de données Retek avec lesquels SIM communique, ainsi que le flux de données global entre les produits. Les explications accompagnant le diagramme sont rédigées dans une perspective inter-systèmes mettant en valeur le déplacement de données.

Pour en savoir plus sur les méthodes utilisées par ces interfaces pour transmettre des données, reportez-vous au chapitre 3, « Architecture technique », au chapitre 4, « SIM, le RIB, JMS et RSL » et au chapitre 7, « Traitements par lots JAVA ».

Flux de données SIM inter-systèmes



Flux de données relatif à SIM au sein de l'entreprise

Reporting system
(such as Crystal
Reports)

De SIM à WMS (Warehouse Management System)

- Retours à l'entrepôt
Via le RIB, SIM envoie les données d'AE sortantes pour communiquer à WMS les données d'expédition du magasin à l'entrepôt.

De WMS à SIM

Les données WMS suivantes sont publiées via le RIB pour l'abonnement de SIM :

- Données d'avis d'expédition (AE) sortantes converties en données d'AE entrantes
Pour faciliter les expéditions de l'entrepôt au magasin, WMS fournit à SIM des données d'AE sortantes. Les avis d'expédition sont associés aux expéditions et contiennent des informations sur les sites expéditeurs et destinataires, sur les quantités des conteneurs, etc. Les données d'AE sortantes sont converties en données d'AE entrantes par le RIB pour l'abonnement de SIM. Les données sont identiques mais le format est légèrement différent. La conversion se produit de telle sorte que les données d'AE entrantes peuvent être les mêmes sur différentes applications.

D'un système de point de vente (RPOS par exemple) à SIM



Remarque : RPOS et SIM partageant les mêmes modèles et services de données, la section ci-dessous qui illustre les données fonctionnelles transmises entre les systèmes ne représente qu'une infime partie de la totalité des échanges de données.

Les données suivantes sont envoyées d'un système de point de vente (POS) à SIM via ReSA (facultatif) :

- Données de ventes et retours
SIM utilise les données pour mettre à jour le stock disponible en fonction des combinaisons magasin/articles. En d'autres termes, SIM obtient des informations sur le déplacement du stock (ventes et retours).

De SIM à un système de point de vente (RPOS par exemple)

- Données de mise à jour des prix
SIM peut introduire les mises à jour de prix de la journée (originaires de RPM) dans RPOS.

Du système de commercialisation à SIM

Les données du système de commercialisation suivantes sont publiées via le RIB pour l'abonnement de SIM :

- Données de la commande
SIM permet à l'utilisateur de recevoir des données de commande relatives à des livraisons directes au magasin. La livraison directe au magasin se produit lorsque le fournisseur dépose les marchandises directement au magasin du détaillant.
- Commandes magasin externes
SIM peut créer des commandes directement dans RMS via l'interface graphique de SIM.

- Données d'articles (articles pouvant être vendus ou non)
SIM traite uniquement les articles du niveau de transaction (références) et des niveaux inférieurs (UPC par exemple). Il n'existe donc pas d'interface pour les articles de niveau parent (ou style). Reportez-vous à la documentation de RMS pour obtenir plus d'informations sur sa structure des articles à trois niveaux. Outre les enregistrements des articles approuvés, les données d'articles contiennent des informations d'en-tête, ainsi que des détails article/fournisseur et article/fournisseur/pays. Les données de la hiérarchie des marchandises sont un attribut des données d'articles auxquelles SIM est abonné.
- Données de site (informations mises à jour sur le site du magasin et de l'entrepôt)
- Données article-site
SIM utilise ces données pour les paramètres de commande (par exemple, autoriser l'utilisateur à déterminer si un article est du type commande magasin).
- Données de Diff.
- Données sur le fournisseur et son adresse
- Données de demande de transfert
Les utilisateurs d'entreprise peuvent déplacer le stock d'un magasin à un autre à l'aide de demandes de transfert RMS.
- Demandes de retour
Le système de commercialisation envoie les demandes de retour depuis un magasin vers un entrepôt (RE) et/ou vers un fournisseur (RF). Les marchandises sont expédiées par le magasin lui-même.

De SIM au système de commercialisation

Les données SIM suivantes sont publiées via le RIB pour l'abonnement du système de commercialisation :

- Données de réception
En envoyant les données de réception, SIM communique les articles qu'il a reçus au système de commercialisation. Les types de données de réception sont associés aux éléments suivants :
 - Transferts
 - Commandes (de système de commercialisation) associées aux livraisons directes au magasin ;
 - Nouvelles commandes associées aux livraisons directes au magasin ;
 - Commandes du système de commercialisation (RMS par exemple).
- Données RTV et RTW
SIM communique au système de commercialisation les retours aux fournisseurs et aux entrepôts.
- Données de retour entrepôt
SIM utilise les données d'AE sortantes pour communiquer au système de commercialisation les retours à l'entrepôt.
- Données de commande magasin
SIM envoie ces données pour communiquer une demande d'inventaire pour un article spécifique. Le système de commercialisation peut utiliser ces données pour calculer une quantité de commande recommandée d'un article de type réapprovisionnement « commande magasin ».

- **Données de planning de comptage**
Le système de commercialisation utilise ces données pour la synchronisation des niveaux d'inventaire dans SIM et dans le système de commercialisation. Une fois que le système de commercialisation dispose des données de planning de comptage, SIM et le système de commercialisation répliquent les données de comptage au même moment. Le magasin effectue un comptage physique et télécharge les résultats. Le système de commercialisation compare ensuite les écarts.
- **Données de demande de modification de prix**
Un utilisateur SIM peut demander des modifications de prix, ainsi que des dates d'entrée en vigueur, depuis le système de commercialisation.

De SIM au système de commercialisation via le module de téléchargement de stock du système de commercialisation

- **Résultats de comptage**
Lorsqu'un comptage est autorisé et terminé, SIM crée un fichier plat et le transfère dans un répertoire. En utilisant le fichier plat généré par SIM, le module de téléchargement de stock du système de commercialisation récupère et télécharge les données de comptage physique. Le système de commercialisation utilise ces données pour la synchronisation des niveaux d'inventaire dans SIM et dans le système de commercialisation.

De SIM au système de création de rapports (Crystal Enterprise par exemple)

- **Données pour rapports**
SIM propose la fonction de création de rapports que les détaillants peuvent personnaliser pour refléter les besoins spécifiques de leur activité. Pour faciliter la création de rapports, SIM exploite un outil tiers de création de rapports, Crystal Enterprise.

De SIM à un système de gestion des prix (RPM par exemple)

- **Demande d'approbation des données de modification des prix**
Les données de modification des prix standard, de soldes et de promotion sont envoyées à RPM. RPM regarde s'il y a des conflits et renvoie un statut de validation (que des conflits aient été trouvés ou non). Si la validation est positive, la modification des prix est immédiatement renvoyée à SIM et implémentée.

D'un système de gestion des prix (RPM par exemple) à SIM

- **Données de modification des prix**
RPM envoie les données de modification des prix à SIM. Ce type de données de modification des prix peut provenir d'un niveau entreprise ou d'un niveau magasin.

Chapitre 6 – Conception et présentations des fonctionnalités

Ce chapitre contient des informations sur les différents aspects des fonctionnalités de SIM.



Remarque : pour obtenir des informations sur les raisons fonctionnelles des messages RIB associés aux données de base, reportez-vous à la section « Descriptions fonctionnelles des messages » du chapitre 4, « SIM, le RIB, JMS et RSL ».

Présentation fonctionnelle des ajustements de stock

Pour vous aider à maintenir un stock constant, SIM permet la création d'ajustements de stock pour tous les articles d'un magasin. SIM transmet les modifications au système de commercialisation. La fonctionnalité d'ajustement de stock du système SIM peut être exécutée dans le cadre d'un déploiement sur ordinateur ou sur périphérique de poche sans fil, ou en combinant ces deux méthodes.

Le processus d'ajustement de stock dans SIM possède les caractéristiques suivantes :

- Des codes de motif, correspondant aux statuts, peuvent être attribués aux ajustements de stock, déplaçant le stock vers différentes catégories d'inventaire. Ce code est non seulement utilisé pour créer des rapports mais aussi pour indiquer au système si la quantité doit être augmentée ou diminuée et dans quelle direction. Voici deux exemples :

Exemple 1 :

Un code de motif « Sortie pour réparation » indique au système que le stock doit être diminué dans la catégorie Stock disponible et augmenté dans la catégorie Stock non disponible pour l'article sélectionné.

Exemple 2 :

Un code de motif « Retour de réparation » indique au système de déplacer le stock sélectionné en diminuant la catégorie Stock non disponible et en augmentant la catégorie Stock disponible.

- Plusieurs catégories d'inventaire sont utilisées non seulement pour indiquer à l'utilisateur en magasin les articles disponibles à la vente, mais aussi pour mettre à jour le système de réapprovisionnement central, garantissant ainsi que les commandes appropriées sont passées pour tous les magasins.
- Le niveau d'inventaire de stock disponible d'un article peut être ajusté manuellement.
- Les ajustements peuvent être mis En attente, plaçant ainsi le stock dans une catégorie Non disponible. Ces ajustements peuvent ensuite être transformés en ajustements de stock ordinaires.
- Le système communique tous les ajustements de stock au système de commercialisation.

Récapitulatif des codes de motif et des statuts

Le tableau suivant (présenté à titre d'exemple) fournit une liste de codes de motif de SIM, ainsi que leur description et les statuts liés aux codes de motif. Les détaillants configurent souvent les informations au niveau de la base de donnée afin de mieux satisfaire leurs exigences commerciales. Les abréviations du tableau correspondent aux éléments suivants :

- D : Disponible
- S : Sortant
- ND : Non disponible

Par exemple, le code 83 fait référence à un vol, indiquant que le stock passe de Disponible à Sortant (il a quitté le magasin).

Code	Description du code de motif	Statut
1	Perte	D -> S
81	Dommages - Sortant	D -> S
82	Dommages - Conservé	D -> ND
83	Vol	D -> S
84	Magasin	D -> S
85	Réparation - Sortant	D -> ND
3	Réparation - Entrant	ND -> D
86	Don	D -> S
87	Stock entrant	S -> D
88	Stock sortant	D -> S
89	Elimination de stock conservé	ND -> S
90	Elimination de stock disponible	D -> S
91	Stock - Conservé	D -> ND
92	Admin	D -> S
93	Retour client au magasin	S -> D
95	Hors gestion	D -> S
96	Prêt à la vente	ND -> D
97	Retours	ND -> D
94	Transformation du produit - Entrant	S -> D
98	Transformation du produit - Sortant	D -> S

Présentation fonctionnelle des commandes magasin

Les commandes magasin sont utilisées pour créer, modifier et approuver des commandes auprès d'un fournisseur ou des demandes de transfert vers un entrepôt. Lorsqu'il y a une pénurie d'articles ou une augmentation de la demande d'articles spécifiques, les utilisateurs en magasin doivent pouvoir créer des commandes magasin. L'utilisateur sélectionne un entrepôt ou un fournisseur et ajoute les articles et quantités. Les commandes magasin utilisent RSL pour activer la commande dans RMS.

Le traitement des commandes magasin dans SIM possède les caractéristiques suivantes :

- Créer des commandes auprès d'un fournisseur ou d'un entrepôt.
- Enregistrer la création de la commande sans l'approuver.
- Modifier les articles et commandes de SIM qui ont été créés manuellement ou par réapprovisionnement dans RMS.
- Supprimer des commandes en attente.
- Approuver des commandes magasin.
- Demander des rabais sur facture lors de la modification d'une commande magasin auprès d'un fournisseur.
- Demander les ventes et commandes magasin d'un article lors de la modification d'une commande magasin.

Demandes d'articles

La fonction de demande d'article permet à l'utilisateur de demander l'inventaire d'articles individuels à l'aide des paramètres de réapprovisionnement et d'approvisionnement du système de commercialisation (RMS) directement depuis l'application SIM. Grâce à cette fonction, la responsabilité des magasins est accrue puisque les utilisateurs en magasin sont habilités à gérer la pénurie de stock et la demande croissante depuis SIM.

Cette fonction est différente de celle concernant les commandes magasins (décrite ci-dessus). Pour les demandes d'articles, SIM envoie une demande au système de commercialisation, qui la traite généralement à l'aide de ses paramètres de réapprovisionnement et d'approvisionnement. Avec la fonction de commandes magasin en revanche, l'utilisateur accède directement au système de commercialisation (RMS) et les paramètres de réapprovisionnement et d'approvisionnement ne sont pas utilisés.

Un utilisateur de SIM peut employer la fonction de demande d'article pour demander des articles indépendamment du type de réapprovisionnement utilisé normalement par le système de commercialisation pour l'article concerné.

Tous les articles proviennent soit d'un entrepôt, soit d'un fournisseur, en fonction des paramètres d'approvisionnement de l'article spécifié dans le système de commercialisation. Les articles qui utilisent le réapprovisionnement de type « Commande magasin » (ou les articles qui ne sont pas configurés pour le réapprovisionnement automatique) sont obtenus par la création de commandes ou de transferts en entrepôt uniques, uniquement une fois que le magasin a demandé l'inventaire à l'aide de la fonction Demande d'articles.

Toutes les quantités demandées d'articles dont le type de réapprovisionnement n'est pas « Commande magasin » sont ajoutées au-dessus et au-delà de la quantité normalement obtenue du système de commercialisation, à la prochaine date de revue du réapprovisionnement de l'article. Cependant, si la date de livraison demandée est antérieure à la prochaine date de revue du réapprovisionnement de l'article, la demande est obtenue par la création d'une commande ou d'une demande de livraison entrepôt unique. Tout inventaire demandé est livré au magasin à la date la plus proche possible selon la date de revue du réapprovisionnement, le délai d'exécution du fournisseur ou de l'entrepôt et tout autre facteur susceptible d'influencer le temps mis par une livraison pour arriver au magasin.

Outre la possibilité de créer manuellement des demandes d'articles, SIM permet de programmer ces demandes pour qu'elles soient revues régulièrement sur des écrans de groupes de produits du système frontal. Cette fonction facilite la demande d'articles utilisant le réapprovisionnement de type « commande magasin » en permettant aux utilisateurs d'ajouter à un groupe de produits de type demande d'articles des articles individuels ainsi que des sections entières de la hiérarchie des marchandises. Lorsque la revue du groupe de produits de type demande d'article est programmée, SIM crée automatiquement une demande d'article vierge et y ajoute tous les articles des hiérarchies des marchandises spécifiées dont le type de réapprovisionnement est « commande magasin » ainsi que tout article individuel appartenant au groupe de produits de type demande d'article. L'utilisateur peut alors saisir les quantités requises d'articles et soumettre la demande. Remarque : l'utilisateur peut également ajouter à un groupe de produits de type demande d'articles des articles dont le type de réapprovisionnement n'est pas « commande magasin », mais il pourra uniquement ajouter ces articles un par un.

La liste ci-dessous récapitule les possibilités offertes par la fonction de demande d'articles dans SIM. Cette fonction permet aux utilisateurs de SIM de :

- Créer un groupe de produits de type demande d'articles et programmer sa revue.
- Créer manuellement une demande d'articles non programmée.
- Rechercher et afficher une demande d'articles créée soit manuellement par un utilisateur, soit automatiquement par le gestionnaire de groupes de produits.
- Modifier une demande d'article en attente.
- Supprimer une demande d'article en attente.
- Demander une demande d'article en attente.
- Enregistrer les modifications apportées à une demande d'articles en attente sans l'avoir demandé.
- Imprimer un rapport de demande d'article.
Dans la base de données de SIM se trouve une vue contenant toutes les données destinées à ce rapport, appelée Item_Request_Report_V.

Présentation fonctionnelle des modifications de prix

Les modifications de prix sont utilisées par le détaillant pour changer le prix d'un article spécifique d'un site. Les modifications de prix s'effectuent uniquement sur ordinateur.

Dans le système de commercialisation (RMS par exemple), les utilisateurs créent les prix de vente unitaires d'origine des articles. La structure de la zone de prix précédemment définie dans RMS est désormais définie dans le système de gestion des prix (RPM par exemple). Les articles ne sont plus limités à un groupe de zone de niveau magasin, zone ou entreprise spécifique. Quel que soit le niveau auquel les prix initiaux sont établis, tous les prix de RMS sont définis au niveau article/site. Une fois que les prix initiaux ont été définis, le contrôle des prix est géré dans RPM.

Les utilisateurs qui gèrent les prix dans RPM peuvent utiliser une structure flexible pour contrôler les prix unitaires en effectuant des modifications de type Permanent, Soldes ou Promotion. Cette fonction permet d'utiliser divers groupes de sites pour contrôler les prix unitaires des articles sans être verrouillé dans une structure de zone. Grâce à cette flexibilité, tous les prix sont définis au niveau article/site et peuvent être gérés à tous les niveaux supérieurs.

Dans RMS, un nouvel indicateur au niveau article/site détermine si les utilisateurs SIM peuvent demander les modifications du prix de vente unitaire d'un article dans un site spécifique. Cet indicateur est modifiable et contrôle le comportement en amont (pas en aval).

Si les utilisateurs SIM contrôlent le prix de vente unitaire d'un article dans un site, ils peuvent envoyer à RPM des demandes de modification de prix de type Permanent, Soldes ou Promotion. RPM vérifie l'absence de conflits et fournit une réponse à SIM quant au statut de la demande. Si la demande est acceptée, RPM renvoie également un événement de modification de prix à SIM.

Les utilisateurs de SIM peuvent modifier et créer des prix à condition que les restrictions et suppositions suivantes soient vérifiées :

- L'indicateur du contrôle des prix du magasin article/site doit être réglé sur 'Y' (oui).
- Le prix ne doit pas correspondre à une promotion complexe.
- Toute demande de modification approuvée met à jour le prix existant dans RPM.
- Si un utilisateur demande une nouvelle modification de prix pour le même article, le même site ou la même date au lieu de modifier/corriger le prix existant, la demande est envoyée à RPM en tant que nouvelle demande. La présence de conflits est vérifiée dans RPM. Si des conflits sont détectés au niveau des prix, une réponse de rejet est communiquée à SIM. SIM peut modifier le prix unitaire et les dates effectives d'un prix.
- Si un niveau article/site n'est pas configuré pour le contrôle des prix magasin dans RSM, les utilisateurs SIM peuvent afficher tous les prix envoyés depuis RPM mais ne peuvent pas les contrôler. Ils ne peuvent *pas* non plus créer des demandes de modification de prix à envoyer à RPM.
- La communication entre SIM et RPM est gérée par RSL grâce à une connexion inter-applications en temps réel. L'opération normale de tarification implique que RSL et RPM soient tous deux disponibles. Aucune modification manuelle n'est prise en charge dans SIM.

Présentation fonctionnelle de la réception

Dans SIM, les zones de fonction suivantes sont liées à la réception :

- réception des transferts ;
- livraison entrepôt ;
- livraison directe au magasin.

Réception des transferts

Le processus de réception des transferts se produit dans SIM, le détaillant peut donc recevoir des transferts sur le périphérique de poche et sur l'ordinateur. Le détaillant peut recevoir un transfert entre magasins en scannant un article transféré. Le détaillant ajoute les articles devant être reçus au niveau de l'article ou du colis en scannant ou en entrant manuellement les données. Les articles endommagés peuvent être enregistrés lors du processus de réception. Une fois le transfert reçu, il peut être terminé ou enregistré pour être terminé plus tard. Lorsque la réception du transfert est terminée, SIM diminue l'inventaire En transit et augmente l'inventaire Stock disponible en conséquence.

Le processus de réception des transferts dans SIM présente les caractéristiques suivantes :

- Le stock peut être divisé en différentes catégories selon le statut du stock (par exemple stock Non disponible ou stock En transit).
- Le système peut mettre à jour automatiquement l'inventaire du stock en s'appuyant sur le statut du transfert.
- Le système peut envoyer des emails automatiquement au magasin expéditeur en cas de marchandises endommagées ou d'écarts.
- Le système peut recevoir des colis et des articles non prévus (configuration).
- La fonctionnalité de transfert automatique permet au détaillant de définir dans SIM les magasins desquels il souhaite automatiquement recevoir les transferts. Le transfert automatique ajuste automatiquement le stock disponible du magasin destinataire lorsque le magasin répertorié effectue le transfert. Le magasin destinataire ne peut pas recevoir d'exceptions ou enregistrer les articles endommagés lors du transfert.
- Un enregistrement d'ajustement de stock est rédigé pour les articles endommagés (avec un code de motif Endommagé) afin de diminuer le stock disponible du magasin destinataire.
- Selon les configurations système, les utilisateurs peuvent réouvrir des transferts déjà reçus et modifier les quantités reçues (pendant un nombre de jours défini). Les données corrigées sont alors traitées et renvoyées au système de commercialisation. Cette fonctionnalité d'ajustement de la réception de l'unité est uniquement disponible sur ordinateur.

Transferts excédentaires ou insuffisants

Transferts excédentaires

Lorsque vous recevez plus d'articles que la quantité expédiée, le système retire la différence du stock disponible du magasin expéditeur. Aucun enregistrement d'ajustement de stock n'est envoyé au système de commercialisation ou affiché. Une notification de l'ajustement est envoyée par email au magasin expéditeur. L'email inclut le numéro de transfert, le ou les numéros d'articles et les quantités ajustées.

Transferts insuffisants (manque)

Lorsque vous recevez moins d'articles que la quantité expédiée, le système ajoute la différence au stock disponible du magasin expéditeur. Aucun enregistrement d'ajustement de stock n'est envoyé au système de commercialisation ou affiché. Une notification de l'ajustement est envoyée par email au magasin expéditeur. L'email inclut le numéro de transfert, le ou les numéros d'articles et les quantités ajustées.

Livraison entrepôt

La fonctionnalité de livraison entrepôt dans SIM est utilisée lorsque des marchandises sont envoyées depuis un entrepôt vers un magasin destinataire. La livraison entrepôt dans le système SIM peut être exécutée dans le cadre d'un déploiement sur ordinateur ou sur périphérique de poche sans fil, ou en combinant ces deux méthodes. SIM permet de recevoir des articles depuis n'importe quel nombre d'entrepôts d'entreprises. Les entrepôts doivent avoir été approuvés en tant que zones expéditrices pour le magasin destinataire et les articles expédiés doivent être approuvés pour la livraison au magasin destinataire. Une fois que les articles sont reçus et que la réception est confirmée, SIM met à jour le statut des articles En transit entre le centre de distribution et le magasin avec le statut Disponible. Le stock disponible augmente de la quantité reçue.

Grâce à la fonctionnalité de réception dans SIM, le détaillant peut :

- réceptionner des articles au niveau de l'expédition, des conteneurs ou des colis
- s'assurer que le statut des conteneurs non reçus passe automatiquement à Manquant
- rechercher un conteneur depuis la boîte de dialogue de recherche pour trouver des détails sur la livraison entrepôt
- réceptionner des colis et des articles non prévus (configuration).

Les fonctionnalités supplémentaires suivantes sont disponibles :

- Un enregistrement d'ajustement de stock est rédigé pour les articles endommagés (avec un code de motif Endommagé) afin de diminuer le stock disponible du magasin destinataire.
- Selon les configurations système, les utilisateurs peuvent réouvrir des livraisons entrepôt et modifier les quantités reçues (pendant un nombre de jours défini). Les données corrigées sont alors traitées et renvoyées au système de commercialisation. Cette fonctionnalité d'ajustement de la réception de l'unité est uniquement disponible sur ordinateur.

Livraison directe au magasin.

La livraison directe au magasin se produit lorsque le fournisseur dépose les marchandises directement au magasin du détaillant. Ce processus s'applique couramment aux épiceries et aux commerces de proximité où les fournisseurs restockent régulièrement des marchandises. La facture peut alors être remise ou non au magasin (et non envoyée à la société) et peut être payée avec la caisse ou non. Le système SIM permet au détaillant de créer des enregistrements de livraison. Ce processus permet au détaillant d'entrer ou de scanner le code-barre de n'importe quel article de la livraison. Une fois que le système vérifie le code-barre, le détaillant peut choisir le fournisseur pour la livraison. Le système permet au détaillant de sélectionner une commande existante associée à ce fournisseur pour la réception ou de créer une commande. La fonctionnalité de livraison directe au magasin du système SIM peut être exécutée dans le cadre d'un déploiement sur ordinateur ou sur périphérique de poche sans fil, ou en combinant ces deux méthodes.

Le détaillant peut entrer les informations de facture et réceptionner des articles par colis/pack ou par article. Le détaillant peut aussi imprimer un reçu de livraison une fois que tous les articles ont été reçus et que la livraison est finalisée. Le système peut également gérer des livraisons reçues partiellement, autorisant ainsi plusieurs réceptions pour une seule commande.

Une fois la livraison terminée, le stock disponible du magasin est mis à jour avec les quantités reçues. Un enregistrement d'ajustement de stock est rédigé pour les articles endommagés (avec un code de motif Endommagé) afin de diminuer le stock disponible du magasin destinataire.

Les informations de réception et de commande sont publiées dans le RIB pour le système de commercialisation.

Selon les configurations système, les utilisateurs peuvent réouvrir des livraisons directes et modifier les quantités reçues (pendant un nombre de jours défini). Les données corrigées sont alors traitées et renvoyées au système de commercialisation. Cette fonctionnalité d'ajustement de la réception de l'unité est uniquement disponible sur ordinateur.

Réception par avis d'expédition (AE)

Le processus de réception de SIM permet au détaillant de réceptionner des articles par avis d'expédition (AE) sur périphérique de poche et sur ordinateur. Les avis d'expédition (AE) provenant du fournisseur sont publiés dans le RIB et SIM s'abonne aux données.

Lorsqu'une livraison directe est reçue, SIM recherche un avis d'expédition correspondant à la commande.

Un message demande aux détaillants s'ils souhaitent appliquer l'AE à la livraison. Si l'AE est appliqué, les quantités expédiées depuis l'AE sont appliquées à la quantité reçue pour la livraison directe. Si de nouveaux articles sont inclus dans l'AE mais *ne se trouvent pas* dans la commande d'origine, les articles sont ajoutés à la commande. Une fois l'AE appliqué, le détaillant peut modifier les quantités reçues.

Réception DEX (Direct EXchange) et NEX (Network Exchange)

DEX (Direct EXchange) et NEX (Network Exchange) sont des normes de communication uniformes. DEX permet au fournisseur, à l'aide d'un périphérique de poche, d'échanger des informations de facturation électroniques avec le système de livraison directe au magasin de son client. NEX dispose d'un système de livraison différent, utilisant Internet et non un périphérique de poche.

SIM est conçu pour prendre en charge l'intégration des informations DEX/NEX dans des écrans de livraison directe, simplifiant ainsi le processus de réception. Les données sont transférées au système de livraison directe du magasin à l'aide de la transaction EDI 894 (enregistrement livraison/retour). Une fois les données téléchargées, l'utilisateur en magasin peut afficher, modifier et confirmer les informations contenues dans le fichier avant de réceptionner la livraison directe.

Présentation fonctionnelle de la création de rapports

SIM propose la fonction de création de rapports, que les détaillants peuvent personnaliser pour refléter les besoins spécifiques de leur activité. Pour faciliter la création de rapports, SIM exploite un outil tiers de création de rapports, Crystal Enterprise. SIM dispose d'un mécanisme qui permet de lancer les rapports individuels avec des paramètres et le portail d'outils de rapports, mais la conception de rapports n'est accessible que depuis l'outil de rapports lui-même.

Pour plus d'informations sur la configuration, reportez-vous à la section 'reporting.cfg' du chapitre 2, « Configuration du système d'arrière-plan ».

Rapports opérationnels

Les rapports opérationnels sont générés depuis les zones fonctionnelles de SIM. Ils comprennent des informations sur les listes de prélèvement, les rapports de comptage, la documentation d'expédition, etc. Pour générer ces rapports, SIM utilise un outil de rapports afin que l'utilisateur puisse formater/mettre en page les rapports.

L'outil permet à l'utilisateur final de spécifier avec exactitude les champs de données à afficher dans le rapport (bien que ces données soient limitées aux données SIM disponibles pour le rapport opérationnel en question). Il faut utiliser l'outil de rapports pour modifier le format ou les données du rapport.

SIM permet à l'utilisateur d'identifier un modèle de rapport par défaut qu'il pourra utiliser pour chacun des rapports opérationnels. Lorsque l'utilisateur veut créer un rapport opérationnel dans SIM, l'application recherche le modèle de rapport qui correspond au modèle par défaut spécifié pour ce rapport.

Rapports analytiques (et ad hoc)

Les rapports analytiques exploitent les données de SIM pour effectuer une analyse historique. Les détaillants peuvent utiliser ces rapports pour prendre des décisions sur les processus commerciaux clés d'un magasin (livraisons des jours précédents, nombre d'articles réapprovisionnés un jour donné, etc).

L'outil de rapports permet aux détaillants de formater et mettre en page leurs rapports et aux utilisateurs de spécifier avec exactitude les champs de données à afficher. Tous les chiffres et paramètres sont définis à l'aide de l'outil de rapports (bien que ces données soient limitées aux données SIM disponibles pour le rapport en question).

Suppositions

- SIM ne fait référence à aucun autre système de sécurité externe pour appliquer les droits de l'outil de rapports. Si un utilisateur peut créer un rapport ou lancer l'outil de rapports dans SIM, on suppose que cet utilisateur possède également le niveau d'accès nécessaire à l'outil de rapports.
- SIM ne propose aucune fonction de planning pour les rapports analytiques (et ad hoc). Tout planning doit se faire à partir de l'outil de rapports lui-même.

Présentation fonctionnelle des retours et des demandes de retour

Retours

SIM permet à l'utilisateur en magasin de rechercher, de créer, de modifier, de supprimer et de terminer les retours depuis le magasin vers l'entrepôt ou directement au fournisseur. La fonctionnalité de retour du système SIM peut être exécutée dans le cadre d'un déploiement sur ordinateur ou sur périphérique de poche sans fil, ou en combinant ces deux méthodes.

Les quantités d'articles peuvent être saisies par unités ou par colis. Une fois les quantités applicables entrées, l'utilisateur est invité à entrer un code de motif pour le retour. Les codes de motif sont définis par le détaillant. Le stock disponible est diminué de la quantité du retour sauf si un inventaire non disponible existe pour l'article. Dans ce cas, l'utilisateur peut utiliser cet inventaire pour le retour. Une fois le code de motif sélectionné, l'utilisateur peut soit terminer le retour soit l'enregistrer pour le terminer plus tard.

Une fois le retour terminé, un document de retour peut être imprimé à titre de rapport et/ou de bordereau de marchandises pour l'expédition.

Demandes de retour

La fonctionnalité de demande de retour renvoie des demandes devant être exécutées depuis un magasin vers un entrepôt et/ou vers un fournisseur et ayant été générées via le système de commercialisation.

Une demande de retour peut être générée suite à un problème de sécurité (par exemple, des morceaux de verre sont découverts dans un produit). Cette fonctionnalité se résume à un retour généré par le système de commercialisation et pouvant être modifié et approuvé dans SIM. La fonctionnalité de demande de retour du système SIM ne peut être exécutée que dans le cadre d'un déploiement sur ordinateur.

Une fois que SIM reçoit les données de demande de retour, il prend le contrôle de la demande et autorise les utilisateurs en magasin à ajouter, modifier, supprimer, enregistrer et expédier la demande de retour. Une fois la demande supprimée ou expédiée, un message est renvoyé au système de commercialisation.

Présentation fonctionnelle de l'ordonnancement

La fonctionnalité d'ordonnancement permet aux utilisateurs de connaître approximativement l'emplacement d'un article dans un magasin. L'ordonnancement d'un magasin améliore son fonctionnement et réduit la durée de recherche d'articles pour les employés (lors d'un comptage par exemple). Le détaillant peut ordonnancer tous les articles du magasin et créer des zones uniques pour les articles. Le système peut inviter les utilisateurs à rechercher un article dans une zone spécifique. La fonctionnalité d'ordonnancement du système SIM peut être exécutée dans le cadre d'un déploiement sur ordinateur ou sur périphérique de poche sans fil, ou en combinant ces deux méthodes.

La fonctionnalité d'ordonnancement comprend trois méthodes d'attribution des articles à des emplacements d'un magasin : macro ordonnancement, numéros de meuble et micro ordonnancement. Lors de l'ordonnancement, le système suit ces trois modèles.

Le macro ordonnancement représente le plus haut niveau de zones définies dans le magasin. L'utilisateur peut créer des zones macro, attribuer, supprimer ou déplacer des articles de ces zones et réordonnancer une zone macro entière (application sans fil uniquement).

Un utilisateur peut attribuer plusieurs numéros de meuble à un article dans une zone. Les numéros de meuble permettent aux utilisateurs d'associer un numéro, une étiquette ou une lettre spécifique à un article ordonnancé.

Le micro ordonnancement constitue le niveau de zone le plus bas (le plus granulaire) d'un article. Par exemple, un présentoir avec un numéro de meuble unique peut inclure cinq articles micro ordonnancés.

Le tableau suivant présente un exemple d'ordonnancement :

Macro ordonnancement	N° de meuble	Micro ordonnancement
Produits	1	Oranges
	2	Pommes
	3	Bananes
	4	Oranges
Produits surgelés	A	Plats préparés
	A	Burritos
	B	Burritos
Céréales	Rayon 1	Flocons d'avoine
	Rayon 2	Flocons d'avoine

Présentation fonctionnelle du réapprovisionnement des rayons

Le processus de réapprovisionnement veille à ce que l'inventaire de la surface de vente soit défini de façon à satisfaire le mieux possible les clients.

La fonctionnalité de réapprovisionnement de SIM est associée au déplacement des marchandises depuis la réserve vers la surface de vente. Par exemple, lorsqu'un utilisateur s'aperçoit que la quantité d'une boisson gazeuse est peu élevée, il peut démarrer un processus de réapprovisionnement pour en déplacer depuis la réserve vers la surface de vente.

Le processus de réapprovisionnement des rayons de SIM possède les caractéristiques suivantes :

- Le système calcule le nombre d'articles à placer en surface de vente pour pouvoir répondre en permanence aux attentes des clients. Les employés du magasin doivent réapprovisionner les articles les plus urgents en premier.
- Le système affiche une zone de prélèvement du stock et fournit un numéro de conteneur pour faciliter la recherche de stock pour les listes de prélèvement.
- Le système permet de créer des demandes de prélèvement à la demande.
- Le système dirige l'utilisateur dans la réserve par micro ordonnancement, afin que les articles puissent être prélevés de la manière la plus efficace.

Le réapprovisionnement requiert que le stock disponible soit réparti en trois catégories : surface de vente, réserve et zone de livraison. En raison de ces catégories, le réapprovisionnement des rayons concerne presque toutes les zones de l'application. Par exemple, les ajustements de stock et les transferts sont concernés car le système doit prendre en compte les catégories d'inventaire lors de l'exécution de l'un de ces processus. L'inventaire Disponible du système correspond à la somme des trois catégories.

Lorsque des marchandises deviennent disponibles (qu'elles entrent en magasin via un transfert, une livraison directe, etc.), elles sont toujours placées dans la catégorie Réserve.

Lorsque des marchandises deviennent non disponibles (qu'elles quittent le magasin via un transfert, sont endommagées, etc.), le système traite l'ajustement de stock selon l'ordre suivant :

- 1 Réserve
- 2 Surface de vente
- 3 Zone de livraison

Récapitulatif du calcul de réapprovisionnement

Lorsque le calcul commence, le système démarre la procédure suivante :

- Le système collecte tous les articles ordonnancés dans la surface de vente disposant d'une capacité dans le groupe de produits.
- Si une liste de prélèvement précédente en cours existe pour le groupe sélectionné, le système considère que le traitement de tous les articles de la liste de prélèvement est terminé. Si une liste de prélèvement précédente existe mais *n'a pas* démarré, le système supprime l'ancienne liste et en crée une autre.
- Le système vérifie et obtient les paramètres de configuration définis via l'interface utilisateur (unité de mesure du groupe, pourcentage de remplissage, etc.).

- Le système obtient les quantités en rayon et le stock disponible pour les articles trouvés.
- Il convertit les quantités en unités de mesure par défaut du groupe.
- Le système compare ensuite la quantité en rayon à la capacité de surface de vente cumulée afin de déterminer le pourcentage de rupture de stock de chaque article.
- Une fois le pourcentage de « rupture de stock » calculé pour chaque article, le système classe les articles du pourcentage de rupture de stock le plus haut au plus bas. Si certains articles possèdent le même pourcentage de rupture de stock, le système utilise l'article dont la quantité en rayon est la plus basse. Par exemple, si 10 articles A sont disponibles en rayon, sur un maximum de 100, et que 1 article B est disponible en rayon, sur un maximum de 10, ils possèdent le même pourcentage de rupture de stock. Cependant, le système considère que B est prioritaire car sa quantité en rayon est la plus basse.
- Pour chaque article, le système calcule la quantité de prélèvement à transférer depuis la réserve ou la zone de livraison vers la surface de vente. En conservant les articles par ordre de priorité, le système recherche le stock disponible, les articles de la réserve et de la zone de livraison et le pourcentage nécessaire de remplissage de la surface de vente. Le système « prend » d'abord l'inventaire de la réserve, puis celui de la zone de livraison. La quantité en surface de vente doit être égale ou inférieure à la capacité.
- Si le type de liste de prélèvement est En cours de journée, le système s'arrête lorsque la quantité à prélever est égale à la quantité de prélèvement cumulée.
- S'il s'agit d'un type de prélèvement en Fin de journée, le système continue jusqu'à la fin du prélèvement.
- Si la quantité de prélèvement générée par le système présente un nombre après la virgule, le système arrondi par défaut au nombre entier le plus proche.
- Le système génère et affiche la liste par ordonnancement à l'utilisateur. Si les articles ne sont pas ordonnancés dans la réserve, le système les affiche par ordre de numéro d'article.

Présentation fonctionnelle des comptages

SIM permet de planifier, d'effectuer et d'autoriser les comptages. SIM comprend les quatre types de comptages suivants, décrits dans cette section :

- Ad hoc (application sans fil uniquement) ;
- Unité ;
- Unité et valeur ;
- Unité et valeur tous les rayons ;
- Ligne problématique.

Le processus de comptage est identique pour les types de comptage Unité, Unité et valeur et Ligne problématique. Les différences entre ces trois types de comptages reposent sur la configuration ou sur les articles comptés.

Certaines fonctions de comptage, telles que la configuration de groupes de produits, des plannings et des autorisations, sont disponibles sur ordinateur uniquement. Le comptage réel de l'inventaire peut être effectué sur PC et sur périphérique sans fil. Les comptages sans fil sont divisés de façon à ce que chaque utilisateur d'un périphérique de poche effectue un comptage en fonction d'une zone macro (par exemple, Légumes ou Céréales).

Comptages ad hoc

Pour effectuer un comptage ad hoc, l'utilisateur se déplace dans le magasin et scanne tous les articles devant être comptés. Les comptages ad hoc ne sont possibles qu'avec le système sans fil. Ils ne sont associés à aucune fonctionnalité de groupe ou de planning.

Le processus de comptage ad hoc de SIM possède les caractéristiques suivantes :

- Le système réplique les données de l'inventaire au moment de l'identification et de l'ajout d'un article au comptage.
- Vous pouvez enregistrer un comptage ad hoc pour le terminer ultérieurement ; vous pouvez également récupérer et reprendre des comptages existants.
- Le système utilise des seuils d'écart (définis lors de la configuration administrative par le détaillant) basés sur un pourcentage ou une unité de mesure standard selon la famille de l'article dans la hiérarchie marchandises.
- Sur ordinateur, le système permet l'autorisation d'articles présentant des écarts (selon les seuils de pourcentage et d'unité de mesure standard).
- Lorsque les quantités d'articles autorisées entrées sont différentes du stock disponible, le système crée des ajustements de stock qui sont envoyés au système de commercialisation. Reportez-vous à la section « Ajustements de stock » de ce chapitre.

Comptages de type Unité

Le processus de comptage Unité de SIM présente les caractéristiques suivantes :

- Les articles individuels et les hiérarchies d'articles sont rassemblées dans un groupe de produits d'unité unique pour la création d'un planning de comptage.
- Les groupes de produits d'une unité peuvent être planifiés pour un comptage pour un jour spécifique ou à une fréquence planifiée (quotidienne, hebdomadaire, mensuelle ou annuelle par exemple).
- Vous pouvez affecter plusieurs magasins au comptage planifié.
- Une liste d'articles de comptage est générée au niveau du magasin via un processus de traitement par lots (décrit ci-dessous) exécuté quotidiennement.
- Les utilisateurs disposant d'un niveau de sécurité suffisant sont notifiés en cas d'écarts dépassant les seuils de tolérance et le système peut forcer un nouveau comptage si les écarts sont trop élevés. Le système utilise des seuils d'écart (définis lors de la configuration administrative par le détaillant) basés sur un pourcentage ou une unité de mesure standard selon la famille de l'article dans la hiérarchie des marchandises.
- Sur ordinateur, le système permet l'autorisation d'articles présentant des écarts (selon les seuils de pourcentage et d'unité de mesure standard).
- Lorsque les quantités d'articles autorisées entrées sont différentes du stock disponible, le système crée des ajustements de stock qui sont envoyés au système de commercialisation. Reportez-vous à la section « Ajustements de stock » de ce chapitre.

Comptages de type Unité et valeur

Comptages tiers de type Unité et valeur

Les détaillants peuvent effectuer leurs comptages de type Unité et valeur à l'aide d'un fournisseur tiers, RGIS par exemple. S'ils choisissent cette solution, les comptages de type Unité et valeur sont planifiés dans SIM mais le véritable processus de comptage s'effectue dans le système RGIS. Une fois le comptage du stock physique terminé, le système RGIS exporte les résultats dans SIM.

SIM compare les informations du comptage envoyées par RGIS avec les valeurs de stock disponible dont il dispose. Les utilisateurs peuvent afficher dans SIM tous les articles du comptage qui diffèrent des valeurs du stock disponible. Les limites de variance pré-définies (différence d'unités et de pourcentage) sont utilisées pour déterminer quels articles n'appartiennent pas au niveau « acceptable » et doivent par conséquent être considérés comme différents.

Une fois que les articles présentant des écarts ont été revus et approuvés, SIM exporte les informations du comptage dans RMS. RMS met à jour ses positions de stock disponible de manière à refléter les données de SIM et met en rapport les ajustements financiers applicables et le comptage Unité et valeur.

Comptages de type Unité et valeur dans SIM

L'utilisateur peut créer l'un des types de comptage Unité et valeur suivants :

- Comptage Unité et valeur standard
Les hiérarchies d'articles sont rassemblées dans un groupe de produits d'unité et valeur unique pour la création d'un planning de comptage.
- Comptage Unité et valeur Tous les sites
Si le magasin concerné utilise l'ordonnancement, l'utilisateur peut associer tous les sites du macro ordonnancement (ce qui inclut tous les articles du magasin) à un comptage.

Le processus de comptage Unité et valeur de SIM présente les caractéristiques suivantes :

- Les utilisateurs disposant d'un niveau de sécurité suffisant sont notifiés en cas d'écarts dépassant les seuils de tolérance et le système peut forcer un nouveau comptage si les écarts sont trop élevés. Le système utilise des seuils d'écart (définis lors de la configuration administrative par le détaillant) basés sur un pourcentage ou une unité de mesure standard selon la famille de l'article dans la hiérarchie des marchandises.
- Sur ordinateur, le système *requiert* l'autorisation d'articles présentant des écarts (selon les seuils de pourcentage et d'unité de mesure standard).

Comptages Unité et valeur Tous les rayons

- Les groupes de produits d'unité et valeur uniques peuvent être planifiés pour un comptage un jour donné.
- Le planning de comptage pour les comptages de type Unité et valeur est envoyé au système de commercialisation à chaque création, mise à jour et suppression.
- Vous pouvez affecter plusieurs magasins au comptage planifié.
- Une liste d'articles de comptage est générée au niveau du magasin via un processus de traitement par lots (décrit ci-dessous) exécuté quotidiennement.
- Lorsque les quantités d'articles autorisées entrées par l'utilisateur sont différentes du stock disponible, un fichier plat est envoyé au système de commercialisation avec un en-tête contenant le numéro de comptage, la date du comptage et le numéro du magasin qui a exécuté le comptage. Le fichier contient les détails de chaque article et la quantité. Ces informations sont alors transférées au système de commercialisation et peuvent être utilisées pour créer des rapports.
- Les valeurs réelles de quantité comptées pour les marchandises et les numéros de réservation peuvent faire l'objet d'une consolidation à la fin de l'année.

Comptages de type Ligne problématique

Cette fonctionnalité permet aux magasins de créer des comptages automatiques selon des critères prédéfinis ; par exemple, le détaillant peut décider de compter tous les articles ayant des valeurs de stock disponible négatives.

Une fois que les magasins ont établi les critères (en fonction des zones problématiques), un traitement par lots est exécuté pour trouver tous les articles répondant à ces critères. Les articles trouvés sont ajoutés à un comptage généré par le système. Le comptage s'effectue de la même façon qu'un comptage planifié. Les comptages de type Ligne problématique ne sont associés à aucune fonctionnalité de planning.

Le processus de comptage de type Ligne problématique de SIM présente les caractéristiques suivantes :

- Les articles individuels et les hiérarchies d'articles sont rassemblés dans un groupe de produits de ligne problématique unique.
- Une liste d'articles de comptage de ligne problématique est générée au niveau du magasin via un processus de traitement par lots de ligne problématique reposant sur les paramètres correspondants. Le traitement par lots est décrit ci-dessous.
- Les utilisateurs disposant d'un niveau de sécurité suffisant sont notifiés en cas d'écarts dépassant les seuils de tolérance et le système peut forcer un nouveau comptage si les écarts sont trop élevés. Le système utilise des seuils d'écart (définis lors de la configuration administrative par le détaillant) basés sur un pourcentage ou une unité de mesure standard selon la famille de l'article dans la hiérarchie des marchandises.
- Sur l'ordinateur, le système permet l'autorisation d'articles présentant des écarts (selon les seuils de pourcentage et d'unité de mesure standard).
- Lorsque les quantités d'articles autorisées entrées sont différentes du stock disponible, le système crée des ajustements de stock qui sont envoyés au système de commercialisation. Reportez-vous à la section « Ajustements de stock » de ce chapitre.

Présentation fonctionnelle de l'étiquetage

L'étiquetage est une fonctionnalité interne de SIM permettant d'imprimer des étiquettes de facing et des étiquettes d'articles.

Des formats et des quantités d'étiquettes sont définis pour les articles de l'ordonnement. L'intégralité des zones macro (par exemple, produits) peut être imprimée depuis les écrans d'ordonnement. Pour en savoir plus sur l'ordonnement, reportez-vous à la section « Présentation fonctionnelle de l'ordonnement » de ce chapitre. La boîte de dialogue Etiquettes d'articles permet aux utilisateurs de créer des étiquettes pour un article spécifique sur un ordinateur de bureau et sur un périphérique sans fil.

Les étiquettes peuvent être créées à partir des modifications de prix et des commandes reçues.

Présentation fonctionnelle du transfert sortant

Dans SIM, les zones de fonction suivantes, liées aux transferts, sont traitées dans cette section :

- création de transferts entre magasins ;
- création de demandes de transfert entre magasins ;
- alertes email.

Création de transferts entre magasins

SIM permet la recherche, la création, la modification et la suppression de transferts entre magasins. Un transfert entre magasins correspond au déplacement de stocks entre deux magasins au sein de la même entreprise.

Cette fonctionnalité peut être effectuée dans le cadre d'un déploiement sur ordinateur ou sur périphérique sans fil, ou en combinant ces deux méthodes. Les utilisateurs peuvent créer un transfert en sélectionnant le magasin destinataire et en ajoutant des articles qu'ils scannent et/ou entrent manuellement. Le système vérifie que le magasin destinataire dispose de l'approbation pour recevoir les articles sélectionnés et que le magasin expéditeur détient le stock disponible. Un transfert peut être immédiatement envoyé ou enregistré pour être expédié plus tard. Au moment où le transfert est expédié, SIM diminue l'inventaire disponible du magasin expéditeur et augmente l'inventaire en transit du magasin destinataire.



Remarque : pour plus d'informations sur la réception de transferts, reportez-vous à la section « Présentation fonctionnelle de la réception » de ce chapitre.

La fonctionnalité de transfert dans SIM présente les caractéristiques suivantes :

- Le stock est divisé en différentes catégories en fonction de son statut (par exemple stock en transit ou stock réservé au transfert).
- Le système met à jour automatiquement l'inventaire du stock en fonction du statut du transfert.
- La fonctionnalité de magasin partenaire permet de configurer dans SIM des groupes de magasins au sein d'une zone de transfert vers lesquels le détaillant transfère régulièrement des articles. La liste de valeurs proposées aux utilisateurs lors de la création d'un transfert est ainsi réduite. Le détaillant peut toujours créer un transfert vers n'importe quel magasin n'appartenant pas au groupe de magasins partenaires, à condition que celui-ci se trouve dans la zone de transfert.

Présentation du déplacement des stocks après une expédition réussie

- 1 Le stock est déplacé de la catégorie Réservé au transfert et de la catégorie Prévu pour le transfert.
- 2 La quantité réservée pour le transfert ainsi que le stock disponible diminuent pour la zone expéditrice.
- 3 Le transfert prévu pour le magasin destinataire se traduit par un déplacement de stock vers la catégorie En transit. La quantité de transfert est déplacée de la catégorie En transit vers la catégorie Stock disponible lorsque le magasin destinataire reçoit le transfert.

Demandes de transfert

Les demandes de transfert permettent aux magasins de demander des produits auprès d'autres magasins ou d'autoriser les utilisateurs de l'entreprise à déplacer le stock entre des magasins via les demandes de transfert du système de commercialisation. Le processus suivant se produit :

- 1 La demande de transfert est lancée par le magasin demandeur (ou en son nom).
- 2 Le magasin qui reçoit la demande accepte ou refuse la demande de transfert.
- 3 Si elle est acceptée, le transfert est créé et les marchandises sont transférées du magasin ayant reçu la demande de transfert au magasin demandeur.

Les détaillants sont autorisés à accepter ou à refuser une demande de transfert en attente de réponse uniquement sur ordinateur. La demande de transfert réelle peut être créée sur l'ordinateur ou sur le périphérique sans fil.

Situations fonctionnelles de processus RIB relatifs aux transferts

Transfert entre magasins initié dans SIM par le magasin expéditeur

Les explications numérotées ci-après se rapportent à un transfert entre magasins initié dans SIM par le magasin expéditeur, les deux magasins se trouvant dans SIM.

- 1 Le processus a lieu dans SIM.
- 2 SIM publie un ou plusieurs messages d'AE sortant après l'expédition du transfert pour en aviser le système de commercialisation.
- 3 SIM publie un ou plusieurs messages de réception lors de la réception du transfert pour en aviser le système de commercialisation.

Transfert entre magasins initié dans SIM par le magasin destinataire

Les explications des éléments numérotés ci-après se rapportent à un transfert entre magasins initié dans SIM par le magasin destinataire, les deux magasins se trouvant dans SIM.

- 1 Le processus a lieu dans SIM.
- 2 SIM publie un ou plusieurs messages d'AE sortant après l'expédition du transfert pour en aviser le système de commercialisation.
- 3 SIM publie un ou plusieurs messages de réception lors de la réception du transfert pour en aviser le système de commercialisation.

Transfert entre magasins initié au niveau de l'entreprise dans le système de commercialisation

Les explications numérotées ci-après se rapportent à un transfert entre magasins initié au niveau de l'entreprise dans le système de commercialisation.

- 1 Après la création d'un transfert, le système de commercialisation publie un ou plusieurs messages d'ordre de réapprovisionnement.
- 2 Le processus a lieu dans SIM. SIM étant abonné à la demande de transfert initialement publiée depuis le système de commercialisation, SIM publie une réponse au système de commercialisation via un message de statut de l'ordre de réapprovisionnement, que l'utilisateur accepte ou refuse la demande de transfert.
- 3 Les messages suivants sont associés à l'expédition du transfert :
 - a SIM publie un ou plusieurs messages d'AE sortant après l'expédition du transfert pour en aviser le système de commercialisation.
 - b SIM s'abonne à un ou plusieurs messages d'AE entrant pour les données de notification d'expédition. Pour limiter la quantité de code devant être maintenue dans SIM, le RIB convertit les messages d'AE sortant au format de message d'AE entrant pour permettre à SIM de conserver un abonné unique quelle que soit la source de l'AE. Autrement dit, les livraisons entrepôt et les livraisons directes sont communiquées à l'aide du même format de message (AE entrant).
 - c Le ou les messages de réception sont envoyés lors de la réception du transfert pour en aviser le système de commercialisation.

Présentation fonctionnelle des pertes

Les pertes correspondent au stock perdu (bananes devenues noires, par exemple).

Pour que les valeurs d'inventaire soient plus précises, la fonctionnalité de pertes de SIM permet aux utilisateurs en magasin de créer des groupes de produits de pertes. Les valeurs de pourcentage ou d'UDM standard de variance peuvent être définies dans le groupe de produits de pertes. Les articles individuels et les hiérarchies d'articles peuvent être rassemblés dans un groupe de produits.

Un utilisateur peut planifier la date d'exécution d'un processus de traitement par lots de groupe de produits de pertes. Les ajustements de stock sont effectués automatiquement en fonction de la configuration des variances du groupe de produits. Les ajustements de stock sont envoyés via le RIB au système de commercialisation.

Chapitre 7 – Traitements par lots JAVA

Ce chapitre comporte les éléments suivants :

- Une présentation du traitement par lots de SIM
- Une description du mode d'exécution des traitements par lots et des paramètres clés
- Un récapitulatif fonctionnel de chaque traitement par lots et de ses dépendances
- Une description de certaines caractéristiques des traitements par lots (valeurs renvoyées, redémarrage, reprise, etc.).

Présentation du traitement par lots

Les traitements par lots de SIM sont exécutés dans Java. La plupart de ces traitements disposent de leur propre traitement principal. Cependant, certains appels des traitements par lots utilisent du code provenant des services exécutés sur le serveur. En général, ce traitement se produit lorsque les traitements par lots lancent des actions ne faisant pas partie de leur traitement principal (par exemple, lorsqu'ils utilisent une méthode d'assistant ou qu'ils accèdent à la base de données).

Les traitements par lots de SIM présentent les caractéristiques suivantes :

- Ils ne sont pas accessibles via une interface utilisateur graphique.
- Ils sont planifiés par le détaillant.
- Ils sont conçus pour traiter d'importants volumes de données

Exécution d'un traitement par lots Java

Sous Unix, les processus Java sont planifiés via des scripts shell exécutables (fichiers .sh). Sous Windows, les processus Java sont planifiés via des fichiers de traitement par lots exécutables (fichiers .bat).

Les scripts shell (fichiers .sh) et les fichiers de traitement par lots (fichiers .bat) sont fournis par Retek. Ils effectuent les opérations suivantes en interne :

- configuration de l'environnement d'exécution Java (JRE) avant l'exécution du processus Java ;
- déclenchement du processus de traitement par lots Java.

Pour les processus de traitement par lots `DexnexFileParser`, `PosmodDnldFileParser` et `ResaFileParser`, un paramètre entré dans la ligne de commande après le nom du script shell ou du fichier de traitement par lots fournit au traitement par lots le nom et l'emplacement du fichier à traiter.

Récapitulatif des fichiers exécutables associés aux paquetages et aux classes Java

Le tableau suivant donne une description des scripts shell exécutables, des fichiers de traitement par lots et des paquetages Java ainsi que des classes principales qu'ils contiennent définissant la classe Java (de traitement par lots) exécutée.

Script shell exécutable	Fichier de traitement par lots exécutable pour Windows	Paquetage Java	Classe
ActivatePriceChanges.sh	ActivatePriceChanges.bat	com.iso.cs.batch	UpdatePriceChanges
CleanupPickList.sh	CleanupPickList.bat	com.iso.cs.batch	CleanupPickList
DexnexParser.sh	DexnexParser.bat	com.iso.cs.dataaccess.ms.rms	DexnexFileParser
FrontEnd.sh	FrontEnd.bat	com.iso.cs.batch.frontend	FrontEnd
PurgeData.sh	PurgeData.bat	com.iso.cs.batch	PurgeData
ResaFileParser.sh	ResaFileParser.bat	com.iso.cs.dataaccess.resa	ResaFileParser
ReturnNotAfterDateAlert.sh	ReturnNotAfterDate	com.iso.cs.batch	ReturnNotAfterDateAlert
rssAlertScan.sh	rssAlertScan.bat	com.iso.cs.util	AlertDispatcher
SameDayPriceChangeActivator.sh	SameDayPriceChangeActivator.bat	com.retek.iso.cs.batch	SameDayPriceChangeActivator
ThirdPartyStockCountParser.sh	ThirdPartyStockCountParser.bat	com.iso.cs.dataaccess.ms.rms	ThirdPartyStockCountParser

Gestionnaire et ligne de commande

Si le détaillant utilise un gestionnaire, les arguments sont placés dans le gestionnaire.

Dans le cas contraire, les arguments doivent être transmis par la ligne de commande.

Normes de traitement par lots de valeur renvoyée

Les explications suivantes décrivent les valeurs renvoyées par la fonction et le programme que les processus de traitement par lots de SIM utilisent :

- 0 : La fonction s'est exécutée sans erreur et le traitement doit se poursuivre normalement.
- 1 : Une erreur sans gravité s'est produite (la validation d'un enregistrement entrant a échoué par exemple) et la fonction d'appel va faire passer cette erreur à un autre niveau ou gérer l'exception.

Descriptions fonctionnelles et dépendances

Le tableau suivant offre un récapitulatif des processus de traitement par lots de SIM. Il contient une description de chaque fonctionnalité commerciale et des dépendances du traitement par lots.

Traitement par lots	Détails	Dépendances de traitement par lots
<p>UpdatePriceChanges (ActivatePriceChanges.sh)</p>  <p>Remarque : ce traitement par lots inclut un paramètre de ligne de commande <i>facultatif</i>. Le traitement est décrit dans la colonne Détails de ce tableau.</p>	<ul style="list-style-type: none"> • Ce traitement par lots attribue le statut Actif aux modifications de prix En attente ou de la liste des étiquettes d'articles de RPM (type = modification de prix) qui entrent en vigueur le lendemain. <p>Si le statut d'une modification de prix passe de En attente à Actif, le traitement par lots détermine si l'article dispose d'une modification de prix active existante. Si c'est le cas, il fait passer ce statut à Terminé. Ce programme de traitement par lots remplace également le prix unitaire actuel par le prix unitaire en attente modifié.</p> <ul style="list-style-type: none"> • Paramètre de date facultatif avec UpdatePriceChanges (ActivatePriceChanges.sh) – 'mm/dd/yyyy' <p>Pour permettre au détaillant de "rattraper" le traitement par lots, UpdatePriceChanges (ActivatePriceChanges.sh) peut être exécuté avec un paramètre <i>facultatif</i> contenant la date (mm/dd/yyyy) à comparer avec la date d'entrée en vigueur de la modification de prix. Si la date n'est pas spécifiée, la date du serveur sert de date par défaut.</p>	<p>Aucune dépendance</p>

Traitement par lots	Détails	Dépendances de traitement par lots
CleanupPickList	<p>Le processus de traitement par lots de fin de journée est exécuté chaque soir pour redéfinir la zone de livraison et fermer toute liste de prélèvement ouverte. Le système déplace l'intégralité du stock de la zone de livraison à la réserve. Toute liste de prélèvement en attente ou en cours se voit attribuer le statut Annulé. Les utilisateurs activant une liste de prélèvement sont déconnectés de force du système. Le système prend le contrôle du verrouillage de leur base de données afin qu'ils ne puissent effectuer aucun enregistrement. Après l'exécution du traitement par lots, toutes les listes de prélèvement sont terminées ou annulées et la zone de livraison ne compte aucun stock.</p>	Aucune dépendance
DexnexFileParser	<p>Ce processus importe les données d'une expédition de livraison directe (article, fournisseur, zone destinataire, etc.) depuis un fichier du répertoire DEX/NEX (dexnex). Le traitement par lots ouvre le fichier à lire, l'analyse et insère la commande, l'expédition et les données de réception dans la base de données.</p> <p>Une fois les données téléchargées, le processus SIM crée une livraison directe DEX/NEX, ce qui permet à l'utilisateur en magasin d'afficher, de modifier et de confirmer les informations contenues dans le fichier DEX/NEX avant de l'approuver ; la livraison peut donc devenir une livraison directe En cours.</p> <p>Reportez-vous à l'« Annexe B » plus loin dans ce document pour connaître les spécifications de mise en page du fichier.</p>	Aucune dépendance

Traitement par lots	Détails	Dépendances de traitement par lots
<p>Frontend-‘O’</p>  <p>Remarque : le paramètre de ligne de commande qui suit le nom du processus de traitement par lots détermine le processus exécuté.</p> <p>O = Commandes magasin</p>	<p>Commandes magasin</p> <p>Le processus de traitement par lots recherche les groupes de produits définis comme « groupe de produits de type commande magasin » planifiés pour le jour en cours. Le processus crée la commande magasin (avec des articles et des quantités) et lui attribue le statut En attente ou Brouillon. L'utilisateur (un responsable par exemple) peut alors ajouter ou supprimer des articles, modifier des quantités, etc. Il soumet ensuite les données au système de commercialisation. Le système de commercialisation peut générer des commandes ou des transferts d'entrepôt à magasin, selon le cas.</p>	<p>Aucune dépendance</p>
<p>Frontend-‘P’</p>  <p>Remarque : le paramètre de ligne de commande qui suit le nom du processus de traitement par lots détermine le processus exécuté.</p> <p>P = ligne problématique</p>	<p>Ligne problématique</p> <p>Avant l'exécution du traitement par lots, le détaillant établit un groupe d'articles et des hiérarchies d'articles (en les associant au type de groupe Ligne problématique) et sélectionne les paramètres applicables (stock disponible négatif, etc.). Le processus de traitement par lots de ligne problématique parcourt la liste d'articles du groupe et détermine lesquels correspondent aux paramètres. Le système crée automatiquement un comptage à partir de ces derniers.</p> <p>Si un article est un article de ligne problématique (stock négatif par exemple) lors d'un comptage et que l'utilisateur ne peut pas effectuer le comptage ce même jour, l'article ne constituera probablement plus une ligne problématique le jour suivant (stock positif). Cependant, le système continue à créer un comptage pour cet article car un problème s'est produit à un moment donné.</p>	<p>Aucune dépendance</p>

Traitement par lots	Détails	Dépendances de traitement par lots
<p>Frontend- 'S'</p>  <p>Remarque : le paramètre de ligne de commande qui suit le nom du processus de traitement par lots détermine le processus exécuté.</p> <p>S = Unité ou Unité et valeur</p>	<p>Unité ou Unité et valeur</p> <p>Chaque jour, un processus de traitement par lots crée les comptages planifiés pour la journée. Le système examine tous les enregistrements de comptage planifiés et détermine si certains sont planifiés pour le jour même. Le processus crée les comptages pour chaque magasin. Si un comptage planifié comprend une liste de 5 magasins, 5 enregistrements de comptage distincts sont créés.</p> <p>Pour les comptages de type Unité et valeur uniquement, si un comptage est exécuté pour tous les sites, le traitement par lot crée des comptes individuels pour chaque site de macro ordonnancement.</p>	<p>Aucune dépendance</p>
<p>Frontend- 'W'</p>  <p>Remarque : le paramètre de ligne de commande qui suit le nom du processus de traitement par lots détermine le processus exécuté.</p> <p>W = Pertes</p>	<p>Pertes</p> <p>Le processus de traitement par lots recherche les groupes de produits de pertes planifiés pour le jour et crée un ajustement de stock (diminution) pour chaque article du groupe de produits. Le processus utilise des valeurs basées sur des pourcentages et des unités. Si un pourcentage et une unité existent, le processus applique la valeur la moins élevée. Prenons par exemple un article dont la valeur de stock disponible est 100. Si les 2 valeurs sont 10 % et 5 unités, le processus de traitement par lots crée un ajustement de stock de 5 unités pour l'article.</p> <p>Le processus de traitement par lots crée un enregistrement d'ajustement de stock terminé qui utilise le motif d'ajustement Pertes (code = 1) pour chaque article transmis au système de commercialisation.</p>	<p>Aucune dépendance</p>

Traitement par lots	Détails	Dépendances de traitement par lots
PurgeData	Le processus de purge supprime les données des tables de base de données tout en conservant l'intégrité de la base de données. Ce processus supprime les enregistrements de l'application SIM répondant à certains critères commerciaux (par exemple, les enregistrements marqués pour la suppression par l'utilisateur de l'application ou les enregistrements présents dans le système après un certain nombre de jours).	Aucune dépendance
ResaFileParser	Ce processus de traitement par lots importe les données de ventes et de retours provenant d'un système de point de vente (PDV). SIM utilise les données pour mettre à jour le stock disponible en fonction des combinaisons magasin/articles de chaque fichier. En d'autres termes, SIM obtient des informations à partir du processus de traitement par lots sur le déplacement du stock (ventes et retours). Une fois que SIM atteint les données, il considère que les ventes proviennent de la catégorie d'inventaire liée au rayon du magasin. Cela est important pour le processus de réapprovisionnement de rayon de SIM. De même, SIM considère que les retours doivent d'abord être dirigés vers la catégorie Réserve, la logique du système étant que les retours doivent être inspectés. Reportez-vous à l'« Annexe B » plus loin dans ce document pour connaître les spécifications de mise en page du fichier.	Aucune dépendance
ReturnNotAfterDateAlert	Ce traitement par lots avertit les utilisateurs « x » jours avant que le RF/RE est sur le point d'atteindre la date d'échéance et doit être expédié. Notez que la valeur « x » est configurable via les écrans de l'interface graphique du système.	Aucune dépendance

Traitement par lots	Détails	Dépendances de traitement par lots
rssAlertScan	Le processus de traitement par lots d'alertes email détermine depuis combien de temps les transferts disposent du statut En attente, puis envoie une alerte email selon des paramètres de date définis par le détaillant. Le processus de traitement par lots vérifie d'abord tous les transferts expédiés et le nombre de jours configurable. Si le programme trouve un transfert expédié dont la date d'expédition correspond ou est antérieure au nombre de jours établis, une alerte email est envoyée au magasin expéditeur et au magasin destinataire.	Aucune dépendance
SameDayPriceChangeActivator	Le traitement par lots introduit les modifications de prix de la journée dans la rubrique JMS pour que RPOS récupère les modifications de prix.	Aucune dépendance
ThirdPartyStockCountParser	Ce processus de traitement par lots importe un fichier de comptage d'un système de comptage tiers (RGIS par exemple). SIM met à jour ses enregistrements de comptage de type Unité et valeur à l'aide des données article/quantité renvoyées.	Aucune dépendance

Détails supplémentaires concernant ThirdPartyStockCountParser

Indicateur admin d'autorisation automatique réglé sur Non

Si l'indicateur admin d'autorisation automatique est réglé sur Non, les énoncés suivants sont vrais :

- Le fichier d'importation contient les informations sur les articles et les quantités comptées. SIM renseigne la quantité comptée dans les enregistrements de comptage et spécifie une quantité autorisée égale à la quantité comptée. Une fois que le fichier a été importé depuis le système RGIS, le type d'enregistrement de comptage est « autoriser » et le statut est « en cours ».
- Si un ou des articles envoyés par RGIS n'étaient pas encore ordonnancés dans un magasin, SIM ajoute ces articles à l'enregistrement de comptage approprié (en fonction du rayon) et la valeur répliquée du stock disponible est 0.
- Pendant l'importation de RGIS à SIM, toute donnée d'article « inconnue » est inscrite dans le tableau Non classé.

Indicateur admin d'autorisation automatique réglé sur Oui

Si l'indicateur admin d'autorisation automatique est réglé sur Oui, les énoncés suivants sont vrais :

- Le fichier d'importation contient les informations sur les articles et les quantités comptées. SIM renseigne la quantité comptée dans les enregistrements de comptage et spécifie une quantité autorisée égale à la quantité comptée. Une fois que le fichier a été importé depuis le système RGIS, le type d'enregistrement de comptage est « autoriser » et le statut est « terminé ».
- Si un ou des articles envoyés par RGIS n'étaient pas encore ordonnancés dans un magasin, SIM ajoute ces articles à l'enregistrement de comptage approprié (en fonction du rayon) et la valeur répliquée du stock disponible est 0.
- Pendant l'importation de RGIS à SIM, toute donnée d'article « inconnue » est inscrite dans le tableau Non classé.
- Une fois l'importation terminée, SIM autorise automatiquement les comptages de type Unité et valeur et exporte les données du comptage dans RMS. Dans des conditions d'utilisation normales, ce processus manuel est déclenché par un utilisateur SIM à partir du système « frontal ». Si l'indicateur admin de magasin pour l'autorisation automatique d'un comptage tiers est réglé sur Oui, ce processus fait partie de l'importation du fichier tiers. Notez que dans ce cas, tout article considéré comme « non classé » n'est pas attribué à un article existant. On estime que ce processus commercial n'a lieu que si le détaillant a corrigé tous les écarts et les conflits de données avant d'exporter les données de comptage du système tiers. **On estime également qu'aucune donnée ne sera revue ni modifiée avec SIM.** Ce processus ne fait que mettre à jour SIM avec les données du comptage. SIM met alors à jour RMS avec ces mêmes données. L'utilisateur n'a pas besoin d'intervenir dans SIM pour que ce processus s'effectue.

Suppositions d'intégration

- Avant le comptage, RMS fournit un fichier « d'exportation d'articles » à RGIS afin que celui-ci puisse valider les articles scannés.
- Les articles provenant de RGIS sont identifiés par un numéro d'article RMS : RIN (numéro d'article Retek), code UPC ou tout autre numéro défini dans RMS.
- On considère que toutes les quantités transmises par RGIS sont exprimées dans l'unité de mesure (UDM) standard de l'article définie dans RMS (unités, KG, etc).
- Le fichier RGIS renvoie la quantité comptée totale de chaque article, même si l'article a été compté dans plusieurs zones du magasin (total cumulé de chaque article).
- Pour les articles inscrits dans les enregistrement de comptage de SIM pour lesquels le système RGIS n'a renvoyé aucune quantité comptée, SIM estime que la quantité est 0, valeur qu'il introduit dans l'enregistrement.
- Pour les articles dont la quantité de stock disponible est inscrite dans SIM mais dont la valeur du comptage RGIS est 0, le contrôle des écarts utilise la valeur des unités de variance (et non pas des pourcentages) pour déterminer si l'article présente un écart et doit être affiché sur le système frontal.

Processus à plusieurs tâches légères et processus multiples

Les processus de traitement par lots de SIM ne sont actuellement *pas* configurés pour exécuter plusieurs tâches légères ou pour un usage multiple.

Redémarrage et reprise

La plupart des processus de traitement par lots Java de SIM n'utilisent aucun type de redémarrage ou de reprise. En revanche, si un redémarrage est nécessaire, il suffit de redémarrer le processus de traitement par lots. Reportez-vous à la section « Processus spécifique à PosmodDnldFileParser et à DexnexFileParser » ci-dessous pour connaître les exceptions.

Processus spécifique à DexnexFileParser

Pour obtenir des informations sur la connexion à SIM pour exécuter des processus de traitement par lots autres que l'analyse syntaxique effectuée par DexnexFileParser, reportez-vous au chapitre 2, « Configuration du système d'arrière-plan ».

- Fichier journal pour DexnexFileParser
Un fichier journal pour la partie analytique de DexnexFileParser se trouve dans le répertoire dexnexErrors.

Annexe A – Spécifications de mise en page du fichier de comptage

Fichier de configuration rmsupload.cfg

Le fichier de configuration rmsupload.cfg spécifie l'emplacement du fichier de sortie du comptage de type Unité et valeur qui doit être chargé dans RMS. Ce répertoire par défaut est le suivant :

```
\retek\sim\files\prod\upload\
```

Ce répertoire n'existe pas à l'origine mais est automatiquement créé dès que le premier comptage de type Unité et valeur est terminé. Le répertoire peut également être créé manuellement.

Spécifications du fichier plat des résultats de comptage

Lorsqu'un comptage est autorisé et terminé, le serveur SIM crée un fichier plat lors de l'exécution, qu'il place dans un répertoire configuré pendant l'installation. En utilisant le fichier plat généré par SIM, le module de téléchargement de stock du système de commercialisation récupère et télécharge les données de comptage physique. Le format du fichier est le suivant :

Nom de l'enregistrement	Nom du champ	Type de champ	Description
En-tête du fichier	Descripteur de l'enregistrement du type de fichier	Char(5)	« FHEAD » codé en dur
	Identifiant de la ligne de fichier	Number(10)	Numéro de la ligne actuellement traitée, codé en dur, « 000000001 »
	Type de fichier	Char(4)	« STKU » codé en dur
	Date de création du fichier	Date(14) YYYYMM DDHHMIS S	Date inscrite par un programme de conversion
	stocktake_date	Date(14) YYYYMM DDHHMIS S	stake_head.stocktake_date
	Comptage du cycle	Number(8)	stake_head.cycle_count
	loc_type	Char(1)	« W » ou « S » codé en dur.
	site	Number(10)	stake_location.wh ou stake_location.store

Retek Store Inventory Management

Nom de l'enregistrement	Nom du champ	Type de champ	Description
Enregistrement de la transaction	Descripteur de l'enregistrement du type de fichier	Char(5)	« FDETL » codé en dur
	Identifiant de la ligne de fichier	Number(10)	Numéro de la ligne actuellement traitée. Augmente selon un processus interne
	Type d'article	Char(3)	« ITM » codé en dur
	valeur de l'article	Char(25)	Numéro de l'article
	Quantité d'inventaire	Number(12, 4)	Nombre total d'unités ou poids total
	Description du site	Char(30)	Emplacement précis de l'article dans le site. Ex : Ex. : arrière de la réserve ou en vitrine.
Fin de fichier	Descripteur de l'enregistrement du type de fichier	Char(5)	« FTAIL » codé en dur.
	Identifiant de la ligne de fichier	Number(10)	Numéro de la ligne actuellement traitée. Augmente selon un processus interne.
	Comptage d'enregistrement du fichier	Number(10)	Nombre d'enregistrements de détails

Annexe B – Spécifications de mise en page du fichier de traitement par lots

Fichier plat utilisé lors du processus de traitement par lots ResaFileParser

Le fichier ci-dessous contient les données de ventes et de retours provenant d'un système de point de vente (PDV). Il est envoyé à SIM via ReSA dans un fichier plat. Le format du fichier est le suivant :

Nom de l'enregistrement	Nom du champ	Type de champ	Valeur par défaut	Description	Requis
En-tête du fichier	Descripteur de l'enregistrement du type de fichier	Char(5)	FHEAD	Identifie le type d'enregistrement de fichier	
	Identifiant de la ligne de fichier	Char(10)	Spécifié par le système externe	Numéro de la ligne actuellement traitée par le fichier en lecture.	Oui
	Définition du type de fichier	Char(4)	POSU	Identifie le fichier comme « chargement PDV »	Oui
	Date de création du fichier	Char(14)	Date de création	La date du fichier a été inscrite par un système externe	Oui
	Numéro du site	Number(4)	Spécifié par le système externe	Identifiant de l'entrepôt ou du magasin	Oui
	Indicateur d'inclusion de la TVA	Char(1)		Détermine si les valeurs du magasin comprennent la TVA. Non requis mais renseigné par l'audit des ventes de Retek	Oui

Nom de l'enregistrement	Nom du champ	Type de champ	Valeur par défaut	Description	Requis
	Région de la TVA	Number(4)		Région de TVA dans laquelle se trouve le site. Non requis mais renseigné par l'audit des ventes de Retek	Oui
	Code de devise	Char(3)		Devise du site. Non requis mais renseigné par l'audit des ventes de Retek	Oui
	Décimales de devise pour le prix de vente	Number(1)		Nombre de décimales pris en charge par une devise pour les prix de ventes. Non requis mais renseigné par l'audit des ventes de Retek	Oui
En-tête de la transaction	Descripteur de l'enregistrement du type de fichier	Char(5)	THEAD	Identifie le type d'enregistrement de la transaction.	
	Identifiant de la ligne de fichier	Char(10)	Spécifié par le système externe	Numéro de la ligne actuellement traitée par le fichier en lecture.	Oui
	Date de la transaction	Char(14)	Date de la transaction	Date à laquelle la transaction de vente/retour a été traitée au point de vente	Oui
	Type d'article	Char(6)	UPC ou SKU	Le type d'article est représenté par un code de produit universel (UPC) ou une référence	Oui

Annexe B – Spécifications de mise en page du fichier de traitement par lots

Nom de l'enregistrement	Nom du champ	Type de champ	Valeur par défaut	Description	Requis
	Valeur de l'article	Char(13)	Identifiant de l'article	Numéro du code de produit universel ou de la référence.	Oui
	Supplément	Char (5)	Identifiant supplémentaire	Utilisé pour une identification supplémentaire d'un article UPC ou le numéro de référence pré-pack	Non
	System_ind	Char (1)	'S' – référence d'article de base 'f' – référence de mode 'P' – article de pack	Type d'article vendu ou retourné. Non requis mais renseigné par l'audit des ventes de Retek	Oui
	Rayon	Number(4)	Rayon de l'article	Rayon de l'article vendu ou retourné. Non requis mais renseigné par l'audit des ventes de Retek	Oui
	Famille	Number(4)	Famille de l'article	Famille de l'article vendu ou retourné. Non requis mais renseigné par l'audit des ventes de Retek	Oui
	Sous-famille	Number(4)	Sous-famille de l'article	Sous-famille de l'article vendu ou retourné. Non requis mais renseigné par l'audit des ventes de Retek	Oui

Nom de l'enregistrement	Nom du champ	Type de champ	Valeur par défaut	Description	Requis
	Type de perte	Char (6)	Type de perte de l'article	Type de perte d'article vendu ou retourné. Non requis mais renseigné par l'audit des ventes de Retek	Oui
	Pourcentage de perte	Number(12)	Pourcentage de perte de l'article	Pourcentage de perte de l'article vendu ou retourné avec 4 décimales possibles. Non requis mais renseigné par l'audit des ventes de Retek	Oui
	Type de transaction	Char (1)	'S' – ventes 'R' - retour	Code du type de transaction spécifiant si la transaction est une vente ou un retour	Oui
	Quantité totale des ventes	Number(12)		Nombre d'unités vendues dans une zone spécifique avec 4 décimales possibles.	Oui
	Signe des ventes	Char (1)	'P' - positif 'N' - négatif	Précise si la quantité totale des ventes et la valeur totale des ventes sont positives ou négatives.	Oui
	Valeur totale des ventes	Number(20)		Valeur des ventes, valeur des ventes nette des marchandises vendues/retournées avec 4 décimales possibles.	Oui

Annexe B – Spécifications de mise en page du fichier de traitement par lots

Nom de l'enregistrement	Nom du champ	Type de champ	Valeur par défaut	Description	Requis
	Date de la dernière modification	Char (14)		Pour une utilisation administrative ultérieure.	
Détail de la transaction	Descripteur de l'enregistrement du type de fichier	Char (5)	TDETL	Identifie le type d'enregistrement de la transaction.	
	Identifiant de la ligne de fichier	Char (10)	Spécifié par le système externe	Numéro de la ligne actuellement traitée par le fichier en lecture.	Oui
	Type de transaction promotionnelle	Char (6)	Type de promotion - pour les valeurs valides, reportez-vous à la table code_detail.	Code pour le type de promotion depuis code_detail, code_type = PRMT	Oui
	Numéro de promotion	Number(4)	Numéro de promotion	Numéro de promotion depuis RMS	Non
	Quantité des ventes	Number(12)		Nombre d'unités vendues pour cette promotion avec 4 décimales possibles.	Oui
	Valeur des ventes	Number(20)		Valeur des unités vendues pour cette promotion avec 4 décimales possibles.	Oui
	Valeur de la remise	Number(20)		Valeur de la remise pour cette promotion avec 4 décimales possibles.	Oui

Nom de l'enregistrement	Nom du champ	Type de champ	Valeur par défaut	Description	Requis
Fin de la transaction	Descripteur de l'enregistrement du type de fichier	Char (5)	TTAIL	Identifie le type d'enregistrement de fichier	
	Identifiant de la ligne de fichier	Char (10)	Spécifié par le système externe	Numéro de la ligne actuellement traitée par le fichier en lecture.	Oui
	Comptage de la transaction	Number(6)	Spécifié par le système externe	Nombre d'enregistrements TDETL dans cette transaction	Oui
Fin de fichier	Descripteur de l'enregistrement du type de fichier	Char (5)	FTAIL	Identifie le type d'enregistrement de fichier	
	Identifiant de la ligne de fichier	Number(10)	Spécifié par le système externe	Numéro de la ligne actuellement traitée par le fichier en lecture.	Oui
	Compteur de l'enregistrement du fichier	Number(10)		Nombre d'enregistrements /transactions traités dans le fichier en cours (enregistrements entre l'en-tête et la fin uniquement).	Oui

Fichier plat utilisé lors du processus de traitement par lots DexnexFileParser

Structure du fichier : livraison 894

DEX/NEX utilise la transaction EDI 894 standard pour communiquer avec le système de réception des livraisons directes. Le format de base pour le fichier est le suivant :

En-tête

ST = en-tête de la transaction

G82 = enregistrement de base
livraison/retour

N9 = identification de référence

Détails (répété...)

LS = en-tête de la boucle

G83 = détails de la ligne article de la livraison directe au magasin

G72 = remise ou frais au niveau des détails

LE = fin de la boucle

Récapitulatif

G84 = totaux d'enregistrement de
livraison/retour

G86 = signature

G85 = vérification de l'intégrité de
l'enregistrement

SE = fin de la transaction

ST : contient le numéro de la transaction (894 par exemple) et un numéro de contrôle.

G82 : contient le type de livraison (livraison ou retour), des informations sur le fournisseur et la date de livraison.

N9 : contient des informations supplémentaires sur le fournisseur (Canada uniquement).

LS : contient un numéro pour les boucles de détails à suivre.

G8 : contient le numéro de l'article, la quantité, l'UDM, le coût unitaire et la description de l'article.

G72 : contient la remise (par exemple 10 %) ou des informations sur les frais (par exemple la taxe de pollution).

LE : contient la fin de la boucle.

G84 : contient la quantité totale et le coût de la livraison.

G86 : contient la signature UCC des fournisseurs.

G85 : contient un identifiant d'authentification.

SE : contient le nombre de transactions dans la transmission.

Détails du fichier :

Segment	Sous-segment	Nom	Requis	Valeur SIM
ST		En-tête de transaction	Oui	
ST	ST01	Numéro de la transaction	Oui	894 : identifie le type de fichier EDI, utilisé lors de la validation.
ST	ST02	Numéro de contrôle de la transaction	Oui	Ignorer
G82		Enregistrement de base livraison/retour	Oui	
G82	G8201	Indicateur de débit/crédit	Oui	D = livraison, C = retour
G82	G8202	Numéro de retour/livraison du fournisseur	Oui	Utilisé comme numéro de commande du fournisseur.
G82	G8203	Numéro DUNS	Oui	Ignorer
G82	G8204	Numéro de zone du destinataire	Oui	Contient le numéro du magasin
G82	G8205	Numéro DUNS	Oui	Numéro DUNS du fournisseur : utilisé pour déterminer le fournisseur
G82	G8206	Numéro de zone du fournisseur	Oui	Zone DUNS du fournisseur : utilisé avec le numéro DUNS pour déterminer le fournisseur
G82	G8207	Date de livraison/de retour	Oui	Date de livraison
N9		Identification de référence	Non	
N9	N901	Qualitatif de l'identifiant de référence	Oui	Ignorer
N9	N902	Numéro de référence	Oui	Utilisé comme numéro de facturation SIM
N9	N903	Description format libre	Non	Ignorer
LS	LS01	En-tête de la boucle	Oui	Fournit un numéro pour la boucle à suivre dans le fichier.
G83		Détails de la ligne article	Oui	

Annexe B – Spécifications de mise en page du fichier de traitement par lots

Segment	Sous-segment	Nom	Requis	Valeur SIM
G83	G8301	Numéro de la livraison directe au magasin	Oui	Ignorer
G83	G8302	Quantité	Oui	Quantité d'unités
G83	G8303	Code de l'unité de mesure	Oui	CA = colis, EA = unité
G83	G8304	UPC		Numéro de l'article
G83	G8305	Qualitatif du numéro du produit		
G83	G8306	Numéro du produit		
G83	G8307	Code UPC du colis	Non	Numéro du pack
G83	G8308	Coût de la liste d'articles	Non	Coût unitaire
G83	G8309	Pack	Non	
G83	G8310	Description de la caisse	Non	Ignorer
G72		Code de remise ou de frais	Non	Ignorer
G72	G7201	Code de remise ou de frais		Ignorer
G72	G7202	Code de gestion de remise ou des frais		Ignorer
G72	G7203	Numéro de remise ou de frais		Ignorer
G72	G7205	Taux de remise ou de frais		Ignorer
G72	G7206	Quantité de remise ou de frais		Ignorer
G72	G7207	Code de l'unité de mesure		Ignorer
G72	G7208	Montant total de la remise ou des frais		Ignorer
G72	G7209	Pourcentage de remise ou de frais		Ignorer
G72	G7210	Valeur en dollar pour le pourcentage de remise ou de frais		Ignorer

Segment	Sous-segment	Nom	Requis	Valeur SIM
LE	LE01	Identifiant de la boucle		Fin de boucle, contenant le même numéro que l'en-tête de la boucle.
G84		Totaux des enregistrements de livraison ou de retour.	Oui	
G84	G8401	Quantité	Oui	Somme de toutes les valeurs G8302.
G84	G8402	Montant total de la facture	Oui	Coût total, comprenant les frais sans les remises
G86	G8601	Signature	Oui	Ignorer
G85	G8501	Valeur de vérification de l'intégrité	Oui	Ignorer
SE	SE01	Nombre de segments inclus	Oui	Nombre total de segments entre ST et SE. Utilisé pour la validation.
SE	SE02	Numéro de contrôle de la transaction	Oui	Identique à ST02. Utilisé pour la validation.
GE	GE01	Nombre de transactions incluses	Oui	Nombre d'ensembles dans un groupe fonctionnel. Utilisé pour la validation.
GE	GE02	Numéro de contrôle du groupe	Oui	Identique à GS06. Utilisé pour la validation.

Fichier plat utilisé lors du processus de traitement par lots ThirdPartyStockCountParser

Définition de la mise en page des fichiers dans RGIS

- Nombre de champs : 9
- Longueur de l'enregistrement : 80

Nom des données	Nom du champ	Longueur déc.	Positionner depuis	Positionner jusqu'à	Type de champ
DLSSTR	NUMERO MAGASIN	6	1	6	Character
DLSDAT	DATE MMJJAA	6	7	12	Character
DLSRAN	NUMERO ZONE RGIS	10	13	22	Character
DSLFI2	12 CARACTERES DE REMPLISSAGE	12	23	34	Character
DSLFI3	13 CHARACTER FILLER	13	35	47	Character
DLSUPC	CODE UPC	13	48	60	Character
DLSFI2	12 ZERO DE REMPLISSAGE	12	61	72	Character
DLSQTY	QUANTITE COMPTAGE	7	73	79	Character
DLSFI01	CONSTANTE D'UN "+"	1	80	80	Character

Données des fichiers d'exemple RGIS

```

00030105010212      068853600204 00000000000000000025
00030105010212      024000010265 00000000000000000007
00030105010212      027000422380 00000000000000000019
00030105010212      024000010265 00000000000000000004
00030105010212      755566004718 00000000000000000027
00030105010212      074027062006 00000000000000000017
00030105010212      074027062006 00000000000000000005
00030105010212      074027062006 00000000000000000003
00030105010212      035549874270 00000000000000000012
00030105010212      074027075464 00000000000000000003
00030105010212      042600065492 00000000000000000006
00030105010212      070320801199 00000000000000000014
00030105010212      067703680038 00000000000000000005
00030105010212      030267300667 00000000000000000009
00030105010212      045700155001 00000000000000000001
00030105010212      755566004718 00000000000000000018

```