

IBM Cúram Social Program Management
Version 6.0.5

Guide des opérations Cúram



Important

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations contenues dans la section «Remarques», à la page 31

Dernière révision : Mars 2014

Cette édition s'applique à IBM Cúram Social Program Management version 6.0.5 et à toutes les versions ultérieures, sauf indication contraire dans les nouvelles éditions.

Eléments sous licence - Propriété d'IBM.

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

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

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

Vous pouvez également consulter les serveurs Internet suivants :

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

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

© Copyright IBM France 2014. Tous droits réservés.

© Copyright IBM Corporation 2012, 2013.

© Cúram Software Limited. 2011. All rights reserved.

Table des matières

Figures v

Tableaux vii

Avis aux lecteurs canadiens. ix

Administration d'un système de production 1

Introduction	1
Préface	1
Références et autres informations	1
Configuration avancée	1
Présentation	1
Stockage de modèles XSL à utiliser dans l'application	1
Création de la base de données de production dans l'application.	2
Problèmes de conversion de données	3
Création de votre organisation dans l'application	4
Exécution du système en ligne Cúram	4
Présentation	4
Démarrage du système	4
Serveur XML Cúram.	4
Cúram Server Application	4
Arrêt du système	5
Cúram Server Application	5
Serveur XML Cúram.	5
Journaux système.	5
IBM WebSphere Application Server.	5
Serveur Oracle WebLogic	6
Exécution de traitements par lots dans l'application	6
Présentation	6
Présentation des fonctions des traitements par lots	6
Lanceur par lots	6

Demandes de traitements par lots	7
Exécution du lanceur par lots.	7
Serveur XML Cúram.	7
Configuration de l'interface en ligne des traitements par lots	7
Gestion des groupes de traitements par lots	8
Exécution de traitements par lots autonomes	9
Format de la commande	10
Planification des traitements par lots	10
Journaux système	10
Traitements par lots de base	11
Présentation	11
Paramètres par lots et date de traitement	11
Traitement Cúram	11
Gestion des dossiers	11
Gestionnaire de dépendances	16
Financial	18
Flux de travaux	24
Interfaces d'entrée	26
LoadPaymentsReceived	26
Interfaces de sortie	27
Export de calendrier	27
Générer l'interface de grand livre	28
Paramètres de configuration Cúram	28
Introduction	28
Serveur XML Cúram	28
Cúram Server Application	28
Gestion des paramètres de configuration	28
Modification des paramètres de configuration	29
Paramètres de configuration disponibles.	29

Remarques 31

Politique de confidentialité	33
Marques	34

Figures

Tableaux

Avis aux lecteurs canadiens

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

Illustrations

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

Terminologie

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

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

Claviers

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

OS/2 et Windows - Paramètres canadiens

Au Canada, on utilise :

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

Nomenclature

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

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

Brevets

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

Assistance téléphonique

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

Administration d'un système de production

Vous pouvez gérer des modèles XSL, des conversions de données de base de données de production, des traitements par lots et des structures d'organisation pour un système de production d'environnement d'exécution Cúram. Un serveur XML Cúram et une application serveur doivent être configurés pour le déploiement de Cúram. Le fichier Application.prx est utilisé pour définir les paramètres de configuration initiale dans la base de données.

Introduction

Préface

Ce guide décrit comment administrer un système d'exécution (de production) Cúram.

Il ne traite *pas* de l'installation du système de développement Cúram. Si vous installez l'application à des fins de développement et de personnalisation, consultez le manuel Cúram - Guide d'installation.

Références et autres informations

- Cúram - Architecture de la solution
- Cúram - Guide d'administration du système
- Cúram Batch Processing Guide
- Guide de déploiement Cúram pour WebLogic Server
- Guide de déploiement Cúram pour WebSphere Application Server
- Guide de déploiement Cúram pour WebSphere Application Server sur z/OS
- Cúram - Guide d'installation
- Cúram XML Infrastructure Guide

Configuration avancée

Présentation

Ce chapitre présente des étapes de configuration avancée que vous pouvez réaliser sur votre système de production Cúram. Voici une liste des principaux thèmes abordés :

- Stockage de modèles XSL à utiliser dans l'application
- Création de la base de données de production dans l'application
- Problèmes de conversion de données
- Création de votre organisation dans l'application

Si vous installez l'application à des fins de démonstration ou d'enquête, ce chapitre peut être ignoré.

Stockage de modèles XSL à utiliser dans l'application

L'application utilise le serveur XML Cúram pour mettre en forme et imprimer les correspondances pro forma. Pour ce faire, le serveur XML Cúram fusionne un

modèle de lettre (au format XSL) avec du texte libre saisi par l'utilisateur et des données récupérées depuis la base de données.

Des modèles XSL d'exemple sont insérés dans l'utilitaire de configuration des bases de données décrit dans le chapitre Exécution Cúram du manuel Cúram - Guide d'installation correspondant à votre type de plateforme. Vous pouvez créer vos propres modèles XSL à l'aide de votre outil d'édition XSL habituel disponible sur votre plateforme matérielle.

Remarque : L'application ne contient pas d'outils d'édition XSL.

Pour rendre un modèle disponible au serveur XML Cúram, procédez comme suit.

1. Connectez-vous à l'application d'administration système à l'aide de vos nom d'utilisateur et mot de passe¹.
2. La page d'accueil s'affiche. Cliquez sur la section Configurations du système.
3. Cliquez sur le lien Communications - XSL Templates (Communications-Modèles XSL) dans le panneau de raccourcis.
4. La page Modèles s'affiche et présente la liste des modèles XSL stockés dans le système.
5. Cliquez sur le bouton Créer pour ajouter un modèle.
6. La page Create Template (Créer un modèle) s'affiche et demande des informations sur votre modèle. Cliquez sur le bouton Parcourir pour localiser le modèle sur votre système de fichiers. Entrez les informations de votre modèle et cliquez sur Enregistrer.

Remarque : La version de XSL prise en charge est la version XSL-FO 1.0 recommandée par le W3C.

Création de la base de données de production dans l'application

Le chapitre Exécution Cúram du manuel Cúram - Guide d'installation correspondant à votre plateforme décrit la création d'une base de données simple pour l'installation de l'application.

Toutefois, dans un environnement de production, il est probable que vous devrez effectuer de nombreuses actions supplémentaires pour créer dans l'application une base de données répondant à vos pratiques d'administration des bases de données. Vous pouvez suivre les étapes ci-après pour créer une base de données pouvant être utilisée avec une installation de production de l'application :

1. A l'invite de commande, accédez au répertoire situé à la racine du répertoire d'installation d'exécution.
2. Exécutez la commande suivante :
`build database`
3. Passez en revue le contenu du sous-répertoire build/datamanager situé dans votre répertoire d'installation. Ce répertoire contient plusieurs scripts générés qui définissent la structure et les données initiales de la base de données.
4. Affinez ou restructurez les commandes DDL/SQL selon vos besoins. Vous pouvez par exemple :
 - Spécifier la création de votre base de données physique, par exemple mettre en oeuvre une stratégie de partitionnement

1. L'application fournit un nom d'administrateur (sysadmin) et un mot de passe (password) par défaut.

- Partager la base de données physique avec d'autres applications
 - Indiquer les attributs physiques des tables dans l'application, par exemple, le taux de croissance attendu
 - Personnaliser les données initiales requises par l'application, par exemple modifier les privilèges fonctionnels de vos rôles de sécurité
 - Omettre les contraintes de clé externe, car elles ne sont pas prises en charge dans un environnement de production²
 - Charger des données de systèmes existants (voir «Problèmes de conversion de données» ci-après)
 - Affiner la stratégie d'indexation, par exemple implémenter des index avec hachage
5. Utilisez vos scripts affinés/restructurés (en plus des procédures administratives qui peuvent déjà être mises en place) pour créer votre base de données de production dans l'application.

Problèmes de conversion de données

Dans un environnement de production, vous devrez probablement migrer vos données vers la base de données Cúram.

La description d'une telle migration n'entre pas dans le cadre de ce document, car cette opération nécessite une connaissance approfondie des éléments suivants :

- Modèle de référence Cúram et personnalisations éventuelles
- Structure et intégrité des données existantes
- Toute exigence de synchronisation continue des données entre Cúram et vos autres systèmes
- Toute procédure de migration dont vous pouvez disposer

En plus des éléments ci-dessus, il convient de tenir compte des suggestions suivantes :

- Des données initiales d'exemple nécessaires au démarrage du serveur en ligne Cúram sont fournies. Ces données initiales contiennent notamment l'administrateur (admin). Elles peuvent être personnalisées avant leur chargement dans la base de données ou par le biais de l'application elle-même (avant la mise en ligne).
- Des données de démonstration sont fournies. Nous vous suggérons de ne pas les charger dans votre base de données de production.
- Vous pouvez pré-trier vos données conformément à votre stratégie de mise en cluster.
- Le modèle de référence Cúram inclut des contraintes de clé externe qui facilitent le maintien de l'intégrité des données Cúram. Lorsque ces contraintes sont appliquées avant le chargement des données migrées, des violations peuvent se produire si vos données ne sont pas chargées dans l'ordre "parent-enfant"³ Nous suggérons donc d'appliquer les contraintes de clé externe après le chargement de

2. L'application de référence Cúram applique l'intégrité référentielle. Si vous utilisez une intégrité référentielle appliquée par base de données, les performances seront donc dégradées. Il est fortement recommandé de ne pas créer de contrainte de clé externe sur votre base de données dans un environnement de production.

3. Dans une base de données relationnelle, une entité parent est souvent associée à zéro, une ou plusieurs entités "enfant". Chacune de ces entités porte la clé de ses parents, et le parent doit donc être créé en premier pour que la clé soit disponible (pour définir la clé du parent sur l'enfant) lorsque l'enfant est stocké par la suite.

vos données migrées⁴. Toute contrainte rejetée par la base de données indique un problème d'intégrité dans vos données migrées.

- Le modèle de référence Cúram inclut des index permettant la prise en charge de toutes les requêtes SQL utilisées depuis Cúram Server Application. Vous pouvez supprimer certains de ces index pour améliorer les performances d'écriture de la base de données, mais vous dégraderez alors les performances des requêtes peu souvent utilisées. Cette opération ne peut être effectuée que lorsque vous savez quelles transactions de l'application seront utilisées fréquemment en ligne dans votre installation et lesquelles seront rarement utilisées en ligne ou seulement utilisées par lots.

Création de votre organisation dans l'application

La base de données de l'application doit contenir des informations sur votre organisation, ses unités organisationnelles et ses utilisateurs. Des données exemples vous permettront de démarrer Cúram Online Server Application et de vous connecter depuis un client Cúram. Ces données incluent un exemple de structure d'organisation. Une structure d'organisation se compose d'une hiérarchie d'unités organisationnelles dans laquelle chaque unité peut se voir affectée un nombre quelconque de postes, chaque poste se voyant lui-même affecté un nombre quelconque d'utilisateurs

La configuration des données relatives à votre organisation est décrite dans le manuel Cúram - Guide d'administration.

Exécution du système en ligne Cúram

Présentation

Ce chapitre décrit comment exécuter et administrer le système en ligne de Cúram.

Démarrage du système

Cette section explique comment démarrer le système en ligne Cúram. Notez que Cúram Server Application doit être démarrée après le serveur XML Cúram⁵.

Serveur XML Cúram

Pour démarrer le serveur XML Cúram, exécutez la commande suivante à partir du sous-répertoire XMLServer, situé dans le répertoire d'installation :

```
ant -f xmlserver.xml
```

Pour Microsoft Windows : Sous Microsoft[®] Windows, vous pouvez également démarrer le serveur XML Cúram en cliquant sur le raccourci **Démarrer > Programmes > Cúram > XML Server (Serveur XML) > Start XML Server (Démarrer le serveur XML)**.

Consultez Cúram XML Infrastructure Guide pour plus d'informations.

Cúram Server Application

Pour démarrer Cúram Server Application, exécutez la commande suivante à partir du répertoire de l'installation de l'exécution Cúram :

4. Il est recommandé de créer de telles contraintes pour identifier les problèmes d'intégrité de vos données converties, puis, une fois que les problèmes sont résolus, de les supprimer, car elles ne sont pas prises en charge pour les bases de données de production.

5. Si votre configuration d'exécution inclut le serveur XML Cúram.

```
build startserver -Dserver.name=curam1
```

où curam1 est la valeur de la propriété curam.server.name.

Pour Microsoft Windows : Vous pouvez également démarrer Cúram Server Application à l'aide des méthodes suivantes :

- Cliquez sur le raccourci **Démarrer > Programmes > Cúram > Application Server (Serveur d'application) > Online (En ligne) > Start Server (Démarrer le serveur)**
- A partir du répertoire d'installation de l'exécution Cúram, exécutez la commande suivante :
StartServer.bat

Reportez-vous au guide de déploiement Cúram correspondant à votre serveur d'application pour plus d'informations.

Arrêt du système

Cette section explique comment arrêter le système en ligne Cúram. Notez que Cúram Server Application doit être arrêtée avant le serveur XML Cúram.

Cúram Server Application

Pour arrêter Cúram Server Application, exécutez la commande suivante à partir du répertoire d'installation de l'exécution Cúram :

```
build stopserver -Dserver.name=curam1
```

où curam1 est la valeur de la propriété curam.server.name.

Windows :: Vous pouvez également arrêter Cúram Server Application à l'aide des méthodes suivantes :

- Cliquez sur **Démarrer > Programmes > Cúram > Application Server (Serveur d'application) > Online (En ligne) > Stop Server (Arrêter le serveur)**
- A partir du répertoire d'installation de l'exécution Cúram, exécutez la commande suivante :
StopServer.bat

Remarque : Si vous avez l'intention d'arrêter Cúram Server Application, vous devez informer les utilisateurs en ligne qu'ils doivent terminer leur travail avant l'arrêt du serveur.

Serveur XML Cúram

Pour arrêter le serveur XML Cúram, entrez ^C.

Windows :: Pour arrêter le serveur XML Cúram, vous pouvez également cliquer sur "X" pour fermer la fenêtre console.

Consultez Cúram XML Infrastructure Guide pour plus d'informations.

Journaux système

IBM WebSphere Application Server

Pour IBM[®] WebSphere[®] Application Server, par défaut, les journaux en ligne sont stockés dans le répertoire <répertoire d'installation WebSphere>\AppServer\

profiles\AppServ01\logs\

Serveur Oracle WebLogic

Pour le serveur Oracle[®] WebLogic, par défaut, les journaux en ligne sont stockés dans le répertoire <répertoire d'installation BEA/user_projects/domains/<NomDomaine>/servers/<NomServeur> où <NomServeur> est le nom du serveur.

Exécution de traitements par lots dans l'application

Présentation

Ce chapitre explique comment exécuter des traitements par lots dans l'application. Il fournit des informations sur les points suivants :

- Présentation des fonctions des traitements par lots
- Lanceur par lots
- Exécution de traitements par lots autonomes
- Journaux système

Présentation des fonctions des traitements par lots

Les traitements par lots de l'application présentent les caractéristiques suivantes :

- Chaque traitement par lots est spécifié dans et généré à partir du modèle Cúram Rose. Pour plus de détails sur la modélisation des traitements par lots dans l'application, consultez le manuel Cúram Batch Processing Guide.
- Lors de l'exécution, chaque traitement par lots accepte des paramètres de configuration⁶.
- Chaque traitement par lots stocke un journal de sa progression (un journal est stocké à chaque appel d'un traitement par lots).
- Chaque journal est envoyé automatiquement par e-mail à un utilisateur nommé.

Lanceur par lots

Le lanceur par lots fournit un mécanisme de demande différée dans lequel :

- Les utilisateurs d'applications en ligne peuvent enregistrer leurs demandes d'exécution de traitements par lots.
- L'opérateur par lots peut facilement exécuter tous les traitements par lots requis par les utilisateurs en ligne.

Remarque : Le lanceur par lots exécute les traitements par lots dans l'ordre dans lequel les demandes des utilisateurs ont été reçues. Si les traitements par lots doivent s'exécuter dans un ordre particulier (par exemple, s'il existe des dépendances fonctionnelles entre eux), le lanceur par lots peut s'avérer inapproprié pour leur exécution ; consultez plutôt «Exécution de traitements par lots autonomes», à la page 9 ci-après.

6. La manière dont ces paramètres sont transmis au traitement par lots varie selon que :

- le traitement par lots est démarré par le biais du lanceur par lots ; ou
- le traitement par lots est démarré comme un traitement autonome.

Pour obtenir une description des paramètres communs à tous les traitements par lots, consultez le manuel Cúram Batch Processing Guide.

Demandes de traitements par lots

Pour enregistrer une demande d'exécution d'un traitement par lots, procédez comme suit.

1. Connectez-vous à l'application à l'aide de vos nom d'utilisateur et mot de passe⁷.
2. La page d'accueil s'affiche. Cliquez sur le raccourci d'administration.
3. La page d'accueil d'administration s'affiche. Cliquez sur la section Configurations du système.
4. Cliquez sur l'onglet Traitements par lots.
5. La page Traitements par lots s'affiche et présente la liste des traitements par lots disponibles.
6. Pour exécuter un traitement par lots, cliquez sur le bouton processus Exécute (Exécuter les processus).
7. La page Exécute Batch Process (Exécuter les traitements par lots) s'affiche et présente les paramètres acceptés par le traitement par lots. Entrez les paramètres appropriés au traitement par lots sélectionné (le cas échéant).
8. Cliquez sur Exécuter pour mettre votre traitement par lots en file d'attente d'exécution.
9. Le système enregistre la demande d'exécution du traitement par lots.

Exécution du lanceur par lots

Dans l'idéal, le lanceur par lots est exécuté lorsque l'application en ligne Cúram n'est pas disponible ; toutefois, s'il n'y a aucun risque que les programmes exécutés à partir du lanceur par lots verrouillent la base de données d'applications (par exemple avec un grand nombre de mises à jour non validées) pendant une période prolongée, vous pouvez exécuter le lanceur même lorsque l'application en ligne Cúram est disponible.

Pour exécuter le lanceur par lots, exécutez la commande suivante à partir du répertoire racine de votre installation d'exécution.

```
build runbatch
```

Pour Microsoft Windows : Vous pouvez également exécuter le lanceur par lots en cliquant sur le raccourci **Démarrer > Programmes > Curam > Runtime (Exécution) > Run Batch Launcher (Exécuter le lanceur par lots)**.

Serveur XML Cúram

Pour démarrer le serveur XML Cúram, exécutez la commande suivante à partir du sous-répertoire XMLServer, situé dans le répertoire d'installation :

```
ant -f xmlserver.xml
```

Configuration de l'interface en ligne des traitements par lots

La liste des traitements par lots en ligne peut être personnalisée en :

- gérant la liste des traitements par lots ;
- gérant les groupes de traitements par lots.

Gestion de la liste des traitements par lots : Les traitements par lots peuvent être ajoutés ou supprimés de la liste visible par un utilisateur de l'application en ligne⁸.

7. L'application fournit un nom d'administrateur (sysadmin) et un mot de passe (password) par défaut.

8. Initialement, cette liste contient tous les traitements par lots disponibles dans l'application.

Cela peut s'avérer utile si, par exemple, certains traitements par lots doivent s'exécuter en tant que programmes autonomes et non faire l'objet d'une demande d'exécution par un utilisateur ; ces programmes peuvent être supprimés de la liste.

Affichage de la liste des traitements par lots : Pour afficher la liste des traitements par lots, procédez comme suit:

1. Connectez-vous à l'application à l'aide de vos nom d'utilisateur et mot de passe⁹.
2. La page d'accueil s'affiche. Cliquez sur la section Configurations du système.
3. Cliquez sur l'onglet Traitements par lots.
4. La page Traitements par lots s'affiche et présente la liste des traitements par lots disponibles.

Ajout d'un traitement par lots à la liste : Pour ajouter un programme à la liste, procédez comme suit.

1. Accédez à la liste des traitements par lots comme décrit ci-dessus.
2. Cliquez sur le bouton Créer.
3. Une page s'affiche et demande des informations sur le nouveau traitement par lots. Entrez les informations demandées et cliquez sur Enregistrer.

Mise à jour d'un traitement par lots déjà dans la liste : Pour mettre à jour un traitement par lots de la liste, procédez comme suit.

1. Accédez à la liste des traitements par lots comme décrit ci-dessus.
2. Cliquez sur l'action Editer en regard du traitement par lots sélectionné.
3. Une page s'affiche et présente des informations sur le nouveau traitement par lots. Modifiez les informations et cliquez sur Enregistrer.

Suppression d'un traitement par lots de la liste : Pour supprimer un traitement par lots de la liste, procédez comme suit.

1. Accédez à la liste des traitements par lots comme décrit ci-dessus.
2. Cliquez sur l'action Supprimer en regard du traitement par lots sélectionné.

Gestion des groupes de traitements par lots

L'application prend en charge le regroupement des traitements par lots pour faciliter la catégorisation et la navigation¹⁰.

Remarque : Chaque traitement par lots peut appartenir à zéro, un ou plusieurs groupes.

Affichage de la liste des groupes de traitements par lots : Pour afficher la liste des groupes de traitements par lots, procédez comme suit.

1. Cliquez sur le lien Batch - Process Groups (Lot - Groupe de traitements) dans le panneau de raccourcis.
2. Une liste des groupes de traitements par lots s'affiche.

Ajout d'un groupe de traitements par lots à la liste : Pour ajouter un nouveau groupe à la liste, procédez comme suit.

1. Accédez à la liste des groupes de traitements par lots comme décrit ci-dessus.
2. Cliquez sur le bouton Créer.

9. L'application fournit un nom d'administrateur (sysadmin) et un mot de passe (password) par défaut.

10. Par exemple, vous pouvez créer un groupe contenant des rapports statistiques, un autre contenant des traitements financiers, etc.

3. Une page s'affiche et demande des informations sur le nouveau groupe de traitements par lots. Entrez les informations demandées et cliquez sur Enregistrer.

Suppression d'un groupe de traitements par lots de la liste : Pour supprimer un groupe de la liste, procédez comme suit.

1. Accédez à la liste des groupes de traitements par lots comme décrit ci-dessus.
2. Cliquez sur l'action Supprimer en regard du groupe de traitements par lots sélectionné.

Ajout d'un traitement par lots à un groupe : Pour ajouter un traitement par lots à un groupe, procédez comme suit.

1. Accédez à la liste des groupes de traitements par lots comme décrit ci-dessus.
2. Cliquez sur Afficher en regard du groupe de traitements par lots sélectionné.
3. Les détails du groupe de traitements par lots sont affichés. Cliquez sur le bouton Ajouter.
4. Une liste de traitements par lots s'affiche. Choisissez un traitement par lots et cliquez sur Enregistrer.

Suppression d'un traitement par lots d'un groupe : Pour supprimer un traitement par lots d'un groupe, procédez comme suit.

1. Accédez à la liste des groupes de traitements par lots comme décrit ci-dessus.
2. Cliquez sur Afficher en regard du groupe de traitements par lots sélectionné.
3. Une liste des traitements par lots du groupe s'affiche.
4. Cliquez sur Supprimer en regard du traitement par lots que vous souhaitez supprimer du groupe de traitements par lots.

Exécution d'un traitement par lots à partir d'un groupe : Une fois dans un groupe, un traitement par lots peut être exécuté à partir de l'arborescence des groupes comme suit :

1. Accédez à la liste des groupes de traitements par lots comme décrit ci-dessus.
2. Cliquez sur Afficher en regard du groupe de traitements par lots sélectionné.
3. Une liste des traitements par lots du groupe s'affiche.
4. Cliquez sur l'action Exécuter en regard du traitement par lots que vous souhaitez exécuter.

Exécution de traitements par lots autonomes

Chaque traitement par lots peut être exécuté dans un mode "autonome".

Les traitements par lots sont généralement exécutés en tant que programmes autonomes dans les cas suivants :

- Un traitement par lots est utilisé à des fins de démonstration ou de test.
- Il existe une dépendance fonctionnelle ou un ordre entre une suite de traitements par lots, c'est-à-dire qu'un traitement par lots particulier doit être terminé avant qu'un autre processus de lot particulier ne puisse démarrer.
- Un traitement par lots (ou une suite de programmes) est exécuté régulièrement (par exemple, la nuit) et il est fastidieux ou peu pratique d'utiliser le lanceur par lots pour enregistrer une demande d'exécution du traitement par lots chaque fois que nécessaire.

Format de la commande

Chaque traitement par lots accepte des paramètres d'entrée passés sur la ligne de commande. Les paramètres requis varient entre les traitements par lots¹¹.

Pour exécuter un traitement par lots en mode autonome, exécutez la commande suivante (sur une seule ligne) à partir du répertoire d'exécution :

```
build runbatch
-Dbatch.program=
curam.intf.
<nom classe de programme>.<nom d'opération du programme>
-Dbatch.username=<utilisateur du traitement par lots Cúram>
-Dbatch.parameters=
"<comma-separated list of parameter name=
value pairs>"
```

Exemple de ce type de commande :

```
build runbatch
-Dbatch.program=
curam.intf.PersonExtract.extractPersonDetails
-Dbatch.username=superuser
-Dbatch.parameters="extractFilePath=c:\testfile.dat"
```

Remarque : Si vous exécutez *un* seul traitement par lots (par ex, à des fins de test), il peut s'avérer pratique de supprimer les paramètres "-D" de la ligne de commande pour les placer dans le fichier `project\properties\Bootstrap.properties` de votre répertoire d'installation d'exécution Cúram.

Planification des traitements par lots

Si vous devez planifier l'exécution d'un traitement par lots (ou d'une suite de traitements par lots) à un moment précis, vous pouvez utiliser votre outil de planification habituel. Pour configurer votre outil de planification, les traitements par lots renvoient une valeur d'état comme spécifié dans votre table de codes `BatchErrorCodes`.

Remarque : Aucun outil de planification n'est inclus dans l'application.

Journaux système

Les journaux par lots sont stockés dans le sous-répertoire `buildlogs` de votre répertoire d'installation de l'exécution Cúram et sont nommés `<nom traitement par lots><date et horodatage>.log`.

Un journal est créé pour chaque appel de l'exécutable du traitement par lots. La progression du traitement par lots y est enregistrée. Le contenu du journal est automatiquement envoyé par courrier électronique à l'adresse configurée dans le fichier `project\properties\Bootstrap.properties` ou dans la zone `Properties Administration` (Administration des propriétés) de la page Administration de l'application.

11. Les paramètres acceptés par un traitement par lots peuvent être visualisés dans l'application Cúram en ligne avant la planification de l'exécution.

Traitements par lots de base

Présentation

Ce chapitre répertorie les traitements par lots qui fournissent les fonctionnalités de base de l'application.

Remarque : Sauf indication contraire, les traitements par lots peuvent être exécutés dans n'importe quel ordre.

Paramètres par lots et date de traitement

Un grand nombre de travaux par lots de l'application acceptent le paramètre 'processingDate'. Lorsqu'il est spécifié, il correspond à la date d'exécution prévue du travail par lots (date renvoyée par l'API `getCurrentDate()`). Lorsque ce paramètre n'est pas spécifié, le travail par lots est exécuté pour la date système. Ce paramètre n'est généralement pas répertorié dans les descriptions ci-après.

Traitement Cúram

Gestion des dossiers

GenerateCommunications : Ce traitement par lots génère des communications qui font partie de la gestion des dossiers, par exemple les communications relatives à l'approbation d'un dossier. Une variable d'environnement peut être définie pour que les communications relatives à la gestion des dossiers ne soient pas traitées automatiquement, mais passent au statut En attente. Ce traitement par lots génère toutes les communications en attente.

Ce programme est censé être exécuté chaque nuit. Toutefois, une exécution moins régulière n'aura pas de conséquence hormis un temps d'exécution plus long et la possibilité que certaines communications relatives aux dossiers ne soient pas traitées avant l'exécution suivante. L'exécution de ce programme plus d'une fois par jour n'a aucun effet.

Ce traitement par lots n'accepte aucun paramètre.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont
`curam.core.intf.GenerateCommunications.generateAllCommunications`.

ProductDeliveryFinalClosure : Ce traitement par lots permet de clôturer les dossiers dans l'état En attente de clôture, lorsque la période de grâce de clôture a expiré.

Ce programme est censé être exécuté chaque nuit. Toutefois, une exécution moins régulière n'aura pas de conséquence hormis un temps d'exécution plus long et la possibilité que certains dossiers aient dépassé la période de grâce de clôture avant d'être clôturés. L'exécution de ce programme plus d'une fois par jour n'a aucun effet.

Ce traitement par lots n'accepte aucun paramètre.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont
`curam.core.intf.CloseCasesPendingClosure.closeCasesPendingClosure`.

EvaluateCertificationGracePeriod : Ce traitement par lots permet de définir des dossiers sur En attente de clôture lorsqu'ils sont restés sans certification jusqu'à la fin de la période de grâce de certification.

Ce programme est censé être exécuté chaque nuit. Toutefois, une exécution moins régulière n'aura pas de conséquence hormis un temps d'exécution plus long et la possibilité que certains dossiers aient dépassé la période de grâce de certification avant d'être définis En attente de clôture. L'exécution de ce programme plus d'une fois par jour n'a aucun effet.

Ce traitement par lots n'accepte aucun paramètre.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.EvaluateCertificationGracePeriod.run`.

DetermineProductDeliveryEligibility : Ce traitement par lots permet de traiter les dossiers nouvellement approuvés et d'enregistrer les décisions liées à ces dossiers. Tout dossier non éligible se verra attribuer le statut En attente de clôture par ce processus, alors que les dossiers éligibles prendront l'état Actif. Si une erreur se produit lors du traitement d'un dossier, le traitement est suspendu et le propriétaire du dossier notifié par un courrier électronique ou une tâche.

Ce programme est censé être exécuté chaque nuit. Toutefois, une exécution moins régulière n'aura pas de conséquence hormis un temps d'exécution plus long et la possibilité que certains dossiers approuvés ne soient pas traités avant l'exécution suivante. L'exécution de ce programme plusieurs fois par jour entraîne le traitement de tous les dossiers approuvés du système au moment de l'exécution et ne provoque pas de problème particulier.

Ce traitement par lots accepte les paramètres suivants :

- **Product ID** (Identificateur de produit) - identificateur unique utilisé afin d'exécuter ce processus pour les dossiers d'un produit spécifique
Lorsqu'aucun identificateur de produit n'est spécifié, tous les dossiers sont traités.
- **Batch Process Instance ID** (Identificateur de l'instance du traitement par lots) - identificateur unique utilisé pour permettre l'exécution simultanée efficace de plusieurs instances d'un même traitement par lots.
Par exemple, ce processus peut s'exécuter pour plusieurs produits. Lorsqu'aucun identificateur d'instance n'est indiqué, une seule instance du traitement par lots peut être exécutée.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.DetermineProductDeliveryEligibility.process`.

DetermineProductDeliveryEligibilityStream : Ce traitement par lots permet de prendre en charge la diffusion en flux du processus `Determine Product Eligibility` (Déterminer l'éligibilité au produit). Il ne peut être exécuté qu'en association avec le processus `Determine Product Eligibility` (Déterminer l'éligibilité au produit). La diffusion en flux des traitements par lots permet l'exécution simultanée de programmes batch, éventuellement sur des machines différentes, pour garantir que la capacité de la base de données est pleinement exploitée.

Ce traitement par lots accepte le paramètre Batch Process Instance ID (Identificateur de l'instance du traitement par lots). L'identificateur d'instance est un Identificateur unique qui permet une exécution en parallèle efficace de plusieurs instances d'un même traitement par lots. Lorsqu'aucun identificateur d'instance n'est indiqué, une seule instance du traitement par lots peut être exécutée.

Diffusion en flux de plusieurs instance de Déterminer l'éligibilité de la distribution de produits : Il est possible d'exécuter en parallèle plusieurs instances du traitement par lots Déterminer l'éligibilité de la distribution de produits en attribuant à chaque instance un Batch Process Instance ID (Identificateur de l'instance du traitement par lots) différent. Pour démarrer la diffusion en flux d'une instance spécifique du traitement par lots Déterminer l'éligibilité de la distribution de produits, vous devez lier le traitement par lots Déterminer le flux d'éligibilité de la distribution de produits (ou plusieurs traitements par lots de diffusion) à l'instance spécifique du traitement à l'aide du paramètre Batch Process Instance ID (Identificateur de l'instance du traitement par lots). Par exemple si Batch Process Instance ID (Identificateur de l'instance du traitement par lots) est "eligibility_determination_1" pour une instance du traitement par lots Déterminer l'éligibilité de la distribution de produits, vous devez également définir le paramètre Batch Process Instance ID (Identificateur de l'instance du traitement par lots) pour le traitement par lots Déterminer le flux d'éligibilité de la distribution de produits (ou plusieurs traitement par lots de diffusion en flux) sur "eligibility_determination_1". N'importe quel nombre de traitements par lots Déterminer le flux d'éligibilité de la distribution de produits peut être associé à une même instance du traitement par lots Déterminer l'éligibilité de la distribution de produits.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.DetermineProductDeliveryEligibilityStream.process`.

ReassessOutstandingCases : Ce traitement par lots lit tous les enregistrements de l'entité Planification de réévaluation et procède à la réévaluation de chaque dossier. L'entité Planification de réévaluation contient les entrées des dossiers en attente de réévaluation et ne doit être lue qu'en mode de traitement. Les dossiers peuvent également être réévalués en mode différé mais dans ce mode, la planification de la réévaluation n'est pas utilisée.

Ce traitement par lots n'accepte aucun paramètre.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.ReassessOutstandingCases.reassess`.

FullPropagationToRuleObjects : Ce traitement par lots effectue une propagation complète des données de la base de données de la table de taux aux objets règles.

Ce processus devrait être exécuté chaque fois qu'il y a des raisons de penser que les objets règles CER stockés ne reflètent plus avec exactitude les données de la base de données de la table de taux. Des différences peuvent apparaître lorsqu'une propagation incrémentielle de données de base de données a été ignorée, par exemple si des tables de taux sont mises à jour en dehors de l'application.

Remarque : En règle générale, vous ne devriez pas avoir besoin d'exécuter ce programme, sauf si vous avez apporté de nombreux changements à des tables de taux sans passer par les API de l'application. Si vous n'avez apporté qu'un petit

nombre de changements, vous pouvez utiliser l'action Appliquer les changements dans l'application d'administration en ligne.

Un résumé des différences trouvées est écrit dans les journaux d'application standard. Pour obtenir une consignation détaillée de tous les traitements, le niveau de journalisation doit être défini sur prolix ou ultra-prolix.

Ce traitement par lots n'accepte aucun paramètre.

Classe et méthode du traitement par lots : La classe et la méthode pour ce traitement par lots sont `curam.core.sl.infrastructure.propagator.interfaces.FullPropagationToRuleObjects.execute`.

RateCreateInitialRuleObjects : Ce traitement par lots crée les objets règles CER initiaux correspondant aux entrées de table de taux dans un environnement dans lequel les données de la table de taux ont été renseignées sans passer par les API de l'application.

Consultez le manuel Propagation de données non Cúram pour Cúram Express Rules pour plus d'informations.

Ce traitement par lots accepte le paramètre suivant :

- `rateTableType` (Type de table de taux) - table de taux à traiter (vide pour traiter toutes les tables de taux).

Classe et méthode du traitement par lots : La classe et la méthode pour ce traitement par lots sont `curam.core.sl.infrastructure.rate.interfaces.RateCreateInitialRuleObjects.propagateRateTable`.

CREOLEBulkCaseChunkReassessmentByProduct : Ce traitement par lots permet d'identifier et d'effectuer une réévaluation complète d'un grand nombre de dossiers CER "Actifs" pour un type de produit donné. Pour les dossiers dont la détermination change suite à cette réévaluation, la nouvelle détermination est stockée et l'ancienne rendue obsolète.

Important : Etant donné que ce processus entraîne la réévaluation de tous les dossiers du type spécifié, il peut générer de nombreuses réévaluations inutiles. Si nécessaire, créez un traitement par lots permettant d'identifier plus précisément les dossiers nécessitant une réévaluation, en particulier lorsque les dossiers sont répartis entre plusieurs produits. Pour une explication complète sur l'écriture de traitements par lots appropriés, consultez le manuel *Inside Cúram Eligibility and Entitlement Using Cúram Express Rules*.

Vous pouvez exécuter ce programme lorsque vous avez apporté des changements au système qui affectent un grand nombre de dossiers CER et que vous souhaitez forcer le système à réévaluer les dossiers par produit (plutôt que laisser la suite par lots du gestionnaire de dépendances déterminer l'ordre dans lequel les dossiers doivent être réévalués¹²).

Les types de changement susceptibles d'affecter un grand nombre de dossiers CER sont les suivants :

12. Consultez le manuel *Inside Cúram Eligibility and Entitlement Using Cúram Express Rules* pour obtenir des informations permettant de déterminer si vous devez utiliser ce traitement par lots en plus de la suite par lots du gestionnaire de dépendances ou à la place de celle-ci.

- publication des changements d'un jeu de règles CER ;
- publication des changements de la configuration d'un produit CER ;
- publication des changements de la configuration de données CER ; et
- application des changements de la table de taux.

Ce traitement par lots accepte les paramètres suivants :

- Product ID (Identificateur de produit) - identificateur unique utilisé afin d'exécuter ce processus pour les dossiers d'un produit spécifique
Lorsqu'aucun identificateur de produit n'est spécifié, tous les dossiers sont traités.
- Batch Process Instance ID (Identificateur de l'instance du traitement par lots) - identificateur unique utilisé pour permettre l'exécution efficace de plusieurs instances d'un même traitement par lots.
Par exemple, ce processus peut s'exécuter pour plusieurs produits. Lorsqu'aucun identificateur d'instance n'est indiqué, une seule instance du traitement par lots peut être exécutée.

Classe et méthode du traitement par lots : La classe et la méthode pour ce traitement par lots sont `curam.core.sl.infrastructure.assessment.intf.CREOLEBulkCaseChunkReassessmentByProduct.process`.

CREOLEBulkCaseChunkReassessmentStream : Ce traitement par lots prend en charge la diffusion en flux du processus Réévaluation de segment de dossier par lot CREOLE par produit. Il ne peut être exécuté qu'en association avec le processus Réévaluation de segment de dossier par lot CREOLE par produit. La diffusion en flux des traitements par lots permet l'exécution simultanée de programmes batch, éventuellement sur des machines différentes, pour garantir que la capacité de la base de données est pleinement exploitée.

Ce traitement par lots accepte le paramètre Batch Process Instance ID (Identificateur de l'instance du traitement par lots). L'identificateur d'instance est un identificateur unique qui permet une exécution en parallèle efficace de plusieurs instances d'un même traitement par lots. Lorsqu'aucun identificateur d'instance n'est indiqué, une seule instance du traitement par lots peut être exécutée.

Diffusion en flux de plusieurs instances de Réévaluation de segment de dossier par lot CREOLE par produit : Il est possible d'exécuter en parallèle plusieurs instances du traitement par lots Réévaluation de segment de dossier par lot CREOLE par produit en attribuant à chaque instance un Batch Process Instance ID (Identificateur de l'instance du traitement par lots) différent. Pour démarrer la diffusion en flux d'une instance spécifique du traitement par lots Réévaluation de segment de dossier par lot CREOLE par produit, vous devez lier le traitement par lots CREOLE Bulk Case Chunk Reassessment Stream (Diffusion en flux de la réévaluation de segment de dossier par lot CREOLE) (ou plusieurs traitements par lots de diffusion) à l'instance spécifique du traitement à l'aide du paramètre Batch Process Instance ID (Identificateur de l'instance du traitement par lots). Par exemple si Batch Process Instance ID (Identificateur de l'instance du traitement par lots) est "batch_reassessment_1" pour une instance du traitement par lots Réévaluation de segment de dossier par lot CREOLE par produit, vous devez également définir le paramètre Batch Process Instance ID (Identificateur de l'instance du traitement par lots) pour le traitement par lots CREOLE Bulk Case Chunk Reassessment Stream (Diffusion en flux de la réévaluation de segment de

dossier par lot CREOLE) (ou plusieurs traitement par lots de diffusion en flux) sur "batch_reassessment_1". N'importe quel nombre de traitements par lots CREOLE Bulk Case Chunk Reassessment Stream CREOLE Bulk Case Chunk Reassessment Stream (Diffusion en flux de la réévaluation de segment de dossier par lot CREOLE) peut être lié à une même instance du traitement par lots Réévaluation de segment de dossier par lot CREOLE par produit.

Classe et méthode du traitement par lots : La classe et la méthode pour ce traitement par lots sont `curam.core.sl.infrastructure.assessment.intf.CREOLEBulkCaseChunkReassessmentStream.process`.

ApplyProductReassessmentStrategy : Vérifie chaque dossier de distribution d'un produit pour déterminer si la prise en charge de la réévaluation du dossier (par exemple, prise en charge de la réévaluation d'un dossier clos) a changé en raison du changement de la stratégie de réévaluation du produit.

Pour chaque dossier de distribution du produit :

- Si le dossier n'était pas réévaluable dans le cadre de l'ancienne stratégie, mais qu'il le devient avec la nouvelle stratégie, une évaluation du dossier est réalisée pour générer les enregistrements de dépendance destinés au résultat de détermination du dossier.
- Si le dossier était réévaluable dans le cadre de l'ancienne stratégie, mais qu'il ne l'est plus avec la nouvelle, les enregistrements de dépendance destinés au résultat de détermination sont supprimés.
- Sinon, aucune action n'est réalisée sur le dossier.

Ce traitement par lots accepte le paramètre suivant :

- Product ID (Identificateur de produit) - identificateur unique utilisé afin d'exécuter ce processus pour les dossiers d'un produit spécifique

Voir le manuel *Inside Cúram Eligibility and Entitlement Using Cúram Express Rules*.

Classe et méthode du traitement par lots : La classe et la méthode du traitement par lots de fragmentation sont `curam.core.sl.infrastructure.assessment.intf.ApplyProductReassessmentStrategy.process me`.

La classe et la méthode du traitement par lots de flux sont `curam.core.sl.infrastructure.assessment.intf.ApplyProductReassessmentStrategyStream.process`.

Gestionnaire de dépendances

Pour des informations complètes sur les traitements par lots inclus avec le gestionnaire de dépendances, consultez le manuel *Cúram Express Rules - Manuel de référence*.

SubmitPrecedentChangeSet : Ce traitement par lots constitue le point de départ de la suite par lots du gestionnaire de dépendances et contient un processus de diffusion en flux unique léger qui soumet l'ensemble de modifications précédent par lot actuellement ouvert et crée un ensemble de modifications précédent par lot qui sera utilisé pour capturer toutes les modifications précédentes identifiées par la suite et en attente de traitement par lots.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.dependency.interfaces.SubmitPrecedentChangeSet.process`.

PerformBatchRecalculationsFromPrecedentChangeSet : Ce traitement par lots est le processus de diffusion multiple qui identifie les dépendances potentiellement affectées par les changements de l'ensemble de modifications précédent en cours de validation, puis les recalcule.

Ce traitement par lots doit être exécuté une fois pour chaque type dépendant enregistré auprès du gestionnaire de dépendances.

Classe et méthode du traitement par lots : La classe et la méthode du traitement par lots segmenteur sont `curam.dependency.interfaces.PerformBatchRecalculationsFromPrecedentChangeSet.process`.

La classe et la méthode du traitement par lots de diffusion sont `curam.dependency.interfaces.PerformBatchRecalculationsFromPrecedentChangeSetStream.process`.

CompletePrecedentChangeSet : Noeud final de la suite par lots du gestionnaire de dépendances, qui contient un processus léger de diffusion en flux unique qui complète l'ensemble de modifications précédent par lot en cours de validation.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.dependency.interfaces.CompletePrecedentChangeSet.process`.

Outils de traitement par lots du gestionnaire de dépendance : Les outils sont mis à disposition sous forme de scripts Ant pour vous aider à exécuter la suite de traitements par lots du gestionnaire de dépendance. L'utilisation des outils garantit que les travaux par lots sont exécutés dans l'ordre approprié et permettent à l'utilisateur de définir une propriété pour gérer les performances de l'exécution par lots. L'outil se compose de deux parties : la première génère un script ant propre au système sur lequel il s'exécute et la deuxième implique l'appel de ce script ant pour exécuter la suite de traitements par lots.

Conditions préalables

Les variables d'environnement suivantes doivent être définies pour un fonctionnement normal de Cúram :

- `SERVER_DIR`
- `CURAMSDEJ`

Les pilotes de base de données se trouvent dans le répertoire `SERVER_DIR/drivers`. L'outil extrait les détails de connexion de la base de données depuis le fichier `Bootstrap.properties` normal.

Génération du script Ant spécifique

Vue d'ensemble :

Au cours de cette étape, l'outil examine les types dépendants pris en charge sur le système, puis construit un script ant pour que chacun de ces types dépendants soit traité au cours de l'exécution de la suite de traitements par lots. De plus, certaines options de configuration peuvent être spécifiées dans le fichier `conf/DependencyManagerBatch.properties` comme suit :

- `batch.default.threads=3` - Spécifie le nombre par défaut d'unités d'exécution dans le script ant généré (si cette propriété a une valeur de n, un fragmenteur et n-1 flux sont créés pour le traitement de chaque type dépendant).

De plus, pour chaque type dépendant, vous pouvez remplacer la valeur par défaut en spécifiant une propriété pour ce type comme suit :

- `batch.CADETERRES.recalculation.threads=5` - Cet exemple remplace le nombre d'unités d'exécution utilisées pour traiter le type dépendant de réévaluation des décisions concernant les dossiers.

Exécution de l'étape :

Pour exécuter l'étape, exécutez simplement la commande 'ant' depuis le répertoire `DependencyManagerBatch`.

Sortie :

L'étape génère un fichier, `DependencyManagerBatch.xml`, dans le répertoire courant.

Appel du script pour exécuter la suite de traitements par lots

Vue d'ensemble :

La deuxième étape de l'outil utilise la sortie générée à partir de la première. Le script effectue les étapes suivantes :

- Il vérifie si des dépendances doivent être traitées ; s'il n'en a pas, il s'arrête.
- Si des dépendances doivent être traitées, soumettez l'ensemble de modifications précédent du lot courant.
- Pour chaque type dépendant, traitez les recalculs avec le nombre spécifié d'unités d'exécution.
- Enfin, terminez l'ensemble de modifications précédent.

Exécution de l'étape :

Pour exécuter cette étape, exécutez simplement la commande 'ant -f `DependencyManagerBatchBase.xml`'.

Sortie :

Si aucune dépendance ne requiert de traitement, un message l'indique et le script s'arrête. Sinon, un fichier journal individuel est écrit dans le dossier des journaux pour chaque unité d'exécution/processus par lots exécuté.

Financial

GenerateInstructionLineItems : Ce traitement par lots permet de créer des lignes d'instruction pour tous les composants Financial devant être traités. Les paramètres d'entrée de ce processus identifient les composants Financial devant être traités.

Ce traitement par lots effectue les tâches suivantes :

- Lorsqu'une date est spécifiée pour "Date de début" et "Date de fin", tous les composants Financial dont la date de traitement est antérieure à cette date sont récupérés. Lorsqu'aucune date n'est indiquée, la date système en cours est utilisée.

- De même, lorsqu'aucune méthode de livraison n'est spécifiée, toutes les méthodes de livraison sont traitées.
- Ce processus utilise la "Date de fin" pour vérifier le calendrier Financial et déterminer si des jours supplémentaires doivent être inclus dans le traitement en cours des composants Financial.
- Ce processus extrait tous les composants Financial devant être traités et les regroupe par dossier.
- Ce processus réévalue chaque dossier afin de déterminer si un changement de circonstances peut avoir modifié l'éligibilité à Financial.
- Pour chaque composant Financial restant, le montant et la période de couverture sont calculés et une ligne d'instruction est créée.
- Ce processus avance la date de traitement du composant Financial et lorsque après cette opération le composant Financial atteint la fin de sa durée de vie, le composant financier est arrivé à expiration.

Ce traitement par lots accepte les paramètres suivants :

- Batch Process Instance ID (Identificateur de l'instance du traitement par lots) - identificateur unique utilisé pour permettre l'exécution simultanée efficace de plusieurs instances d'un même traitement par lots.
Par exemple, ce processus peut s'exécuter pour plusieurs méthodes de livraison. Lorsqu'aucun identificateur d'instance n'est indiqué, une seule instance du traitement par lots peut être exécutée.
- Date de début - date de début de l'identification des composants Financial à inclure dans le traitement
- Date de fin - date de fin de l'identification des composants Financial à inclure dans le traitement
- Méthode de livraison - méthode de livraison à utiliser pour identifier les composants Financial à inclure dans le traitement

Classe et méthode du traitement par lots : La classe et la méthode pour ce traitement par lots sont `curam.core`
`.intf.GenerateInstructionLineItems.processAllFinancialComponentsDue`.

Remarque : Ce traitement par lots doit être exécuté jusqu'à son terme avant toute exécution de `GenerateInstruments`.

GenerateInstructionLineItemsStream : Ce traitement par lots permet de prendre en charge la diffusion en flux du processus Générer des lignes d'instruction. Il ne peut être exécuté qu'en association avec le processus Générer des lignes d'instruction. La diffusion en flux des traitements par lots permet l'exécution simultanée de programmes batch, éventuellement sur des machines différentes, pour garantir que la capacité de la base de données est pleinement exploitée.

Ce traitement par lots accepte le paramètre Batch Process Instance ID (Identificateur de l'instance du traitement par lots). L'identificateur d'instance est un identificateur unique qui permet une exécution en parallèle efficace de plusieurs instances d'un même traitement par lots. Lorsqu'aucun identificateur d'instance n'est indiqué, une seule instance du traitement par lots peut être exécutée.

Diffusion en flux de plusieurs instances de Générer des lignes d'instruction : Il est possible d'exécuter en parallèle plusieurs instances du traitement par lots Générer des lignes d'instruction en attribuant à chaque instance un Batch Process Instance ID (Identificateur de l'instance du traitement par lots)

différent. Pour démarrer la diffusion en flux d'une instance spécifique du traitement par lots Générer des lignes d'instruction, vous devez lier le traitement par lots Générer un flux de lignes d'instruction (ou plusieurs traitements par lots de diffusion) à l'instance spécifique du traitement à l'aide du paramètre Batch Process Instance ID (Identificateur de l'instance du traitement par lots). Par exemple si Batch Process Instance ID (Identificateur de l'instance du traitement par lots) est "generate_instruction_line_items_1" pour une instance du traitement par lots Générer des lignes d'instruction, vous devez également définir le paramètre Batch Process Instance ID (Identificateur de l'instance du traitement par lots) pour le traitement par lots Générer un flux de lignes d'instruction (ou plusieurs traitement par lots de diffusion en flux) sur "generate_instruction_line_items_1". N'importe quel nombre de traitements par lots Générer des lignes d'instruction peut être associé à une même instance du traitement par lots Générer un flux de lignes d'instruction.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont
`curam.core.intf.GenerateInstructionLineItemsStream.process.`

GenerateInstruments : Ce traitement par lots effectue les tâches suivantes :

- Identification des lignes d'instruction à traiter
- Cumul des lignes d'instructions dans Instructions
- Création d'un instrument (le cas échéant)

Ce traitement par lots accepte le paramètre Batch Process Instance ID (Identificateur de l'instance du traitement par lots). L'identificateur d'instance est un identificateur unique qui permet une exécution en parallèle efficace de plusieurs instances d'un même traitement par lots. Lorsqu'aucun identificateur d'instance n'est indiqué, une seule instance du traitement par lots peut être exécutée.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont
`curam.core.intf.GenerateInstruments.processInstructionLineItemsDue.`

Remarque : Ce processus doit être exécuté entre les processus `GenerateInstructionLineItems` et `IssueConcernPayments`.

GenerateInstrumentsStream : Ce traitement par lots permet de prendre en charge la diffusion en flux du processus Générer des instruments. Il ne peut être exécuté qu'en association avec le processus Générer des instruments. La diffusion en flux des traitements par lots permet l'exécution simultanée de programmes batch, éventuellement sur des machines différentes, pour garantir que la capacité de la base de données est pleinement exploitée.

Ce traitement par lots accepte le paramètre Batch Process Instance ID (Identificateur de l'instance du traitement par lots). L'identificateur d'instance est un identificateur unique qui permet une exécution en parallèle efficace de plusieurs instances d'un même traitement par lots. Lorsqu'aucun identificateur d'instance n'est indiqué, une seule instance du traitement par lots peut être exécutée.

Diffusion de plusieurs instances de Générer des instruments : Il est possible d'exécuter en parallèle plusieurs instances du traitement par lots Générer des

instruments en attribuant à chaque instance un Batch Process Instance ID (Identificateur de l'instance du traitement par lots) différent. Pour démarrer la diffusion en flux d'une instance spécifique du traitement par lots Générer des instruments, vous devez lier le traitement par lots Générer un flux d'instruments (ou plusieurs traitements par lots de diffusion) à l'instance spécifique du traitement à l'aide du paramètre Batch Process Instance ID (Identificateur de l'instance du traitement par lots). Par exemple si Batch Process Instance ID (Identificateur de l'instance du traitement par lots) est "generate_instruments_1" pour une instance du traitement par lots Générer des instruments, vous devez également définir le paramètre Batch Process Instance ID (Identificateur de l'instance du traitement par lots) pour le traitement par lots Générer un flux d'instruments (ou plusieurs traitement par lots de diffusion en flux) sur "generate_instruments_1". N'importe quel nombre de traitements par lots Générer un flux d'instruments peut être associé à une même instance du traitement par lots Générer des instruments.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.GenerateInstrumentsStream.process`.

IssueConcernPayments : Ce traitement par lots effectue les tâches suivantes :

- Identification des entreprises des services collectifs, fournisseurs de services ou fournisseurs de produits pouvant devoir être payées selon les détails fournis en entrée
- Extraction des instructions de paiement en attente pour chaque entreprise et cumul dans un instrument de paiement à destination de l'entreprise
- Mise à jour de la date du prochain paiement de l'entreprise

Ce traitement par lots accepte les paramètres suivants :

- Date de début - début de la plage de dates pour laquelle les paiements de l'entreprise doivent être traités
- Date de fin - fin de la plage de dates pour laquelle les paiements de l'entreprise doivent être traités
- Mode de règlement - méthode de livraison à utiliser pour l'identification des composants Financial à inclure dans le traitement
Lorsque ce paramètre n'est pas spécifié, tous les modes de règlement sont traités.
- Concern Type (Type d'entreprise) - service collectif, fournisseur de services ou fournisseur de produits. Le traitement par lots doit être exécuté pour un seul type d'entreprise
Lorsque ce paramètre n'est pas spécifié, le traitement par lots est exécuté pour les trois types.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.IssueConcernPayments.issueConcernTypePayment`.

Remarque : Ce processus doit être exécuté entre les processus `GenerateInstruments` et `GeneratePayslips`.

GeneratePayslips : Ce traitement par lots effectue les tâches suivantes :

- Recherche des bulletins de salaire qui n'ont pas encore été émis
- Récupération des données nécessaires à la génération des informations dans un format lisible
- Mise à jour du statut du bulletin de salaire sur Emis

Ce traitement par lots est un exemple d'implémentation qui démontre que Cúram permet de générer des bulletins de salaire. Les données de sortie (qui génèrent actuellement un fichier de sortie par type de destinataire) ne constituent pas l'implémentation définitive de ce traitement par lots. Par exemple, une implémentation personnalisée peut fournir un fichier de sortie par destinataire.

Ce traitement par lots n'accepte aucun paramètre.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.GeneratePayslips.generateNewPayslips`.

Remarque : Ce processus doit être exécuté au terme du processus `IssueConcernPayments`.

LoadServiceSupplierReturns : Les retours du fournisseur de services peuvent être stockés dans un fichier d'entrée. Ce traitement par lots charge les retours du fournisseur de services à partir du fichier d'entrée.

Ce traitement par lots accepte le paramètre Nom de fichier . Répertoire du fichier d'entrée.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.LoadServiceSupplierReturns.loadSupplierReturns`.

ElectronicEmployerReturn : Ce traitement par lots transfère les détails de retour de contribution reçus d'un employeur au format électronique.

Il accepte les paramètres suivants :

- Chemin d'accès au fichier - répertoire du fichier électronique
- Nom de fichier - nom complet du fichier électronique, comprenant l'extension, qui contient les détails du retour de contribution

Lorsqu'il est exécuté, ce traitement par lots recherche le Nom de fichier spécifié dans le Chemin d'accès au fichier indiqué.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.ElectronicEmployerReturn.capture`.

ConsolidationProcess : Ce traitement par lots consolide les éléments de lignes d'assurance pour l'entreprise d'une personne. Pour ce faire, le traitement par lots procède comme suit :

- Localisation de tous les éléments de lignes d'assurance de l'entreprise d'une personne
- Regroupement de tous les éléments de lignes présentant la même période de rendement d'assurance, le même type d'assurance et le même type de produit
- Calcul du nombre total d'unités d'assurance pour tous les éléments de lignes d'assurance regroupés

Ce traitement par lots n'accepte aucun paramètre.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.ConsolidationProcess.batchConsolidation`.

ProcessPaymentInstrumentTypes : Ce traitement par lots effectue les tâches suivantes :

- Extraction des paiements qui n'ont pas encore été émis
- Rassemblement des données nécessaires pour le paiement non émis et envoi dans un fichier utilisé comme interface avec un système externe
- Mise à jour du statut du paiement pour indiquer qu'il a été émis

Ce traitement par lots accepte le paramètre Méthode de livraison . Il s'agit de la méthode de livraison à utiliser pour l'identification des paiements non émis. Lorsque ce paramètre n'est pas spécifié, tous les modes de règlement sont traités.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont
`curam.core.intf.ProcessPaymentInstrumentTypes.processPmtInstrumentType.`

Remarque : Ce traitement par lots est un exemple d'implémentation qui montre comment l'application peut s'interfacer avec les systèmes de traitement des paiements. Il peut être exécuté uniquement pour les paiements non émis. Lorsque la méthode de livraison est spécifiée, tous les paiements non émis pour cette méthode sont traités ; si la méthode de livraison n'est pas spécifiée, tous les paiements non émis sont traités.

ExpirePayments : Ce traitement par lots effectue les tâches suivantes :

- Extraction de tous les instruments de paiement correspondant à la méthode de livraison spécifiée en entrée et dont la date d'effet est antérieure ou correspond à la date du jour moins la période d'expiration
- Définition du statut sur Expiré pour chacun de ces instruments de paiement et génération d'un enregistrement dans le fichier journal
Ce fichier journal peut être utilisé pour indiquer à l'institution financière appropriée que les paiements ne doivent pas être encaissés.

Ce traitement par lots accepte les paramètres suivants :

- Période d'expiration - nombre de jours de la période d'expiration
- Méthode de livraison - méthode de livraison à utiliser pour identifier les paiements expirés

Lorsque ce paramètre n'est pas spécifié, tous les modes de règlement sont traités.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont
`curam.core.intf.ExpirePayments.expirePaymentInstrument.`

PaymentReconciliation. : Ce traitement par lots synchronise un compte en comparant les montants dus aux montants réglés. Toutes les différences détectées lors de cette comparaison sont enregistrées dans un rapport.

Ce traitement par lots accepte les paramètres suivants :

- Chemin d'accès au fichier - répertoire du fichier de sortie
- Nom de fichier - nom complet du fichier de sortie, extension comprise, qui contient le détail des paiements

Lorsqu'il est exécuté, ce traitement par lots recherche le Nom de fichier spécifié dans le Chemin d'accès au fichier indiqué.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.PaymentReconciliation.reconcilePayments`.

ReconcileCaseAccount : Ce traitement par lots effectue les tâches suivantes :

- Synchronisation de tous les dossiers de surendettement pour lesquels un sous-paiement a été appliqué
- Synchronisation de tous les dossiers de surendettement pour lesquels un trop-perçu a été appliqué

Ce traitement par lots n'accepte aucun paramètre.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.ReconcileCaseAccount.reconcileCaseAccount`.

Flux de travaux

ScanTaskDeadlines : Le traitement effectué par ce traitement par lots peut aussi être effectué par les traitements par lots `ProcessTaskDeadlines` et `ProcessTaskDeadlinesStream`. Ce traitement par lots effectue les tâches suivantes :

- Recherche des tâches en retard, c'est-à-dire dont la date et l'heure d'échéance sont postérieures à la date et à l'heure en cours
- Si une fonction de gestionnaire d'échéance a été spécifiée dans la définition de processus de flux de travaux associée pour l'échéance, ce gestionnaire est appelé.
- Sinon, si l'indicateur d'activité terminée n'a pas la valeur true, le moteur de flux de travaux termine l'activité associée à l'échéance et fait progresser le flux de travaux.
- L'échéance qui a été traitée est supprimée pour qu'elle ne soit pas traitée à nouveau.
- Les données associées à l'attribut d'objet de données de flux de travaux `Context_Deadline` sont conservées.
- Si une tâche est associée à l'échéance qui a expiré, un enregistrement d'historique des tâches détaillant ce fait est écrit.

Ce traitement par lots n'accepte aucun paramètre.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.ScanTaskDeadlines.scanDeadlines`.

Remarque : Lors de l'exécution de ce traitement par lots à partir de la ligne de commande, le paramètre `batch.username` doit être fourni et doit faire référence à un utilisateur valide doté des privilèges appropriés.

ProcessTaskDeadlines : Ce traitement par lots effectue les tâches suivantes :

- Recherche des tâches en retard, c'est-à-dire dont la date et l'heure d'échéance sont postérieures à la date et à l'heure en cours
- Si une fonction de gestionnaire d'échéance a été spécifiée dans la définition de processus de flux de travaux associée pour l'échéance, ce gestionnaire est appelé.
- Sinon, si l'indicateur d'activité terminée n'a pas la valeur true, le moteur de flux de travaux termine l'activité associée à l'échéance et fait progresser le flux de travaux.

- L'échéance qui a été traitée est supprimée pour qu'elle ne soit pas traitée à nouveau.
- Les données associées à l'attribut d'objet de données de flux de travaux *Context_Deadline* sont conservées.
- Si une tâche est associée à l'échéance qui a expiré, un enregistrement d'historique des tâches détaillant ce fait est écrit.

Ce traitement par lots n'accepte aucun paramètre.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.ProcessTaskDeadlines.process`.

Remarque : Lors de l'exécution de ce traitement par lots à partir de la ligne de commande, le paramètre `batch.username` doit être fourni et doit faire référence à un utilisateur valide doté des privilèges appropriés.

ProcessTaskDeadlinesStream : Ce traitement par lots permet de prendre en charge la diffusion en flux du processus `ProcessTaskDeadlines`.

Il ne peut être exécuté qu'en conjonction avec le processus `ProcessTaskDeadlines`. La diffusion en flux des traitements par lots permet l'exécution simultanée de programmes batch, éventuellement sur des machines différentes, pour garantir que la capacité de la base de données est pleinement exploitée.

Ce traitement par lots accepte le paramètre `Batch Process Instance ID` (Identificateur de l'instance du traitement par lots). L'identificateur d'instance est un identificateur unique qui permet une exécution en parallèle efficace de plusieurs instances d'un même traitement par lots. Lorsqu'aucun identificateur d'instance n'est indiqué, une seule instance du traitement par lots peut être exécutée.

Diffusion de plusieurs instances de ProcessTaskDeadlines. : Il est possible d'exécuter en parallèle plusieurs instances du traitement par lots `ProcessTaskDeadlines` en attribuant à chaque instance un `Batch Process Instance ID` (Identificateur de l'instance du traitement par lots) différent. Pour démarrer la diffusion en flux d'une instance spécifique du traitement par lots `ProcessTaskDeadlines`, vous devez lier le traitement par lots `ProcessTaskDeadlinesStream` (ou plusieurs traitements par lots de diffusion) à l'instance spécifique du traitement à l'aide du paramètre `Batch Process Instance ID` (Identificateur de l'instance du traitement par lots). Par exemple si `Batch Process Instance ID` (Identificateur de l'instance du traitement par lots) est "process_deadlines_1" pour une instance du traitement par lots `ProcessTaskDeadlinesStream`, vous devez également définir le paramètre `Batch Process Instance ID` (Identificateur de l'instance du traitement par lots) pour le traitement par lots `ProcessTaskDeadlinesStream` (ou plusieurs traitements par lots de diffusion en flux) sur "process_task_1". N'importe quel nombre de traitements par lots `ProcessTaskDeadlinesStream` peut être associé à une même instance du traitement par lots `ProcessTaskDeadlines`.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.ProcessTaskDeadlinesStream.process`.

RestartTask : Ce traitement par lots effectue les tâches suivantes :

- Recherche des tâches dont le statut est `Différé` et dont la date de redémarrage a dépassé la date en cours.

- Pour chacune de ces tâches, il définit le statut sur Ouvert. Il définit également la date et l'heure de redémarrage à la date et l'heure zéro.

Ce traitement par lots n'accepte aucun paramètre.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.RestartTask.restart`.

ExpireWaitListEntry : Lorsqu'un client est ajouté à une liste d'attente pour une ressource, une date d'expiration peut être spécifiée. La date d'expiration peut être saisie par l'utilisateur ou créée par le système. Ce traitement par lots permet de faire expirer une entrée de la liste d'attente si le client n'est ni alloué à une ressource ni supprimé de la liste avant la date d'expiration. Le processus fait expirer l'entrée de la liste d'attente si la date d'expiration de la liste d'attente est égale ou antérieure à la date du traitement par lots, et si l'entrée de la liste d'attente est à l'état 'Ouvert'. Si la liste d'attente nécessite une renumérotation, alors le système renumérote la liste d'attente en réduisant de 1 la position de toutes les entrées de la liste d'attente à l'état Ouvert et dont les positions sont supérieures ou égales à celle de l'entrée expirée. Une fois l'entrée de la liste d'attente expirée, le processus génère un événement de flux de travaux 'WAITLIST.WAITLISTENTRYEXPIRED'. Ce traitement par lots est censé être exécuté quotidiennement.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.ExpireWaitListEntry.expireWaitListEntry`.

WaitListReview : Ce traitement par lots permet de générer un événement de flux de travaux 'WAITLIST.WAITLISTENTRYSELECTEDFORREVIEW' qui crée des tâches de révision de listes d'attente pour toutes les entrées de listes d'attente éligibles concernées. Les entrées de liste d'attente éligibles sont toutes les entrées de liste d'attente dont la date de révision est égale ou antérieure à la date du traitement par lots. Si aucune date de révision n'est définie, cette date est calculée en soustrayant la période de rappel de révision du statut configurée à la date d'expiration. La période de rappel de révision du statut est configurée par l'administrateur système à l'aide de la propriété `'curam.waitlist.statusreviewreminderperiod'`. Par exemple, si la date de révision de la liste d'attente est le 30 et que la période de rappel de révision du statut est définie sur 5, alors le traitement par lots 'WaitListReview' lève l'événement de flux de travaux pour générer la tâche de révision de la liste d'attente le 25. Pour que la génération des tâches de révision réussisse, la propriété `'curam.batchlauncher.dbtojms.enabled'` doit être définie sur true et les valeurs appropriées doivent être définies pour les propriétés `'curam.batchlauncher.dbtojms.notification.host'` et `'curam.batchlauncher.dbtojms.notification.port'`. Ces propriétés sont configurées par l'administrateur système. Ce traitement par lots doit s'exécuter quotidiennement.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.WaitListReview.processWaitListEntriesDueForReview`.

Interfaces d'entrée

LoadPaymentsReceived

Ce traitement par lots effectue les tâches suivantes :

- Validation des données pour chaque enregistrement du fichier d'entrée, PaymentReceivedFile.txt
- Chargement de l'enregistrement de paiement reçu sur un compte d'entreprise lorsqu'une entreprise est identifiée. Sinon, l'enregistrement est chargé sur un compte d'attente
- Gestion d'un ensemble de totaux de contrôle et comparaison à l'enregistrement de total de contrôle à la fin du fichier d'entrée
- Annulation de tout le traitement si les totaux en cours ne correspondent pas aux données du fichier d'entrée

Ce traitement par lots accepte les paramètres suivants :

- Chemin d'accès au fichier - répertoire du fichier d'entrée
- Nom de fichier - nom complet du fichier d'entrée, extension comprise, qui contient le détail des paiements reçus.

Lorsqu'il est exécuté, ce traitement par lots recherche le Nom de fichier spécifié dans le Chemin d'accès au fichier indiqué.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.LoadPaymentsReceived.loadPaymentReceivedFile`.

Interfaces de sortie

Export de calendrier

Ce traitement par lots effectue les tâches suivantes :

- Utilisation des paramètres pour collecter les activités de calendrier
- Exportation des activités de l'application vers un fichier de sortie au format vCalendar

Ce fichier de sortie peut ensuite être importé dans un calendrier externe.

Paramètres requis : Pour que ce traitement par lots s'exécute, vous devez entrer le paramètre Nom d'utilisateur, le paramètre Organization ID (Identificateur d'organisation) ou les paramètres Date de début et Date de fin.

Ce traitement par lots accepte les paramètres suivants :

- Chemin du fichier d'exportation : répertoire du fichier de sortie
- Nom d'utilisateur : nom de l'utilisateur dont les activités sont exportées
Si ce paramètre est défini, le traitement par lots n'exporte que les activités de l'utilisateur spécifié.
- Organization ID (Identificateur d'organisation) : identificateur unique de l'organisation dont les activités sont exportées
Si ce paramètre est défini, le traitement par lots exporte les activités de l'organisation.
- Date de début : date de début des activités à exporter dans un fichier de sortie
- Date de fin : date de fin des activités à exporter dans un fichier de sortie
Pour que le traitement par lots exporte les activités dans une plage spécifiée, les paramètres Date de début et Date de fin doivent être définis.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont `curam.core.intf.CalendarExport.exportActivityDetails`.

Générer l'interface de grand livre

Regroupe les transactions financières pour une date ou une plage de dates spécifiée, et les exporte de l'application dans un fichier de sortie. Le fichier de sortie contient les détails des lignes d'instruction pour la date ou plage de dates spécifiée. Ce fichier de sortie peut ensuite être importé dans le grand livre.

Ce traitement par lots accepte les paramètres suivants :

- Date de début - début de la plage de dates des transactions financières à exporter dans un fichier de sortie
Si la date de début est nulle, c'est-à-dire non spécifiée par un utilisateur, une erreur est générée et le traitement par lots échoue.
- Date de fin - date de fin de la plage de dates des transactions financières à exporter dans un fichier de sortie
Si la date de fin est nulle, c'est-à-dire non spécifiée par un utilisateur, une erreur est générée et le traitement par lots échoue.
- Indicateur de recherche de date de création - indique si l'extraction est basée sur la date de création ou la date d'effet
Si cet indicateur est défini sur true, une recherche de plage de dates de création est effectuée ; sinon, une recherche de plage de dates d'effet est effectuée.

Classe et méthode du traitement par lots : La classe et la méthode de ce traitement par lots sont
`curam.core.intf.GeneralLedgerInterface.exportFinancialDetails`.

Paramètres de configuration Cúram

Introduction

Ce chapitre décrit les paramètres de configuration de

- Serveur XML Cúram
- Cúram Server Application

Les propriétés répertoriées dans ce chapitre correspondent à l'ensemble des propriétés de la suite Cúram Business Application Suite. Toutes les propriétés ne seront donc pas appropriées pour votre application.

Serveur XML Cúram

Le sous-répertoire XMLServer du répertoire d'installation contient le fichier de configuration `xmlserverconfig.xml`.

Consultez le manuel Cúram XML Infrastructure Guide pour plus d'informations sur les paramètres de configuration de ce fichier.

Cúram Server Application

Gestion des paramètres de configuration

Une fois la base de données créée, le fichier `Application.prx` est utilisé pour configurer les paramètres de configuration initiaux de la base de données. Une fois la base de données créée, tous les paramètres peuvent être administrés en ligne dans le cadre de l'administration de l'application (voir le manuel Guide de configuration du système Cúram).

Les paramètres de configuration peuvent être dynamiques ou statiques. Tous les changements apportés aux paramètres de configuration dynamiques sont appliqués automatiquement à l'application d'exécution après leur publication lors de l'exécution. Tous les changements apportés aux paramètres de configuration statiques nécessitent un réamorçage du système pour que ces nouveaux paramètres soient appliqués à l'application d'exécution.

Publication de paramètres de configuration : Les nouveaux paramètres de configuration ne seront appliqués que lors de leur publication ou du réamorçage du système. Pour publier ces changements, cliquez sur Publier sur la page Propriétés.

Modification des paramètres de configuration

Pour modifier les paramètres de configuration de Cúram Server Application, procédez comme suit.

1. Connectez-vous à l'application à l'aide de vos nom d'utilisateur et mot de passe¹³.
2. La page d'accueil s'affiche. Cliquez sur la section Configurations du système.
3. Cliquez sur le lien Données d'application - Administration des propriétés du panneau de raccourcis.
4. La page Administration des propriétés s'affiche et vous permet de filtrer la recherche par environnement local ou catégorie.
5. Sélectionnez le filtre approprié et cliquez sur le bouton Rechercher.
6. Cliquez sur Editer, sur la propriété requise.
7. La page d'édition de propriété de la propriété est affichée.
8. Apportez les changements nécessaires à la propriété.
9. Cliquez sur Sauvegarder pour enregistrer vos changements.
10. Le système enregistre les nouvelles informations.

Remarque : Pour plus d'informations sur la zone Administration des propriétés du client d'administration Cúram, voir le manuel Cúram - Guide d'administration du système.

Paramètres de configuration disponibles

Cúram Server Application est constituée des composants logiques suivants :

- Cúram Server Infrastructure
- Core Cúram Server Application
- Customized Cúram Server Application

Application.prx répertorie les paramètres utilisés par chacun de ces composants.

Paramètres de configuration de Cúram Server Infrastructure : Les paramètres de Cúram Server Infrastructure sont décrits dans Cúram Server - Guide de développement.

Paramètres de configuration de Cúram Core Server Application : Les paramètres de configuration de l'application sont regroupés comme suit :

Important : Pour les indicateurs "YES"/"NO", les valeurs exactes "YES" ou "NO" doivent être utilisées. Les autres capitalisations ou abréviations ("yes", "n", "true") ne fonctionneront *pas* correctement.

13. L'application fournit un nom d'administrateur (sysadmin) et un mot de passe (password) par défaut.

Customized Cúram Server Application : Veuillez prendre contact avec votre équipe de développement pour des informations détaillées sur les paramètres requis par les personnalisations de Core Cúram Server Application.

Remarques

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM. IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous accorde aucune licence pour ces brevets. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM Director of Licensing

IBM Corporation

North Castle Drive

Armonk, NY 10504-1785

U.S.A.

Pour le Canada, veuillez adresser votre courrier à :

IBM Director of Commercial Relations

IBM Canada Ltd

3600 Steeles Avenue East

Markham, Ontario

L3R 9Z7 Canada

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

Intellectual Property Licensing

Legal and Intellectual Property Law.

IBM Japan Ltd.

19-21, Nihonbashi-Hakozakicho, Chuo-ku

Tokyo 103-8510, Japon

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

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

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

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies. Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Corporation

Dept F6, Bldg 1

294 Route 100

Somers NY 10589-3216

U.S.A.

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

Le programme sous licence décrit dans ce document et tous les éléments sous licence associés sont fournis par IBM selon les termes de l'IBM Customer Agreement, de l'IBM International Program License Agreement ou de tout contrat équivalent.

Les données de performance indiquées dans ce document ont été déterminées dans un environnement contrôlé. Par conséquent, les résultats peuvent varier de manière significative selon l'environnement d'exploitation utilisé. Certaines mesures évaluées sur des systèmes en cours de développement ne sont pas garanties sur tous les systèmes disponibles. En outre, elles peuvent résulter d'extrapolations. Les résultats peuvent donc varier. Il incombe aux utilisateurs de ce document de vérifier si ces données sont applicables à leur environnement d'exploitation.

Les informations concernant des produits non IBM ont été obtenues auprès des fournisseurs de ces produits, par l'intermédiaire d'annonces publiques ou via d'autres sources disponibles.

IBM n'a pas testé ces produits et ne peut confirmer l'exactitude de leurs performances ni leur compatibilité. Elle ne peut recevoir aucune réclamation concernant des produits non IBM. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Toute instruction relative aux intentions d'IBM pour ses opérations à venir est susceptible d'être modifiée ou annulée sans préavis et doit être considérée uniquement comme un objectif.

Tous les tarifs indiqués sont les prix de vente actuels suggérés par IBM et sont susceptibles d'être modifiés sans préavis. Les tarifs appliqués peuvent varier selon les revendeurs.

Ces informations sont fournies uniquement à titre de planification. Elles sont susceptibles d'être modifiées avant la mise à disposition des produits décrits.

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

LICENCE DE COPYRIGHT :

Ces informations contiennent des exemples de programmes d'application en langage source qui illustrent des techniques de programmation sur diverses plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation des plateformes pour lesquels ils ont été écrits ou aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes. Les exemples de programmes sont fournis "EN L'ÉTAT", sans garantie d'aucune sorte. IBM décline toute responsabilité relative aux dommages éventuels résultant de l'utilisation de ces exemples de programmes.

Toute copie intégrale ou partielle de ces exemples de programmes et des oeuvres qui en sont dérivées doit inclure une mention de droits d'auteur libellée comme suit :

© (nom de votre société) (année). Des segments de code sont dérivés des exemples de programmes d'IBM Corp.

© Copyright IBM Corp. _année ou années_. All rights reserved.

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

Politique de confidentialité

Les Logiciels IBM, y compris les Logiciels sous forme de services ("Offres Logiciels") peuvent utiliser des cookies ou d'autres technologies pour collecter des informations sur l'utilisation des produits, améliorer l'acquis utilisateur, personnaliser les interactions avec celui-ci, ou dans d'autres buts. Bien souvent,

aucune information personnelle identifiable n'est collectée par les Offres Logiciels. Certaines Offres Logiciels vous permettent cependant de le faire. Si la présente Offre Logiciels utilise des cookies pour collecter des informations personnelles identifiables, des informations spécifiques sur cette utilisation sont fournies ci-après.

Selon la configuration déployée, la présente Offre Logiciels peut utiliser des cookies de session et des cookies persistants destinés à collecter le nom et le mot de passe des utilisateurs pour les fonctions de gestion des sessions et d'authentification, pour faciliter l'utilisation des produits, pour la configuration de la connexion unique et/ou pour d'autres fonctions de suivi ou buts fonctionnels. Ces cookies ou d'autres technologies similaires ne peuvent pas être désactivés.

Si les configurations déployées de cette Offre Logiciels vous permettent, en tant que client, de collecter des informations permettant d'identifier les utilisateurs par l'intermédiaire de cookies ou par d'autres techniques, vous devez solliciter un avis juridique sur la réglementation applicable à ce type de collecte, notamment en termes d'information et de consentement.

Pour plus d'informations sur l'utilisation à ces fins des différentes technologies, y compris celle des cookies, consultez les Points principaux de la Déclaration IBM de confidentialité sur Internet à l'adresse <http://www.ibm.com/privacy>, la section "Cookies, pixels espions et autres technologies" de la Déclaration IBM de confidentialité sur Internet à l'adresse <http://www.ibm.com/privacy/details>, ainsi que la page "IBM Software Products and Software-as-a-Service Privacy Statement" à l'adresse <http://www.ibm.com/software/info/product-privacy>.

Marques

IBM, le logo IBM et [ibm.com](http://www.ibm.com) sont des marques ou des marques déposées d'International Business Machines Corp. dans de nombreux pays. Les autres noms de produits et de services peuvent être des marques d'IBM ou d'autres sociétés. Une liste des marques commerciales actuelles d'IBM est disponible sur Internet sous "Droits d'auteur et marques" à l'adresse <http://www.ibm.com/legal/us/en/copytrade.shtml>.

D'autres noms peuvent être des marques de leurs propriétaires respectifs. Les autres noms de sociétés, de produits et de services peuvent appartenir à des tiers.

