Oracle® Fusion Cloud EPM

Utilisation d'EPM Automate

F28900-25

Oracle Fusion Cloud EPM Utilisation d'EPM Automate,

F28900-25

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Auteur principal : EPM Information Development Team

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Table des matières

Accessibilité de la documentation

Commentaires sur la documentation

1 A propos d'EPM Automate

Installation d'EPM Automate 2
- Exigences en matière de port et de capacité 2
- Plates-formes prises en charge 2
- Environnement JRE (Java Runtime Environment) et EPM Automate 4
- Utilisation d'OpenJDK 4
- Instructions Windows 5
- Instructions Linux/UNIX/macOS X 5
- Exécution côté serveur des commandes EPM Automate 6

Mise à jour d'EPM Automate 6
Désinstallation d'EPM Automate 7
Présentation du niveau de cryptage pour EPM Automate 7
Utilisation du protocole d'autorisation OAuth 2.0 7

2 Référence de commande

A propos de l'exécution des commandes EPM Automate 1
- Prérequis 1
- Emplacements de fichier par défaut 3
- Activation du protocole TLS 4
- Utilisation des commandes EPM Automate 4
- Utilisation d'un fichier de paramètres afin de transmettre des valeurs pour l'exécution de commandes 5
- Utilisation de variables de substitution avec certaines commandes Enterprise Profitability and Cost Management 8
- Spécification de plusieurs valeurs pour un paramètre 9
- Comportement pendant la maintenance quotidienne 9

Exécution d'EPM Automate 9

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Windows 10
Linux 11
Exécution de plusieurs instances d'EPM Automate 12
Vue d'ensemble des commandes 14
Commandes EPM Automate 20
addUsers 20
addUsersToGroup 21
addUsersToTeam 22
addUserToGroups 23
applicationAdminMode 24
applyDataGrants 25
archiveTmTransactions 25
assignRole 27
autoPredict 29
calculateModel 29
clearCube 32
clearDataByPointOfView 32
clearDataByProfile 33
clearPOV 34
cloneEnvironment 35
compactCube 40
copyDataByPointOfView 40
copyDataByProfile 41
copyFileFromInstance 42
copyFromObjectStorage 43
copyFromSFTP 44
copyOwnershipDataToNextYear 45
copyPOV 46
copySnapshotFromInstance 47
copyToObjectStorage 48
copyToSFTP 50
createGroups 51
createNRSnapshot 52
createReconciliations 52
deleteFile 53
deleteGroups 54
deletePointOfView 55
deletePOV 55
deployCube 56
deployEJTemplates 57
deployFormTemplates 58
deployTaskManagerTemplate 59

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

dismissIPMInsights 59
downloadFile 60
enableApp 61
nenableQueryTracking 61
encrypt 62
essbaseBlockAnalysisReport 63
executeAggregationProcess 64
executeBurstDefinition 65
executeReportBurstingDefinition 65
exportAccessControl 66
exportAppAudit 67
exportAppSecurity 68
exportARApplicationProperties 68
exportBackgroundImage 69
exportCellLevelSecurity 70
exportConsolidationJournals 70
exportData 71
exportDataManagement 71
exportDimension 72
exportDimensionMapping 73
exportEJJournals 74
exportEssbaseData 75
exportJobConsole 76
exportLibraryArtifact 79
exportLibraryDocument 80
exportLogoImage 81
exportMapping 82
exportMetadata 82
exportOwnershipData 83
exportQueryResults 83
exportSnapshot 86
exportTemplate 87
exportTaskManagerAccessControl 87
exportValidIntersections 88
extractDimension 89
extractPackage 90
feedback 91
getApplicationAdminMode 92
getDailyMaintenanceStartTime 92
getEssbaseQryGovExecTime 93
getIdleSessionTimeout 94
getIPAllowlist (désapprouvé) 94

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

getRestrictedDataAccess 95
getSubstVar 95
getVirusScanOnFileUploads 96
groupAssignmentAuditReport 97
help 98
importAppAudit 98
importAppSecurity 99
importARApplicationProperties 100
importBackgroundImage 101
importBalances 101
importCellLevelSecurity 102
importConsolidationJournals 103
importData 103
importDataManagement 104
importDimension 105
importEJDimensionMembers 106
importJobConsole 107
importLibraryArtifact 108
importLibraryDocument 109
importLogoImage 110
importMapping 110
importMetadata 111
importOwnershipData 113
importPreMappedBalances 114
importPreMappedTransactions 115
importProfiles 115
importRates 116
importRCAttributeValues 117
importReconciliationAttributes 118
importSDMDimensionMembers 119
importSnapshot 120
importSupplementalCollectionData 123
importSupplementalData 124
importTemplate 125
importTMAttributeValues 125
importTmPremappedTransactions 127
importValidIntersections 128
invalidLoginReport 129
listBackups 131
listFiles 132
loadData 133
loadDimensionViewpoint 133

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

loadDimData 134
loadViewpoint 135
login 136
logout 140
maskData 141
mergeDataSlices 141
mergeSlices 142
optimizeASOCube 142
programDocumentationReport 143
provisionReport 144
purgeArchivedTmTransactions 146
purgeTmTransactions 147
recomputeOwnershipData 148
recreate 149
refreshCube 152
refreshReportPackageDatasource 152
removeUserFromGroups 153
removeUsers 154
removeUsersFromGroup 155
removeUsersFromTeam 156
renameSnapshot 157
replay 157
resetService 159
restoreBackup 160
restructureCube 161
roleAssignmentAuditReport 162
roleAssignmentReport 163
runAutoAlert 165
runAutomatch 165
runBatch 166
runBusinessRule 167
runCalc 168
runComplianceReport 169
runDailyMaintenance 170
runDataRule 171
runDMReport 173
runIntegration 174
runIntercompanyMatchingReport 178
runMatchingReport 179
runPipeline 180
runPlanTypeMap 182
runRuleSet 183

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

runSupplementalDataReport 183
runTaskManagerReport 184
sendMail 185
setApplicationAdminMode 186
setDailyMaintenanceStartTime 187
setDemoDates 188
setEJJournalStatus 189
setEncryptionKey 190
setEssbaseQryGovExecTime 191
setIdleSessionTimeout 191
setIPAllowlist (désapprouvé) 192
setManualDataAccess 194
setPeriodStatus 195
setRestrictedDataAccess 195
setSubstVars 196
setVirusScanOnFileUploads 196
simulateConcurrentUsage 197
skipUpdate 202
snapshotCompareReport 204
sortMember 205
unassignRole 206
updateGuidedLearningSettings 208
updateUsers 208
upgrade 209
uploadFile 210
userAuditReport (désapprouvé) 211
userGroupReport 213
userLoginReport 214
validateConsolidationMetadata 215
validateModel 216
Codes de sortie 217

3 Scénarios d'exemple d'exécution de commandes

A propos de la copie d'exemples de script 1
Exemples de scénarios pour tous les services 1
Sauvegarde d'un instantané d'application sur un ordinateur 3
Communication de la fin de la maintenance quotidienne aux utilisateurs 5
Copie d'un instantané à partir d'Oracle Object Storage ou vers celui-ci 12
Création d'utilisateurs et affectation de rôles prédéfinis 14
Décompte des utilisateurs disposant d'une licence (utilisateurs affectés à des rôles) 18
Création de rapports d'audit sur les utilisateurs affectés à des rôles 20

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Création d'un rapport d'audit sur la révocation et l'affectation de rôle 24
Masquage des journaux d'accès et du rapport d'activité pour respecter les lois sur la confidentialité 27
Automatisation des téléchargements de rapport d'activité sur un ordinateur local 32
Téléchargement des journaux d'accès à partir d'un environnement 36
Automatisation du clonage d'environnements 39
Clonage quotidien de l'environnement principal vers l'environnement de secours une fois la maintenance quotidienne de l'environnement principal effectuée 43
Suppression des fichiers inutiles d'un environnement 52
Recherche et téléchargement de fichiers à partir d'un environnement 54
Recréation d'un ancien environnement Cloud EPM pour les audits 55
Automatisation de l'audit et de la conformité de l'accès à la base de données 66
Réplication des utilisateurs et des affectations de rôle prédéfini 77
Réplication des utilisateurs d'un domaine d'identité dans un autre 78
Réplication des affectations de rôle prédéfini d'un environnement dans un autre 84
Création d'une cadence de mise à niveau Cloud EPM trimestrielle 91
Instructions et script Windows 92
Instructions et script UNIX/Linux 95
Script Groovy 99
Création d'une cadence de mise à niveau Cloud EPM trimestrielle avec des cycles de test de six semaines 103
Exemples de scénarios pour Planning, Consolidation, Tax Reporting et Enterprise Profitability and Cost Management 117
Automatisation de l'export d'un grand nombre de cellules à partir d'un cube en mode "aggregate storage" 118
Import de métadonnées dans une application 127
Import de données, exécution d'un script de calcul et copie des données d'une base de données en mode "block storage" vers une base de données en mode "aggregate storage" 129
Export et téléchargement de données et de métadonnées 132
Export et téléchargement de données d'application 134
Automatisation de l'archivage des enregistrements d'audit d'application 136
Script Windows 137
Script Linux 138
Téléchargement d'un fichier de données dans un environnement et exécution d'une règle de chargement de données 139
Automatisation de l'intégration quotidienne des données 142
Exemples de scénarios pour Account Reconciliation 144
Chargement de soldes préformatés dans une période 145
Chargement et import d'un instantané de sauvegarde 147
Archivage des anciennes transactions mises en correspondance et purge des transactions archivées 149
Exemples de scénarios pour Profitability and Cost Management 155
Import de métadonnées dans une application 155

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Import de données et exécution de règles de programme 157
Exemples de scénarios pour Cloud EDM 160
Synchronisation de dimensions et de mappings Cloud EDM avec des applications Cloud EPM 160
Synchronisation de dimensions Cloud EPM avec des applications Cloud EDM 162
Automatisation de l'exécution de scripts 163
Surveillance des activités EPM Automate 164

4 Exécution de commandes sans installer EPM Automate

Environnements prenant en charge l'exécution de commandes côté serveur 1
Sources d'information 2
Commandes prises en charge 2
Méthodes à utiliser pour l'exécution d'EPM Automate à l'aide de Groovy côté serveur 2
Clonage d'un environnement à l'aide d'un script Groovy côté serveur 3
Envoi par courriel du rapport d'activité à l'aide d'un script Groovy côté serveur 5

5 Réplication d'environnements Cloud EPM

Configuration de la réplication quotidienne 1
Configuration de réplications à la demande 2
Configuration de l'environnement secondaire 3

A Préparation de l'exécution de la commande simulateConcurrentUsage

Création du fichier requirement.csv A-2
Création des fichiers d'entrée A-3
Fichier d'entrée d'ouverture de formulaire A-4
Fichier d'entrée d'enregistrement de formulaire A-5
Fichier d'entrée d'exécution de règle métier A-7
Fichier d'entrée d'exécution d'ensemble de règles métier A-7
Fichier d'entrée d'exécution de règle de données A-7
Fichier d'entrée de grille ad hoc A-8
Fichier d'entrée d'exécution de rapport A-8
Fichier d'entrée d'exécution de liasse A-9
Création du fichier UserVarMemberMapping.csv A-9
Création du fichier options.xml A-10
Création du fichier users.csv A-10
Création du fichier d'entrée ZIP et chargement dans l'environnement A-10
Exemple de rapport de simulation d'utilisation simultanée A-11

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

B Préparation de l'exécution de la commande replay

A propos de la commande replay B-1
Prérequis B-1
Création de fichiers HAR B-2
Création de fichiers de réexécution B-5
Génération de fichiers de trace B-6
Exemple de session de réexécution B-6

C Gestion des caractères spéciaux

D Commandes propres à chaque service Cloud EPM

Commandes Account Reconciliation D-2
Commandes Financial Consolidation and Close D-3
Commandes Narrative Reporting D-4
Commandes Cloud EDM D-5
Commandes Planning, de modules Planning, FreeForm, Strategic Workforce Planning et Sales Planning D-6
Commandes Profitability and Cost Management D-7
Commandes Enterprise Profitability and Cost Management D-8
Commandes Tax Reporting D-9

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Accessibilité de la documentation

Pour plus d'informations sur l'engagement d'Oracle pour l'accessibilité de la documentation, visitez le site Web Oracle Accessibility Program, à l'adresse http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Accès aux services de support Oracle

Les clients Oracle qui ont souscrit un contrat de support ont accès au support électronique via My Oracle Support. Pour plus d'informations, visitez le site http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info ou le site http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs si vous êtes malentendant.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Commentaires sur la documentation

Pour envoyer des commentaires sur cette documentation, cliquez sur le bouton Commentaires situé en bas de la page de chaque rubrique du centre d'aide Oracle. Vous pouvez également envoyer un courriel à l'adresse epmdoc_ww@oracle.com.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

1

A propos d'EPM Automate

EPM Automate permet aux utilisateurs d'effectuer des tâches à distance dans les environnements Oracle Fusion Cloud Enterprise Performance Management.

Les administrateurs de service Cloud EPM peuvent automatiser de nombreuses tâches réitérables, notamment les suivantes :

- Importer et exporter des métadonnées, des données, des instantanés d'application et d'artefact, des modèles et des mappings de gestion des données
- Télécharger des fichiers dans des environnements, répertorier des fichiers et supprimer des fichiers du service
- Télécharger des instantanés, des rapports, et des fichiers de données et de métadonnées à partir du service
- Exécuter des règles métier sur les données et actualiser l'application
- Copier les données d'une base de données vers une autre ; en général, d'une base de données en mode "block storage" vers une base de données en mode "aggregate storage", ou d'une base de données en mode "block storage" vers une autre base de données en mode "block storage"
- Exécuter une règle de lot de gestion des données
- Générer des rapports de gestion des données, le rapport sur le provisionnement et le rapport d'audit d'utilisateur
- Importer des données de soldes pré-mappés, des taux de change, des transactions pré-mappées, des données de soldes et des profils dans Account Reconciliation
- Copier des profils sur une période pour lancer le processus de rapprochement
- Déployer le cube de calcul des applications Profitability and Cost Management
- Effacer, copier et supprimer les points de vue dans les applications Enterprise Profitability and Cost Management et Profitability and Cost Management
- Réexécuter le chargement d'Oracle Smart View for Office ou de l'API REST dans un environnement afin de permettre le test des performances dans des conditions de charge élevée
- Importer des données supplémentaires à partir d'un fichier dans Financial Consolidation and Close

Vous pouvez créer des scripts capables d'effectuer une large gamme de tâches et d'automatiser leur exécution à l'aide d'un planificateur. Par exemple, vous pouvez créer un script afin de télécharger la sauvegarde de maintenance quotidienne à partir des environnements pour créer des sauvegardes locales de vos artefacts et de vos données.

Tutoriel : exécution des commandes Planning à l'aide d'EPM Automate

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Installation d'EPM Automate

Vous installez EPM Automate pour exécuter des commandes. Certaines commandes peuvent également être exécutées directement dans Oracle Fusion Cloud Enterprise Performance Management à l'aide de scripts Groovy sans installer EPM Automate.

Le programme d'installation d'EPM Automate pour Windows, Linux/UNIX et macOS X est disponible à partir de votre environnement Cloud EPM.

Dans la mesure où Windows 10 et versions ultérieures n'autorisent que les administrateurs Windows à installer EPM Automate, eux seuls peuvent les installer et les mettre à niveau. EPM Automate peut être mis à niveau par l'utilisateur qui l'a installé ou par un autre administrateur Windows.

Dans cette section :

- Exigences en matière de port et de capacité
- Plates-formes prises en charge
- Environnement JRE (Java Runtime Environment) et EPM Automate
- Utilisation d'OpenJDK
- Instructions Windows
- Instructions Linux/UNIX/macOS X
- Exécution côté serveur des commandes EPM Automate

Exigences en matière de port et de capacité

Dans la mesure où EPM Automate est un client léger, son empreinte reste limitée. Tous les traitements sont effectués dans Oracle Fusion Cloud Enterprise Performance Management.

Vous pouvez installer EPM Automate sur des ordinateurs client standard, des machines virtuelles ou des machines Oracle Integration Cloud pouvant accéder à des hôtes externes via une connexion HTTP sécurisée.

EPM Automate se connecte à Cloud EPM à l'aide du port TLS standard (port 443). Vous n'avez pas besoin d'ouvrir de ports de sortie supplémentaires pour EPM Automate.

Actuellement, EPM Automate ne prend pas en charge l'authentification TLS mutuelle (mTLS).

Plates-formes prises en charge

Vous pouvez installer EPM Automate sur des machines virtuelles et sur des machines Oracle Integration Cloud (OIC) pouvant accéder à des hôtes externes via une connexion HTTP sécurisée.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Note

- EPM Automate peut être utilisé uniquement sur les systèmes d'exploitation 64 bits qui sont pris en charge actuellement par le fournisseur de système d'exploitation.
- EPM Automate ne fonctionne pas avec un proxy SOCKS, mais uniquement avec un proxy HTTP/HTTPS.
- EPM Automate prend en charge les mécanismes d'authentification de base, Digest, Kerberos, Negotiate et NTLM pour se connecter au serveur proxy.
- EPM Automate peut se connecter à Oracle Fusion Cloud Enterprise Performance Management via des passerelles d'API, telles que Google APIGEE, IBM Data Power et d'autres serveurs de proxy inverse.
  Pour que cela fonctionne, configurez la passerelle ou le proxy inverse en définissant la cible en tant qu'URL de votre environnement Cloud EPM sans aucun contexte, tel que `/epmcloud`. Exemple : `https://epm-idDomain.epm.dataCenterRegion.oraclecloud.com`. Utilisez ensuite l'URL du proxy inverse au lieu de l'URL Cloud EPM dans la commande login. Pour en savoir plus sur la configuration, reportez-vous à la documentation relative à votre passerelle ou serveur proxy.

  Lors de la configuration des paramètres de proxy, veillez à transmettre le code de réponse de Cloud EPM à EPM Automate sans le modifier de quelque manière que ce soit afin de permettre à EPM Automate de traiter correctement les codes de réponse tels que 200, 206, 400, 404, 500, 501, etc. Par exemple, pour IBM DataPower, définissez `proxy HTTP Response` sur `ON`. La passerelle d'API doit également autoriser les méthodes HTTP (GET, POST, PUT, PATCH et DELETE).

Sur les ordinateurs Linux et UNIX, EPM Automate recherche les variables d'environnement suivantes pour déterminer les paramètres de proxy HTTP ou HTTPS :

- proxyHost
- proxyPort

Exemples de paramètres de proxy HTTP :

```
export proxyHost=host.example.com
```

```
export proxyPort=8000
```

Exemples de paramètres de proxy HTTPS :

```
export proxyHost=host.example.com
```

```
export proxyPort=8080
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Note**
>
> EPM Automate peut utiliser le protocole d'authentification OAuth 2.0 pour accéder aux environnements Cloud EPM, s'ils sont configurés pour OAuth, afin d'exécuter des commandes, notamment dans le but d'automatiser leur exécution.
>
> Dans les environnements qui utilisent l'authentification de base, EPM Automate ne fonctionne pas avec les informations d'identification d'authentification unique d'entreprise (fournisseur d'identités). Puisque les utilisateurs ne peuvent pas se connecter à l'aide d'informations d'identification d'entreprise, les comptes utilisateur permettant d'accéder à EPM Automate doivent être conservés dans le domaine d'identité. Si votre abonnement est configuré pour l'authentification unique, vous devez également permettre aux utilisateurs EPM Automate de se connecter avec leurs informations d'identification de domaine d'identité. Reportez-vous à la section Activation de la connexion avec des informations d'identification de domaine d'identité du guide *Administration d'Oracle Cloud Identity Management*.

Instructions de téléchargement : section Téléchargement et installation de clients du guide *Guide de mise en route pour les administrateurs*.

Environnement JRE (Java Runtime Environment) et EPM Automate

Lorsque vous installez EPM Automate sous Windows, l'environnement JRE requis est également installé. Toutefois, aucun environnement JRE n'est inclus dans les programmes d'installation pour Linux, Unix et macOS X. Vous devez disposer d'un accès à une installation d'environnement JRE (version 17) pour utiliser EPM Automate.

Vous avez le droit d'utiliser Oracle Java Standard Edition (SE) avec EPM Automate sans avoir à souscrire séparément un abonnement Java SE. Pour plus de détails sur les licences Oracle JDK avec EPM Automate, reportez-vous au document My Oracle Support 1557737.1 "Support Entitlement for Java SE When Used As Part of Another Oracle Product".

Utilisation d'OpenJDK

Vous pouvez utiliser la version 17 d'OpenJDK au lieu de l'environnement JRE sur les plates-formes Linux, Unix et macOS X.

Vous pouvez télécharger OpenJDK, le JDK de production d'Oracle gratuit et sous licence générale publique, à l'adresse suivante : https://openjdk.java.net. Les instructions d'installation d'OpenJDK y sont également disponibles.

Avant de démarrer une session EPM Automate, définissez la variable d'environnement `JAVA_HOME` afin qu'elle pointe vers votre installation OpenJDK :

Exemple pour macOS X (interpréteur de commandes Bash utilisé) : utilisation d'OpenJDK version 17, installé sur le répertoire de base.

```
cd ~/
export JAVA_HOME=$(/usr/jdk-17.jdk/Contents/Home)
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exemple pour Linux (interpréteur de commandes Bash utilisé) : utilisation d'OpenJDK version 17, installé sur le répertoire de base.

```
cd ~/
export JAVA_HOME=/openjdk/jdk-17.0.12
```

Instructions Windows

Par défaut, EPM Automate est installé dans le dossier `C:\Program Files\Oracle\EPM Automate`. Il s'agit d'un dossier protégé pour lequel seuls les utilisateurs appartenant au groupe Windows Administrateurs disposent d'un contrôle total. Vous devez être membre du groupe Windows Administrateurs pour pouvoir installer EPM Automate.

Pour installer EPM Automate, procédez comme suit :

1. A l'aide d'un navigateur, accédez à un environnement à partir de l'ordinateur Windows sur lequel vous souhaitez installer EPM Automate.
2. Sur la page d'accueil, accédez à **Paramètres et actions** en cliquant sur votre nom d'utilisateur.
3. Cliquez sur **Téléchargements**.
4. Dans la page Téléchargements, cliquez sur **Télécharger pour Windows** dans la section EPM Automate.
5. Enregistrez le programme d'installation sur votre ordinateur.
6. Cliquez avec le bouton droit de la souris sur le programme d'installation (`EPM Automate.exe`) et sélectionnez `Exécuter en tant qu'administrateur`.
7. Dans **Contrôle du compte utilisateur**, cliquez sur **Oui**.
8. Suivez les invites à l'écran pour terminer l'installation.

Instructions Linux/UNIX/macOS X

EPM Automate requiert l'accès à un déploiement de Java 17. La variable d'environnement `JAVA_HOME` doit être définie de façon à pointer vers votre installation JRE.

Pour installer EPM Automate, procédez comme suit :

1. Accédez à un environnement.
2. Sur la page d'accueil, accédez à **Paramètres et actions** en cliquant sur votre nom d'utilisateur.
3. Cliquez sur **Téléchargements**.
4. Sur la page Téléchargements, cliquez sur **Télécharger pour Linux/macOS X** dans la section EPM Automate.
5. Enregistrez le programme d'installation (`EPMAutomate.tar`) dans un répertoire sur lequel vous avez des privilèges de lecture, d'écriture et d'exécution.
6. Procédez comme suit :
   a. Extrayez le contenu du programme d'installation.
   b. Définissez les variables d'environnement requises.
   c. Remplacez le répertoire par `epmautomate/bin` dans le répertoire où le contenu du programme d'installation a été extrait.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

d. Exécutez `epmautomate.sh` :

Dans les scripts suivants, pensez à utiliser les chemins de répertoire réel, le cas échéant. Par exemple, si `EPMAutomate.tar` est extrait dans `HOME/oracle/epmautomate/bin`, utilisez cette valeur pour `EPMAutomate_extract_directory`.

Exemple pour macOS X (interpréteur de commandes Bash utilisé) : installation et exécution à partir du répertoire de base.

```
cd ~/
tar xf directory_containing_downloaded_installer/EPMAutomate.tar
export JAVA_HOME=$(/usr/libexec/java_home)
export PATH $HOME/epmautomate/bin:$PATH
cd EPMAutomate_extract_directory/epmautomate/bin
./epmautomate.sh
```

Exemple pour Linux (interpréteur de commandes Bash utilisé) : installation et exécution à partir du répertoire de base. Il est supposé que la version 17.0.12 de Java est utilisée.

```
cd ~/
tar xf directory_containing_downloaded_installer/EPMAutomate.tar
export JAVA_HOME=/opt/jdk_17.0.12
export PATH ~/Downloads/epmautomate/bin:$PATH
cd EPMAutomate_extract_directory/epmautomate/bin
./epmautomate.sh
```

Exécution côté serveur des commandes EPM Automate

Certaines commandes EPM Automate peuvent être exécutées directement dans Oracle Fusion Cloud Enterprise Performance Management à l'aide de Groovy. Vous n'avez pas besoin d'installer EPM Automate pour exécuter des commandes à l'aide de scripts Groovy.

L'exécution côté serveur des commandes diffère de l'exécution de scripts Groovy sur un ordinateur client afin d'exécuter les commandes EPM Automate.

Pour obtenir des informations détaillées, reportez-vous à Exécution de commandes sans installer EPM Automate.

Mise à jour d'EPM Automate

EPM Automate permet la mise à niveau vers la version la plus récente sans invite.

Une fois que vous êtes connecté pour lancer une session, vous verrez qu'EPM Automate identifie la version actuellement installée. Si la version installée n'est pas la plus récente, l'application vous informe qu'une nouvelle version est disponible.

Exécutez la commande upgrade pour procéder à la mise à niveau sans invite d'EPM Automate.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Désinstallation d'EPM Automate

> **Note**
>
> Suivez ces étapes uniquement si vous voulez supprimer EPM Automate d'un ordinateur client. Si vous voulez mettre à jour la version en cours d'EPM Automate, suivez les instructions de la section Mise à jour d'EPM Automate.

Windows

Pour désinstaller EPM Automate, procédez comme suit :

1. Dans **Paramètres**, ouvrez **Applications**, puis **Applications installées**.
2. Recherchez EPM Automate dans la liste des applications installées.
3. Dans le **menu de dépassement**, sélectionnez **Désinstaller**.
4. Cliquez sur **Désinstaller**.
5. Supprimez tous les fichiers restants du répertoire d'installation, généralement `C:\Program Files\Oracle\EPM Automate`.

Linux/UNIX/macOS X

Supprimez le répertoire EPM Automate et le fichier `EPMAutomate.tar`.

Présentation du niveau de cryptage pour EPM Automate

Oracle Fusion Cloud Enterprise Performance Management utilise TLS (Transport Layer Security) 1.2 ou 1.3 avec des algorithmes de chiffrement forts pour sécuriser la communication avec EPM Automate.

Utilisation du protocole d'autorisation OAuth 2.0

EPM Automate peut utiliser le protocole d'authentification OAuth 2.0 pour accéder aux environnements Oracle Fusion Cloud Enterprise Performance Management afin d'exécuter des commandes, notamment dans le but d'automatiser leur exécution.

Pour activer l'accès à OAuth 2.0, un administrateur de domaine d'identité doit enregistrer votre application en tant que client public dans les services Oracle Cloud Identity. OAuth est mis en place pour l'application et non pour l'ensemble de votre abonnement.

Pour obtenir des instructions détaillées sur la configuration d'OAuth 2.0, reportez-vous à la section Authentification avec OAuth 2 du guide *API REST pour Oracle Fusion Cloud EPM*.

> **Note**
>
> L'authentification de base fonctionne même lorsqu'OAuth est activé pour un environnement. Veillez à ne pas remplacer le fichier de mots de passe cryptés existant si vous envisagez de l'utiliser à l'avenir.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Création d'un fichier de mots de passe cryptés contenant un jeton d'actualisation et un ID client

Les administrateurs de service qui veulent utiliser OAuth 2.0 pour l'accès d'EPM Automate aux environnements ont besoin des informations suivantes afin de créer un fichier de mots de passe cryptés, lequel est ensuite utilisé pour se connecter à l'environnement :

- Jeton d'actualisation
  Reportez-vous à la section Authentification avec OAuth 2 - Uniquement pour les environnements OCI (Gen 2) du guide *API REST pour Oracle Fusion Cloud EPM* afin d'obtenir des instructions détaillées sur l'obtention du jeton d'actualisation.
- ID client
  L'ID client est généré lorsque l'administrateur de domaine d'identité configure l'application pour OAuth. Il est visible dans l'onglet Configuration de l'application, sous **Informations générales**.

Afin de créer le fichier de mots de passe cryptés pour l'authentification OAuth, procédez comme suit :

1. Démarrez une session EPM Automate.
2. Exécutez une commande semblable à la suivante :
   `epmautomate encrypt` *`REFRESH_TOKEN ENCRYPTION_KEY PASSWORD_FILE`* `ClientID=`*`CLIENT_ID`*, où *`REFRESH_TOKEN`* est le jeton d'actualisation décrypté de l'emplacement de stockage sécurisé, *`ENCRYPTION_KEY`* une clé privée de cryptage du mot de passe, et *`PASSWORD_FILE`* le nom et l'emplacement du fichier qui stocke le jeton d'actualisation crypté. Le fichier de mots de passe doit utiliser l'extension `.epw`.

   Reportez-vous au guide encrypt pour obtenir des instructions détaillées.
3. Utilisez le nouveau fichier de mots de passe généré pour vous connecter via OAuth. Pour les exécutions de script automatisées, veillez à mettre à jour les scripts de sorte qu'ils pointent vers le nouveau fichier de mots de passe généré.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

2

Référence de commande

- A propos de l'exécution des commandes EPM Automate
- Exécution d'EPM Automate
- Vue d'ensemble des commandes
- Commandes EPM Automate
- Codes de sortie

Certaines commandes EPM Automate s'appliquent à tous les processus métier, alors que d'autres s'appliquent à un groupe de processus métier. Sauf indication contraire, une commande applicable à un processus métier spécifique (Planning par exemple) ne fonctionne pas avec un autre processus métier (Financial Consolidation and Close par exemple). Toute tentative d'exécution d'une commande sur un processus métier qui ne la prend pas en charge entraînera une erreur. Pour obtenir la liste des commandes applicables à chaque processus métier, reportez-vous à la section Commandes propres à chaque service Cloud EPM.

A propos de l'exécution des commandes EPM Automate

Tous les services Oracle Fusion Cloud Enterprise Performance Management utilisent les commandes EPM Automate pour l'administration à distance des environnements.

- Prérequis
- Emplacements de fichier par défaut
- Activation du protocole TLS
- Utilisation des commandes EPM Automate
- Spécification de plusieurs valeurs pour un paramètre
- Comportement pendant la maintenance quotidienne

Prérequis

Cette section répertorie les prérequis nécessaires à l'utilisation d'EPM Automate, tels que l'utilisation des informations d'identification Oracle Fusion Cloud Enterprise Performance Management et des emplacements de fichier par défaut dans les environnements.

Général

Tous les utilisateurs de Cloud EPM peuvent utiliser leurs informations d'identification de domaine d'identité pour se connecter à un environnement à l'aide d'EPM Automate. Les rôles prédéfinis et les rôles d'application affectés à l'utilisateur déterminent les commandes qu'un utilisateur peut exécuter.

- Le rôle Administrateur de domaine d'identité est requis pour exécuter des commandes qui ajoutent des utilisateurs au domaine d'identité ou en suppriment.
- Tout fichier requis pour exécuter une commande doit exister dans l'environnement. Utilisez la commande uploadFile pour charger des fichiers.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Reportez-vous à la section Emplacements de fichier par défaut pour plus d'informations sur l'emplacement de fichier par défaut utilisé par chaque service.

- Utilisation des extensions de fichier dans les commandes :
  - Indiquez le nom de fichier complet, extension comprise (par exemple, data.csv), pour exécuter les commandes qui effectuent des opérations sur les fichiers. Exemples de commande d'opération sur les fichiers : `deletefile`, `listfiles`, `uploadfile`.
  - N'utilisez pas les extensions de fichier pour exécuter les commandes qui effectuent des opérations de migration. Les opérations de migration requièrent la fourniture du nom d'un instantané.
- Vous devez placer entre guillemets les valeurs de paramètre contenant un espace, par exemple les commentaires, les noms d'emplacement et les chemins de dossier.

Planning

- Jobs
  De nombreuses commandes abordées dans la section suivante exigent des jobs. Les jobs sont des actions, telles que l'import ou l'export de données, qui peuvent être démarrées immédiatement ou planifiées à une date ultérieure ; par exemple, l'import ou l'export des données, et l'actualisation de la base de données.

  A l'aide de la console des jobs, vous devez créer des jobs adaptés pour réaliser les opérations suivantes. Pour obtenir des instructions détaillées sur la création de jobs dans Planning, reportez-vous à la section Gestion des jobs dans le guide *Administration de Planning*.
  - Importer des données dans une application
  - Exporter des données à partir d'une application
  - Importer des métadonnées dans une application
  - Exporter des métadonnées à partir d'une application
  - Copier les données d'une base de données en mode "block storage" vers une base de données en mode "aggregate storage", ou d'une base de données en mode "block storage" vers une autre base de données en mode "block storage"
- Règles métier
  Les règles métier que vous souhaitez exécuter doivent exister dans l'application.

  Utilisez Calculation Manager pour créer des règles métier, qui sont ensuite déployées dans l'application. Reportez-vous au guide *Conception avec Calculation Manager*.

Gestion des données

- Règles de données
  Les règles de chargement des données définissent comment la gestion des données charge des données à partir d'un fichier. Vous devez disposer de règles prédéfinies pour charger des données à l'aide d'EPM Automate.
- Lots
  Vous pouvez charger des données à l'aide de lots définis dans la gestion des données. Les utilisateurs peuvent combiner plusieurs règles de chargement dans un lot et les exécuter en série ou en parallèle.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Emplacements de fichier par défaut

Emplacement de chargement par défaut

Par défaut, tous les fichiers chargés dans Oracle Fusion Cloud Enterprise Performance Management sont stockés à un emplacement par défaut auquel Migration peut accéder.

Vous devez charger dans l'emplacement par défaut les fichiers que Migration doit traiter, par exemple les instantanés que vous souhaitez importer dans le service.

Boîte de réception et boîte d'envoi

Les emplacements des boîtes de réception et d'envoi peuvent différer en fonction des processus métier Cloud EPM. Vous utilisez la boîte de réception pour charger les fichiers que vous souhaitez importer ou traiter d'une autre manière à l'aide d'un processus métier autre que Profitability and Cost Management. La gestion des données peut traiter les fichiers de la boîte de réception ou d'un répertoire de celle-ci.

En général, Cloud EPM stocke dans la boîte d'envoi les fichiers que vous générez via les processus métier, comme les fichiers d'export de données ou de métadonnées.

- La boîte de réception vers laquelle EPM Automate charge les fichiers et la boîte d'envoi qui stocke les fichiers pour téléchargement sont accessibles aux applications ci-après. Vous devez charger des fichiers vers cet emplacement si vous planifiez de les traiter à l'aide d'un processus natif des applications. Vous pouvez également charger des fichiers dans la boîte d'envoi.
  - Planning
  - Modules Planning
  - Account Reconciliation
  - Financial Consolidation and Close
  - Tax Reporting
  - Narrative Reporting
  - Enterprise Profitability and Cost Management

  Vous pouvez utiliser l'explorateur de boîte de réception/d'envoi afin de parcourir les fichiers stockés à l'emplacement par défaut. Les instantanés d'application que vous créez à l'aide d'EPM Automate ne sont pas répertoriés dans l'explorateur de boîte de réception/d'envoi. Vous pouvez les visualiser à partir de l'onglet Instantanés de Migration.
- Les fichiers qui doivent être traités à l'aide d'un processus Profitability and Cost Management doivent être chargés dans `profitinbox`. Vous pouvez également charger des fichiers dans la profitoutbox. Les fichiers exportés par les processus Profitability and Cost Management sont stockés dans `profitinbox`. Utilisez l'explorateur de fichiers pour parcourir ces fichiers.
- Les fichiers qui doivent être traités à l'aide de la gestion des données doivent être disponibles dans la boîte de réception ou dans un dossier qui s'y trouve. Par défaut, les fichiers exportés à l'aide de la gestion des données sont stockés dans la boîte d'envoi, tandis que les sorties de rapport de gestion des données sont stockées dans le dossier `outbox/report` de la gestion des données. Utilisez le navigateur de fichiers de la gestion des données pour parcourir ces fichiers.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Oracle Fusion Cloud Enterprise Data Management utilise l'emplacement par défaut pour les fichiers d'import et d'export qui sont chargés, copiés ou téléchargés. Les fichiers dans l'emplacement par défaut peuvent être affichés à l'aide de la commande `ListFiles`.

Fichiers journaux

Chaque exécution de commande EPM Automate génère un fichier de débogage qui est automatiquement supprimé si la commande aboutit. Si une erreur survient lors de l'exécution de la commande, le fichier de débogage de la commande ayant échoué est conservé dans le répertoire à partir duquel vous exécutez EPM Automate. Par défaut, ce répertoire est `Oracle/epm automate/bin` (Windows) ou `home/user/epmautomate/bin` (Linux/UNIX).

Les fichiers de débogage EPM Automate utilisent la convention de dénomination suivante :

`commandname_date_timestamp.log`. Par exemple, si vous exécutez une commande `listfiles` ayant échoué à 09:28:02 le 23 novembre 2020, le nom du fichier de débogage est `listfiles_23_11_2020_09_28_02.log`.

Vous ne pouvez pas supprimer la création du fichier de débogage pour une commande ayant échoué. Toutefois, vous pouvez écrire des informations de débogage et la sortie de commande dans un fichier qui se trouve dans un autre répertoire en ajoutant `-d` ainsi qu'un nom de fichier de débogage et les flux d'erreur et de sortie (`-d >> c:\logs\`*`LOG_FILE_NAME`*`.log 2>&1`) à la fin de la commande, comme le montre l'exemple suivant pour Windows :

```
epmautomate listfiles -d >> c:\logs\listfiles.log 2>&1
```

Activation du protocole TLS

EPM Automate doit être installé sur un système d'exploitation qui prend en charge la version 1.2 ou supérieure du protocole TLS.

Afin de garantir le plus haut niveau de sécurité lors de l'authentification et du cryptage des données, EPM Automate prend uniquement en charge le protocole TLS 1.2 et 1.3. Si le protocole TLS 1.2 ou 1.3 n'est pas activé sur l'ordinateur utilisé pour exécuter EPM Automate, le message d'erreur `EPMAT-7 : Connexion impossible. Protocole non pris en charge : HTTPS` s'affiche. Pour corriger cette erreur, demandez à votre administrateur informatique de vous aider à activer le protocole TLS 1.2 ou 1.3.

Les procédures d'activation du protocole TLS 1.2 et 1.3 varient en fonction du système d'exploitation. Reportez-vous aux sources d'informations suivantes (des ressources en ligne similaires peuvent être disponibles pour les autres systèmes d'exploitation pris en charge) :

- Activation de TLS 1.3 sous Windows 10 pour en savoir plus sur l'activation du protocole TLS 1.3 sur les ordinateurs Windows.
- TLS version 1.3 sous Red Hat Enterprise Linux 8 afin d'en savoir plus sur l'activation du protocole TLS 1.3 pour Red Hat Enterprise Linux.

Utilisation des commandes EPM Automate

Ordre des paramètres de commande

Tous les paramètres obligatoires d'une commande doivent être transmis dans l'ordre indiqué dans la syntaxe de la commande. Les paramètres obligatoires et leurs valeurs précèdent les paramètres facultatifs, qui peuvent être transmis dans n'importe quel ordre. L'ordre des paramètres facultatifs n'a aucune influence.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Prenons pour exemple la syntaxe de la commande `login` :

```
epmautomate login USERNAME PASSWORD CLOUD_EPM_BASE_URL
[ProxyServerUserName=PROXY_USERNAME] [ProxyServerPassword=PROXY_PASSWORD]
[ProxyServerDomain=PROXY_DOMAIN]
```

Cette commande comporte trois paramètres obligatoires, `USERNAME`, `PASSWORD` et `CLOUD_EPM_BASE_URL`, qui doivent apparaître dans l'ordre indiqué dans la syntaxe. La commande renverra une erreur si l'ordre n'est pas respecté. Les paramètres facultatifs `ProxyServerUserName`, `ProxyServerPassword` et `ProxyServerDomain` et leurs valeurs peuvent être indiqués dans n'importe quel ordre.

Les commandes EPM Automate respectent-elles la casse ?

Les commandes EPM Automate ne respectent pas la casse. L'utilisation de majuscules ou de minuscules n'a pas d'influence sur l'exécution de la commande. Par exemple, pour la commande `addUsers`, vous pouvez saisir `addusers`, `ADDUSERS` ou `AdDuSeRs`.

Les paramètres de commande EPM Automate respectent-ils la casse ?

Les paramètres de commande EPM Automate ne respectent pas la casse. L'utilisation de majuscules ou de minuscules dans le nom de paramètre n'a pas d'incidence sur l'exécution de la commande. Par exemple, vous pouvez saisir le paramètre `FileName` comme suit, sans incidence sur l'exécution de la commande : `filename`, `fileName` ou `fIlEnAmE`.

Utilisation d'un fichier de paramètres afin de transmettre des valeurs pour l'exécution de commandes

La plupart des commandes EPM Automate requièrent des paramètres d'exécution. Vous pouvez fournir ces valeurs directement dans l'invite de commande ou les stocker dans un fichier de paramètres afin que la commande puisse les utiliser.

Toutes les valeurs de paramètre, y compris les informations d'identification (nom d'utilisateur Oracle Fusion Cloud Enterprise Performance Management, et mot de passe crypté ou en texte brut), requises pour exécuter une commande peuvent être stockées dans un fichier de paramètres au lieu d'être saisies dans l'invite de commande. L'utilisation d'un fichier de paramètres améliore la réutilisabilité et la sécurité lors de l'exécution des commandes.

A propos du fichier de paramètres

Chaque commande requiert son propre fichier de paramètres car les paramètres varient en fonction de la commande. Par exemple, des paramètres différents sont utilisés par la commande copyFromObjectStorage et la commande copyFromSFTP.

Les paramètres de commande sont positionnels, ce qui signifie qu'ils doivent être fournis dans un ordre spécifique. Le fichier de paramètres de chaque commande doit suivre le modèle de syntaxe et la séquence des paramètres obligatoires de la commande concernée. Les paramètres facultatifs sont spécifiés après les paramètres obligatoires et peuvent être répertoriés dans n'importe quel ordre.

Par exemple, le modèle de syntaxe des paramètres obligatoires pour l'exécution de la commande login à l'aide d'un mot de passe crypté est le suivant :

```
epmautomate login USERNAME PASSWORD_FILE URL
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Le fichier de paramètres que vous utilisez pour exécuter la commande `login`, par exemple *login_encrypted.txt*, doit répertorier les valeurs de paramètre dans l'ordre exact indiqué par le modèle de syntaxe précédent.

```
serviceAdmin
C:\mySecuredir\password.epw
https://test-cloud-pln.pbcs.us1.oraclecloud.com
```

Un fichier de paramètres n'a pas besoin d'inclure tous les paramètres requis. Par exemple, le fichier *new_login_encrypted.txt* ne contient pas la valeur du paramètre `username`.

```
C:\mySecuredir\password.epw
https://test-cloud-pln.pbcs.us1.oraclecloud.com
```

Etant donné que la valeur de `username` (`serviceAdmin`) est omise dans le fichier de paramètres, vous devez la fournir dans l'invite de commande.

Reportez-vous à la section Utilisation du fichier de paramètres avec des commandes afin d'obtenir des exemples illustrant l'utilisation de ces fichiers pour exécuter la commande `login`.

Création d'un fichier de paramètres

Le contenu du fichier de paramètres doit correspondre au modèle de syntaxe de la commande avec laquelle le fichier est employé. Etant donné que les paramètres requis sont positionnels, assurez-vous qu'ils apparaissent dans l'ordre exact indiqué dans le modèle de syntaxe.
Pour créer un fichier de paramètres, procédez comme suit :

1. A l'aide d'un éditeur de texte, créez un fichier et enregistrez-le dans un dossier.
2. Copiez le modèle de syntaxe de la commande pour laquelle créer le fichier de paramètres et collez-le dans le fichier texte que vous avez créé à l'étape précédente. Par exemple, si vous créez un fichier de paramètres pour la commande cloneEnvironment, le modèle de syntaxe est le suivant :

   ```
   epmAutomate cloneEnvironment TARGET_USERNAME TARGET_PASSWORD
   TARGET_URL [SnapshotName=NAME] [UsersAndPreDefinedRoles=true|false]
   [DataManagement=true|false] [appAudit=true|false] [jobConsole=true|false]
   [storedSnapshotsAndFiles=true|false] [DailyMaintenanceStartTime=true|false]
   [ApplicationProperties=true|false]
   ```

3. Modifiez le fichier :
   a. Supprimez le texte qui précède le premier paramètre obligatoire. Dans cet exemple, supprimez `epmAutomate cloneEnvironment`.
   b. Remplacez chaque paramètre obligatoire par la valeur que vous souhaitez utiliser.
   c. Remplacez chaque paramètre facultatif par la valeur que vous souhaitez utiliser.
   d. Supprimez les paramètres facultatifs que vous ne voulez pas utiliser. EPM Automate utilise leurs valeurs par défaut.
   e. Insérez un saut de ligne après chaque valeur de paramètre obligatoire et facultatif. Par exemple, le contenu de votre fichier peut être semblable à ce qui suit :

      ```
      serviceAdmin
      Password.epw
      https://epm-test-ociarcs.epm.us.region.ocs.oc-test.com
      ```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
UsersAndPreDefinedRoles=true
ApplicationProperties=false
storedSnapshotsAndFiles=true
DailyMaintenanceStartTime=false
```

4. Enregistrez le fichier de paramètres.

Utilisation du fichier de paramètres avec des commandes

Utilisez l'option `-p FILE_NAME` pour transmettre le fichier de paramètres à la commande.

La position de l'option `-p FILE_NAME` dans la commande détermine la façon dont le contenu de fichier est utilisé. EPM Automate insère le contenu du fichier de paramètres dans lequel l'option `-p FILE_NAME` apparaît.

A l'aide du fichier de paramètres *login_encrypted.txt*, qui contient toutes les valeurs obligatoires (reportez-vous à la section précédente), vous allez exécuter la commande suivante pour démarrer une session :

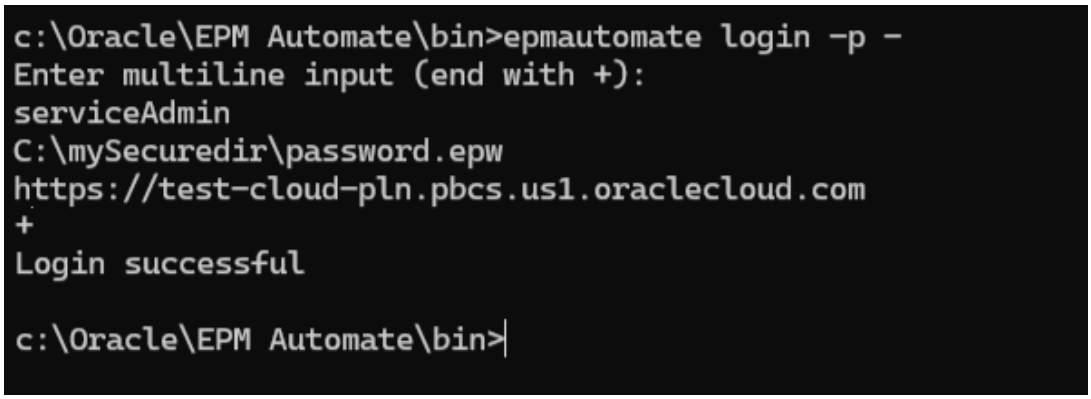
```
epmautomate login -p login_encrypted.txt
```

A l'aide du fichier de paramètres *new_login_encrypted.txt*, qui omet la valeur obligatoire *username*, référencée dans la section précédente, vous allez exécuter la commande suivante pour démarrer une session :

```
epmautomate login serviceAdmin -p new_login_encrypted.txt
```

Saisie de paramètres sur plusieurs lignes

Utilisez l'option `-p -` pour activer l'entrée multiligne, ce qui vous permet de transmettre des paramètres de commande sur plusieurs lignes. Après avoir spécifié tous les paramètres de commande, entrez + (signe plus) sur une nouvelle ligne pour lancer l'exécution de la commande. Voici un exemple :



Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Note**
>
> - Vous ne pouvez pas utiliser un fichier de paramètres en mode d'entrée multiligne. Vous devez saisir manuellement les paramètres dans la console.
> - Chaque paramètre doit être saisi sur une ligne distincte. L'erreur `EPMAT-7 : Paramètre non valide ou manquant` apparaît si vous entrez plusieurs paramètres sur une même ligne après avoir appelé le mode d'entrée multiligne.

Utilisation de variables de substitution avec certaines commandes Enterprise Profitability and Cost Management

Les jobs Enterprise Profitability and Cost Management prennent en charge les variables de substitution en tant que paramètres de job. Vous pouvez utiliser des variables de substitution, qui sont résolues lorsque EPM Automate exécute ces jobs dans le cadre de l'exécution de la commande.

Les jobs utilisés pour exécuter les commandes suivantes prennent en charge les variables de substitution :

- calculateModel
- clearPOV
- copyPOV
- deletePOV
- validateModel

Les variables de substitution sont prises en charge pour tous les paramètres (requis et facultatifs) utilisés avec ces commandes.

Indication de variables de substitution avec des commandes

Pour utiliser une variable de substitution avec une commande, faites commencer la valeur de paramètre par le caractère *&* (esperluette). Dans l'exemple de commande `calculateModel` suivant, vous pouvez remplacer les valeurs réelles `FY22` et `ALL_RULES` par les variables de substitution `&currYear` et `&execType`.

- **Exemple de commande *sans* variables de substitution :**
```
epmautomate calculateModel FY22::Jan::Actual::Working "10 Actuals Allocation Process" ALL_RULES clearCalculatedData=true executeCalculations=true optimizeForReporting=true comment="Running all rules to calculate a POV"
```
- **Exemple de commande avec variables de substitution :**
```
epmautomate calculateModel &currYear::Jan::Actual::Working "10 Actuals Allocation Process" &execType clearCalculatedData=true executeCalculations=true optimizeForReporting=true comment="Running all rules to calculate a POV"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Note

- Seules les valeurs de paramètre qui commencent par le caractère *&* (esperluette) sont traitées comme des variables de substitution.
- La variable de substitution doit exister dans Enterprise Profitability and Cost Management. Les commandes échouent si vous transmettez une variable de substitution qui n'existe pas ou qui est mal orthographiée.

Spécification de plusieurs valeurs pour un paramètre

Certaines commandes EPM Automate acceptent plusieurs valeurs de paramètre séparées par une virgule. Par exemple, une invite d'exécution de type Membres dans les règles métier, les ensembles de règles et les modèles d'une application Planning.

Afin de définir plusieurs membres pour un type Membres d'une invite d'exécution nommée `Entities` dans une commande EPM Automate, utilisez une virgule (,) comme le montre l'exemple suivant pour exécuter la commande `runbusinessrule`.

```
epmautomate runbusinessrule clearDistData TargetYear=FY19
TargetMonth=Feb Entities=District1,District2
```

Les noms de membre contenant des caractères spéciaux, tels que l'espace ou la virgule, doivent être placés entre guillemets et comprendre une barre oblique inverse (\) comme caractère d'échappement, comme le montre l'exemple suivant :

```
epmautomate runbusinessrule clearDistData TargetYear=FY19
TargetMonth=Feb Entities="\"District 1\",\"entity_name, with comma\""
```

Comportement pendant la maintenance quotidienne

N'exécutez pas de commandes EPM Automate pendant la maintenance quotidienne d'un environnement.

Aucune activité utilisateur n'est autorisée pendant la maintenance quotidienne. Si vous tentez d'exécuter des commandes EPM Automate directement ou à l'aide de scripts alors que la maintenance quotidienne est en cours, l'erreur suivante apparaît :

```
EPMAT-11 : Erreur de serveur interne. En raison de la maintenance
quotidienne, votre environnement Oracle EPM Cloud Service est
actuellement indisponible.
```

Exécution d'EPM Automate

Vos informations d'identification Oracle Fusion Cloud Enterprise Performance Management vous permettent de vous connecter à l'aide d'EPM Automate. Vous ne pouvez pas vous connecter à l'aide de vos informations d'identification d'authentification unique.

Tous les utilisateurs de Cloud EPM peuvent utiliser leurs informations d'identification de domaine d'identité pour se connecter à un environnement à l'aide d'EPM Automate. Les rôles prédéfinis et les rôles d'application affectés à l'utilisateur déterminent les commandes que celui-ci peut exécuter.

Par ailleurs, seuls les administrateurs de service peuvent exécuter certaines commandes. Le rôle Administrateur de domaine d'identité peut être nécessaire pour en exécuter d'autres.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Génération du fichier journal de débogage

Le support technique Oracle vous demande le fichier journal de débogage de la session afin de résoudre les problèmes que vous avez rencontrés lors de l'exécution d'EPM Automate. L'option `-d` est prise en charge pour générer les messages de débogage, qui peuvent ensuite être redirigés vers un fichier à l'aide de la directive >. Vous pouvez créer un fichier de débogage pour une seule commande ou un script/fichier d'exécution par lots contenant plusieurs commandes.

Syntaxe : `epmautomate command [command_parameters] -d > log_file 2>&1`

Exemple Windows : `epmautomate downloadfile "Artifact Snapshot" -d > C:\logs\download_log.txt 2>&1`

Exemple Linux : `epmautomate.sh downloadfile "Artifact Snapshot" -d > ./logs/download_log 2>&1`

Windows

Avant d'exécuter EPM Automate, assurez-vous que vous pouvez accéder à votre environnement à partir de l'ordinateur sur lequel vous l'exécutez.

EPM Automate crée un fichier `.prefs`, contenant des informations utilisateur, et des fichiers journaux dans le répertoire en cours. Sur les ordinateurs Windows, le contenu du fichier `.prefs` est uniquement visible par l'utilisateur qui l'a créé et par les administrateurs Windows. Dans les environnements Linux, UNIX et macOS X, le fichier `.prefs` est généré avec l'autorisation 600, qui n'octroie un accès en lecture et en écriture sur ce fichier qu'au propriétaire.

EPM Automate affiche l'erreur `FileNotFoundException: .prefs (Accès refusé)` dans les environnements Windows si vous ne disposez pas d'une autorisation en écriture pour le répertoire Windows à partir duquel vous l'exécutez. Pour résoudre cette erreur, vérifiez que le compte Windows de l'utilisateur actuel dispose d'un accès en lecture/écriture au répertoire à partir duquel EPM Automate est exécuté. De plus, l'utilisateur doit disposer de droits d'accès appropriés sur tout autre répertoire à partir duquel s'effectue l'accès à un fichier (par exemple, lors de l'exécution de la commande `uploadFile`) ou l'écriture dans un fichier (par exemple, lors de l'exécution de la commande `downloadFile`).

> **Remarque**
>
> Vous ne pouvez pas exécuter EPM Automate à partir d'un dossier dont le nom contient `&`, par exemple `C:\Oracle\A&B`.

Pour exécuter EPM Automate sur un client Windows, procédez comme suit :

1. Cliquez sur **Démarrer**, puis **Tous les programmes**, **EPM Automate** et **Lancer EPM Automate**. L'invite de commande EPM Automate s'affiche.
2. **Facultatif :** accédez au répertoire à partir duquel vous voulez exécuter des opérations.
3. **Facultatif :** générez un fichier de cryptage de mot de passe. Le fichier de cryptage de mot de passe permet de transmettre le mot de passe crypté afin de lancer une session.

   ```
   epmautomate encrypt P@ssword1 myKey C:/mySecuredir/password.epw
   ```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

4. Démarrez une session en tant qu'administrateur de service. Utilisez une commande telle que ci-après :
   - Utilisation d'un mot de passe non crypté :

     ```
     epmautomate login serviceAdmin P@ssword1
     https://test-cloudpln.pbcs_us1.oraclecloud.com
     ```

   - Utilisation d'un mot de passe crypté :

     ```
     epmautomate login serviceAdmin C:\mySecuredir\password.epw
     https://test-cloudpln.pbcs_us1.oraclecloud.com
     ```

5. Entrez des commandes permettant d'exécuter les tâches que vous souhaitez réaliser.

   Reportez-vous à la section Codes de sortie pour plus d'informations sur le statut d'exécution des commandes.

6. Déconnectez-vous de l'environnement. Utilisez la commande suivante :

   ```
   epmautomate logout
   ```

Linux

> **Remarque**
>
> Assurez-vous que *JAVA_HOME* est défini dans la variable `PATH` de votre fichier `.profile` ou en tant que variable d'environnement d'interpréteur de commandes. Une version prise en charge de JRE (17) est requise.

Pour exécuter EPM Automate sur un client Linux, procédez comme suit :

1. Ouvrez une fenêtre de terminal et accédez au répertoire dans lequel vous avez installé EPM Automate.
2. **Facultatif :** générez un fichier de cryptage de mot de passe. Utilisez le fichier de cryptage de mot de passe pour transmettre un mot de passe crypté au lieu d'un mot de passe non crypté afin d'initier une session.

   ```
   epmautomate encrypt P@ssword1 myKey ../misc/encrypt/password.epw
   ```

3. Démarrez une session en tant qu'administrateur de service. Utilisez une commande telle que ci-après :
   - Utilisation d'un mot de passe non crypté :

     ```
     ./bin/epmautomate.sh login serviceAdmin P@ssword1
     https://test-cloudpln.pbcs_us1.oraclecloud.com
     ```

   - Utilisation d'un mot de passe crypté :

     ```
     ./bin/epmautomate.sh login serviceAdmin ../misc/encrypt/password.epw
     https://test-cloudpln.pbcs_us1.oraclecloud.com
     ```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

4. Entrez des commandes permettant d'exécuter les tâches que vous souhaitez réaliser.

   Reportez-vous à la section Codes de sortie pour plus d'informations sur le statut d'exécution des commandes.

5. Déconnectez-vous de l'environnement. Utilisez la commande suivante :

   ```
   ./bin/epmautomate.sh logout
   ```

Exécution de plusieurs instances d'EPM Automate

Vous pouvez exécuter plusieurs instances d'EPM Automate sur un seul environnement à partir du même répertoire. Vous pouvez également exécuter plusieurs instances sur différents environnements, à partir du même répertoire ou de plusieurs répertoires.
Par exemple, vous devrez peut-être actualiser simultanément le cube de l'application Planning sur les pages `https://cloudpln.pbcs.us1.oraclecloud.com` et `https://testcloudpln.pbcs.us1.oraclecloud.com`. Dans ce scénario, deux possibilités s'offrent à vous :

- Exécuter deux instances d'EPM Automate à partir du même répertoire pour actualiser les cubes d'application dans différents environnements
- Exécuter EPM Automate à partir de répertoires distincts pour se connecter aux environnements, puis actualiser les cubes d'application

Dans les deux scénarios, chaque instance fonctionne indépendamment : la déconnexion d'une instance ne vous déconnecte pas des autres. Les activités lancées à l'aide d'EPM Automate poursuivent leur exécution jusqu'à achèvement dans l'environnement, même si vous vous déconnectez de l'autre instance.

Cette section contient des échantillons de script pour Windows et Unix/Linux (`caller` et `multisession`) que vous pouvez utiliser afin de créer deux sessions EPM Automate en vue d'exécuter les tâches. Pour exécuter plusieurs sessions simultanées, vous devez ajouter les informations de connexion suivantes dans le script `caller`, qui appelle le script `multisession` pour exécuter les commandes `login`, `uploadfile`, `listfiles` et `logout`. Vous pouvez modifier le script `multisession` pour effectuer d'autres tâches. Assurez-vous que ces deux scripts sont stockés dans le même répertoire.

- EPM Automate utilise la variable d'environnement `EPM_SID` pour distinguer plusieurs sessions. Cette variable doit être définie dans le script demandeur sur une valeur unique pour chaque session. Dans les exemples de scripts, il est défini sur les valeurs uniques suivantes :
  - Dans `caller.BAT`, `EPM_SID` est défini sur `!RANDOM!`, ce qui lui affecte un numéro unique généré par le système. Ce numéro est également utilisé lors de la génération des fichiers journaux des sessions. Si vous souhaitez suivre le fichier journal de chaque session, vous pouvez indiquer un numéro unique au lieu de `!RANDOM!`.
  - Dans `caller.sh`, `EPM_SID` est défini sur l'ID de processus, qui est unique. Si vous souhaitez suivre le fichier journal de chaque session, vous pouvez indiquer un `EPM_SID` unique en modifiant l'instruction `export EPM_SID=$$` dans le script `multisession` afin d'utiliser la valeur transmise, puis transmettre une valeur unique pour ce paramètre dans le script `caller` pour chaque session, par exemple en indiquant la valeur `EPM SID` dans `caller.sh` comme suit :

    ```
    $SCRIPT_DIR/multisession.sh EPM_SID "USERNAME" "PASSWORD" "URL" "/home/
    user/Snapshot1.zip" &
    ```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
$SCRIPT_DIR/multisession.sh EPM_SID "USERNAME" "PASSWORD" "URL" "/home/
user/Snapshot2.zip" &
```

- `USERNAME` : ID de connexion de l'administrateur de service
- `PASSWORD` : mot de passe de l'administrateur de service
- `URL` : URL de connexion de l'environnement

Exemples de script Windows

caller.BAT

```
@echo off
setlocal EnableExtensions EnableDelayedExpansion

REM syntax: start /B multisession.bat EPM_SID "USERNAME" "PASSWORD" "URL"
"SNAPSHOTPATH"
start /B multisession.bat !RANDOM! "USERNAME" "PASSWORD" "URL"
"C:\Snapshot1.zip"
start /B multisession.bat !RANDOM! "USERNAME" "PASSWORD" "URL"
"C:\Snapshot2.zip"

endlocal
```

multisession.BAT

```
@echo off

set EPM_SID=%1
set USERNAME=%2
set PASSWORD=%3
set URL=%4
set SNAPSHOTNAME=%5

echo User: %USERNAME% > %EPM_SID%.log
echo Cloud Instance: %URL% >> %EPM_SID%.log

call epmautomate login %USERNAME% %PASSWORD% %URL% >> %EPM_SID%.log
call epmautomate uploadfile %SNAPSHOTNAME% >> %EPM_SID%.log
call epmautomate listfiles >> %EPM_SID%.log
call epmautomate logout
```

Exemple de script shell Bourne

caller.sh

```
#!/bin/sh

set +x
SCRIPT_DIR=`dirname "${0}"`

# syntax: /home/user/multisession.sh "USERNAME" "PASSWORD" "URL"
"SNAPSHOTPATH" &
$SCRIPT_DIR/multisession.sh "USERNAME" "PASSWORD" "URL"  "/home/user/
Snapshot1.zip" &
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
$SCRIPT_DIR/multisession.sh "USERNAME" "PASSWORD" "URL" "/home/user/
Snapshot2.zip" &
```

multisession.sh

```
#!/bin/sh

set +x

EPM_AUTOMATE_HOME=/home/user/epmautomate

export JAVA_HOME=/home/user/jre
export EPM_SID=$$

USERNAME=$1
PASSWORD=$2
URL=$3
SNAPSHOTNAME=$4

echo User: $USERNAME > $EPM_SID.log
echo Cloud Instance: $URL >> $EPM_SID.log

$EPM_AUTOMATE_HOME/bin/epmautomate.sh login $USERNAME $PASSWORD $URL
>> $EPM_SID.log
$EPM_AUTOMATE_HOME/bin/epmautomate.sh uploadfile $SNAPSHOTNAME >> $EPM_SID.log
$EPM_AUTOMATE_HOME/bin/epmautomate.sh listfiles >> $EPM_SID.log
$EPM_AUTOMATE_HOME/bin/epmautomate.sh logout
```

Vue d'ensemble des commandes

Cette liste répertorie l'ensemble des commandes EPM Automate dans l'ordre alphabétique.

Tableau 2-1 Toutes les commandes EPM Automate

Nom de commande	PLN, SWP, SP, FF	FCC	TR	PCM	EPCM	AR	EDM	NR
addUsers	✓	✓	✓	✓	✓	✓	✓	✓
addUsersToGroup	✓	✓	✓	✓	✓	✓	✓	✓
addUsersToTeam		✓	✓			✓		
addUserToGroups	✓	✓	✓	✓	✓	✓	✓	✓
applicationAdminMode	✓	✓	✓		✓			
applyDataGrants				✓				
archiveTmTransactions						✓		
assignRole	✓	✓	✓	✓	✓	✓	✓	✓
autoPredict* Cf. note de bas de page	✓							
calculateModel					✓			
clearCube	✓				✓			
clearDataByPointOfView					✓			
clearDataByProfile		✓	✓					

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 2-1 (suite) Toutes les commandes EPM Automate

Nom de commande	PLN, SWP, SP, FF	FCC	TR	PCM	EPCM	AR	EDM	NR
clearPOV				✓				
cloneEnvironment	✓	✓	✓	✓	✓	✓	✓	✓
compactCube	✓				✓			
copyDataByPointOfView					✓			
copyDataByProfile		✓	✓					
copyFileFromInstance	✓	✓	✓	✓	✓	✓	✓	✓
copyFromObjectStorage	✓	✓	✓	✓	✓	✓	✓	✓
copyFromSFTP	✓	✓	✓	✓	✓	✓	✓	✓
copyOwnershipDataToNextYear		✓	✓					
copyPOV				✓				
copySnapshotFromInstance	✓	✓	✓	✓	✓	✓	✓	
copyToObjectStorage	✓	✓	✓	✓	✓	✓	✓	✓
copyToSFTP	✓	✓	✓	✓	✓	✓	✓	✓
createGroups	✓	✓	✓	✓	✓	✓	✓	✓
createNRSnapshot								✓
createReconciliations						✓		
deleteFile	✓	✓	✓	✓	✓	✓	✓	✓
deleteGroups	✓	✓	✓	✓	✓	✓	✓	✓
deletePointOfView					✓			
deletePOV				✓				
deployCube				✓				
deployEJTemplates		✓						
deployFormTemplates		✓	✓					
deployTaskManagerTemplate		✓			✓			
dismissIPMInsights**	✓							
downloadFile	✓	✓	✓	✓	✓	✓	✓	✓
enableApp				✓				
nenableQueryTracking	✓				✓			
encrypt	✓	✓	✓	✓	✓	✓	✓	✓
essbaseBlockAnalysisReport	✓	✓	✓					
executeAggregationProcess	✓				✓			
executeBurstDefinition								✓
executeReportBurstingDefinition	✓	✓	✓		✓			
exportAccessControl						✓		
exportAppAudit	✓	✓	✓		✓			
exportAppSecurity	✓	✓	✓		✓			
exportARApplicationProperties						✓		
exportBackgroundImage						✓		
exportCellLevelSecurity	✓		✓		✓			
exportData	✓	✓	✓		✓			

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 2-1 (suite) Toutes les commandes EPM Automate

Nom de commande	PLN, SWP, SP, FF	FCC	TR	PCM	EPCM	AR	EDM	NR
exportConsolidationJournals		✓						
exportDataManagement	✓	✓	✓	✓	✓			
exportDimension							✓	
exportDimensionMapping							✓	
exportEJJournals		✓						
exportEssbaseData	✓	✓	✓		✓			
exportJobConsole	✓	✓	✓		✓			
exportLibraryArtifact								✓
exportLibraryDocument	✓	✓	✓		✓			
exportLogoImage						✓		
exportMapping	✓	✓	✓	✓	✓	✓		
exportMetadata	✓	✓	✓		✓			
exportOwnershipData		✓	✓					
exportQueryResults				✓				
exportSnapshot	✓	✓	✓	✓	✓	✓	✓	
exportTemplate				✓				
exportTaskManagerAccessControl		✓	✓		✓			
exportValidIntersections	✓	✓	✓		✓			
extractDimension							✓	
extractPackage							✓	
feedback	✓	✓	✓	✓	✓	✓	✓	✓
getApplicationAdminMode	✓	✓	✓		✓	✓		
getDailyMaintenanceStartTime	✓	✓	✓	✓	✓	✓	✓	✓
getEssbaseQryGovExecTime	✓	✓	✓	✓	✓			
getIdleSessionTimeout	✓	✓	✓	✓	✓	✓	✓	✓
getIPAllowlist (désapprouvé)	✓	✓	✓	✓	✓	✓	✓	✓
getRestrictedDataAccess	✓	✓	✓	✓	✓	✓	✓	✓
getSubstVar	✓	✓	✓		✓			
getVirusScanOnFileUploads	✓	✓	✓	✓	✓	✓	✓	✓
groupAssignmentAuditReport	✓	✓	✓	✓	✓	✓	✓	✓
help	✓	✓	✓	✓	✓	✓	✓	✓
importAppAudit	✓				✓			
importAppSecurity	✓	✓	✓		✓			
importARApplicationProperties						✓		
importBackgroundImage						✓		
importBalances						✓		
importCellLevelSecurity	✓		✓		✓			
importConsolidationJournals		✓						
importData	✓	✓	✓	✓	✓			
importDataManagement	✓	✓	✓	✓	✓			

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 2-1 (suite) Toutes les commandes EPM Automate

Nom de commande	PLN, SWP, SP, FF	FCC	TR	PCM	EPCM	AR	EDM	NR
importDimension							✓	
importEJDimensionMembers		✓						
importJobConsole	✓	✓	✓		✓			
importLibraryArtifact								✓
importLibraryDocument	✓	✓	✓		✓			
importLogoImage						✓		
importMapping	✓	✓	✓	✓	✓	✓		
importMetadata	✓	✓	✓		✓			
importOwnershipData		✓	✓					
importPreMappedBalances						✓		
importPreMappedTransactions						✓		
importProfiles						✓		
importRates						✓		
importRCAttributeValues						✓		
importReconciliationAttributes						✓		
importSDMDimensionMembers		✓	✓					
importSnapshot	✓	✓	✓	✓	✓	✓	✓	
importSupplementalCollectionData		✓	✓					
importSupplementalData		✓	✓					
importTemplate				✓				
importTMAttributeValues						✓		
importValidIntersections	✓	✓	✓		✓			
invalidLoginReport	✓	✓	✓	✓	✓	✓	✓	✓
listBackups	✓	✓	✓	✓	✓	✓	✓	✓
listFiles	✓	✓	✓	✓	✓	✓	✓	✓
loadData				✓				
loadDimData				✓				
loadDimensionViewpoint							✓	
loadViewpoint							✓	
login	✓	✓	✓	✓	✓	✓	✓	✓
logout	✓	✓	✓	✓	✓	✓	✓	✓
maskData	✓	✓	✓		✓			
mergeDataSlices	✓				✓			
mergeSlices				✓				
optimizeASOCube				✓				
programDocumentationReport				✓				
provisionReport	✓	✓	✓	✓	✓	✓	✓	✓
purgeArchivedTmTransactions						✓		
purgeTmTransactions						✓		
recomputeOwnershipData		✓	✓					

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 2-1 (suite) Toutes les commandes EPM Automate

Nom de commande	PLN, SWP, SP, FF	FCC	TR	PCM	EPCM	AR	EDM	NR
recreate	✓	✓	✓	✓	✓	✓	✓	✓
refreshCube	✓	✓	✓		✓			
refreshReportPackageDatasource								✓
removeUserFromGroups	✓	✓	✓	✓	✓	✓	✓	✓
removeUsers	✓	✓	✓	✓	✓	✓	✓	✓
removeUsersFromGroup	✓	✓	✓	✓	✓	✓	✓	✓
removeUsersFromTeam		✓	✓			✓		
renameSnapshot	✓	✓	✓	✓	✓	✓	✓	
replay	✓	✓	✓	✓	✓	✓	✓	✓
resetService	✓	✓	✓	✓	✓	✓	✓	✓
restoreBackup	✓	✓	✓	✓	✓	✓	✓	✓
restructureCube	✓	✓	✓					
roleAssignmentAuditReport	✓	✓	✓	✓	✓	✓	✓	✓
roleAssignmentReport	✓	✓	✓	✓	✓	✓	✓	✓
runAutoAlert						✓		
runAutomatch						✓		
runBatch	✓	✓	✓	✓	✓	✓		
runBusinessRule	✓	✓	✓		✓			
runCalc				✓				
runComplianceReport						✓		
runDailyMaintenance	✓	✓	✓	✓	✓	✓	✓	✓
runDataRule	✓	✓	✓	✓	✓	✓		
runDMReport	✓	✓	✓	✓	✓	✓		
runIntegration	✓	✓	✓	✓	✓	✓		
runIntercompanyMatchingReport		✓						
runMatchingReport						✓		
runPipeline	✓	✓	✓		✓	✓		
runPlanTypeMap	✓							
runRuleSet	✓	✓	✓		✓			
runSupplementalDataReport		✓	✓					
runTaskManagerReport		✓	✓		✓			
sendMail	✓	✓	✓	✓	✓	✓	✓	✓
setApplicationAdminMode	✓	✓	✓		✓	✓		
setDailyMaintenanceStartTime	✓	✓	✓	✓	✓	✓	✓	✓
setDemoDates		✓	✓		✓	✓		
setEJJournalStatus		✓						
setEncryptionKey	✓	✓	✓	✓	✓	✓	✓	✓
setEssbaseQryGovExecTime	✓	✓	✓	✓	✓			
setIdleSessionTimeout	✓	✓	✓	✓	✓	✓	✓	✓
setIPAllowlist (désapprouvé)	✓	✓	✓	✓	✓	✓	✓	✓

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 2-1 (suite) Toutes les commandes EPM Automate

Nom de commande	PLN, SWP, SP, FF	FCC	TR	PCM	EPCM	AR	EDM	NR
setManualDataAccess	✓	✓	✓	✓	✓	✓	✓	✓
setPeriodStatus						✓		
setRestrictedDataAccess	✓	✓	✓	✓	✓	✓	✓	✓
setSubstVars	✓	✓	✓		✓			
setVirusScanOnFileUploads	✓	✓	✓	✓	✓	✓	✓	✓
simulateConcurrentUsage	✓	✓	✓					
skipUpdate	✓	✓	✓	✓	✓	✓	✓	✓
snapshotCompareReport	✓	✓	✓		✓			
sortMember	✓				✓			
unassignRole	✓	✓	✓	✓	✓	✓	✓	✓
updateGuidedLearningSettings	✓	✓	✓		✓			
updateUsers	✓	✓	✓	✓	✓	✓	✓	✓
upgrade	✓	✓	✓	✓	✓	✓	✓	✓
uploadFile	✓	✓	✓	✓	✓	✓	✓	✓
userAuditReport (désapprouvé)	✓	✓	✓	✓	✓	✓	✓	✓
userGroupReport	✓	✓	✓	✓	✓	✓	✓	✓
userLoginReport	✓	✓	✓	✓	✓	✓	✓	✓
validateConsolidationMetadata		✓						
validateModel					✓			

- * Cette commande est uniquement prise en charge si les cubes Oracle Essbase hybrides sont activés dans l'application. Strategic Workforce Planning et Sales Planning ne prennent pas en charge Essbase en mode hybride. Cette commande n'est pas prise en charge pour FreeForm.
- ** Cette commande n'est pas prise en charge pour FreeForm.

Abréviations

- PLN : Planning (y compris les modules Planning)
- FF : FreeForm
- SWP : Strategic Workforce Planning
- SP : Sales Planning
- FCC : Financial Consolidation and Close
- TR : Tax Reporting
- PCM : Profitability and Cost Management
- EPCM : Enterprise Profitability and Cost Management
- AR : Account Reconciliation
- EDM : Oracle Fusion Cloud Enterprise Data Management
- NR : Narrative Reporting

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Commandes EPM Automate

Cette section détaille chaque commande EPM Automate. Les informations disponibles pour chaque commande incluent les services pouvant utiliser la commande, la syntaxe de la commande et des exemples.

addUsers

Crée un lot d'utilisateurs dans un domaine d'identité à l'aide d'un fichier CSV (délimité par des virgules) ANSI ou UTF-8 chargé vers l'environnement. Indique également aux nouveaux utilisateurs leur nom d'utilisateur et leur mot de passe temporaire.

Un administrateur de service utilise la commande uploadFile pour charger des fichiers dans un environnement. Toutes les colonnes du fichier CSV sont obligatoires. Cette commande valide la valeur indiquée dans chaque colonne d'une définition et affiche des messages d'erreur qui identifient chaque valeur manquante ou non valide. Le format de fichier CSV est le suivant :

```
First Name,Last Name,Email,User Login
Jane,Doe,jane.doe@example.com,jdoe
John,Doe,john.doe@example.com,john.doe@example.com
```

Reportez-vous à la section Import d'un lot de comptes utilisateur du guide *Mise en route d'Oracle Cloud* pour obtenir une description détaillée du format de fichier CSV.

La valeur correspondant à `User Login` indiquée dans le fichier d'import ne respecte pas la casse. Par exemple, la valeur `John.doe@example.com` est considérée comme identique à `John.Doe@example.com`, ou à toute autre variation de la casse.

Si une définition d'utilisateur dans le fichier CSV correspond à un compte utilisateur existant dans le domaine d'identité, aucune modification n'est apportée à ce compte. Cette commande crée des comptes uniquement pour les nouveaux utilisateurs dont les informations de compte sont incluses dans le fichier. Les comptes utilisateur étant communs à tous les environnements pris en charge par un domaine d'identité, les nouveaux utilisateurs sont disponibles dans tous les environnements qui partagent le même domaine d'identité.

Une fois l'exécution terminée, la commande imprime les informations relatives à chaque entrée en échec dans la console. Reportez-vous à ces informations afin de comprendre pourquoi l'exécution de la commande a échoué pour certaines entrées dans le fichier CSV.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de domaine d'identité et tout rôle prédéfini

Syntaxe

`epmautomate addUsers` *`FILE_NAME`* `[userPassword=`*`PASSWORD`*`] [resetPassword=true|false]`
où :

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- *FILE_NAME* est le nom du fichier CSV contenant les informations de l'utilisateur. Le fichier d'entrée contenant des caractères multi-octets doit utiliser l'encodage de caractères UTF-8.
- `userPassword` (facultatif) indique le mot de passe par défaut pour tous les nouveaux utilisateurs qui sont créés dans le domaine d'identité. S'il est spécifié, ce mot de passe doit respecter les exigences minimales en matière de mot de passe de domaine d'identité. Si le paramètre n'est pas spécifié, un mot de passe temporaire unique est affecté à chaque utilisateur.
  Si la valeur du paramètre `userPassword` est spécifiée, celle-ci est utilisée comme mot de passe pour tous les utilisateurs indiqués dans le fichier CSV. L'affectation d'un même mot de passe à tous les utilisateurs peut être utile si vous créez des utilisateurs uniquement à des fins de test. Si vous créez de véritables utilisateurs Oracle Fusion Cloud Enterprise Performance Management et que vous voulez affecter un mot de passe spécifique à chaque utilisateur, utilisez cette commande sans indiquer de valeur pour le paramètre facultatif `userPassword`.
- `resetPassword` (facultatif) indique si les nouveaux utilisateurs doivent modifier le mot de passe lors de la première connexion. La valeur par défaut est `true`. A moins que ce paramètre facultatif ne soit défini sur `false`, les nouveaux utilisateurs sont obligés de modifier le mot de passe lors de la première connexion.
  Cette commande envoie à chaque nouvel utilisateur un courriel contenant des informations sur son compte (nom d'utilisateur et mot de passe) si `resetPassword` est défini sur `true`. Si `resetPassword` est défini sur `false`, le courriel n'est pas envoyé. Si vous définissez `resetPassword` sur false, vous devez indiquer `userPassword`. Sinon, un mot de passe temporaire unique sera affecté à chaque utilisateur. Néanmoins, dans la mesure où aucun courriel n'est envoyé, les utilisateurs ne connaîtront pas leur mot de passe et ne pourront pas se connecter.

Exemples

- Ajout d'utilisateurs de test dans le domaine d'identité disposant du même mot de passe sans demande de modification du mot de passe :
  ```
  epmautomate addUsers user_file.CSV userPassword=Example@Pwd12
  resetPassword=false
  ```
- Ajout d'utilisateurs dans le domaine d'identité avec un mot de passe temporaire et demande de modification du mot de passe :
  ```
  epmautomate addUsers user_file.CSV
  ```

addUsersToGroup

Ajoute un lot d'utilisateurs à un groupe existant dans Contrôle d'accès à l'aide d'un fichier CSV ANSI ou UTF-8 qui a été chargé vers l'environnement.

Un administrateur de service utilise la commande uploadFile pour charger des fichiers dans un environnement. La valeur de connexion utilisateur ne respecte pas la casse. Le format de fichier est le suivant :

```
User Login
jdoe
john.doe@example.com
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Remarque**
>
> L'utilisateur est ajouté au groupe uniquement si les deux conditions suivantes sont remplies :
>
> - La valeur de connexion utilisateur incluse dans le fichier existe dans le domaine d'identité qui gère l'environnement. Les valeurs de connexion utilisateur ne respectent pas la casse.
> - Un rôle prédéfini a été affecté à l'utilisateur dans le domaine d'identité.

Une fois l'exécution terminée, la commande imprime les informations relatives à chaque entrée en échec dans la console. Reportez-vous à ces informations afin de comprendre pourquoi l'exécution de la commande a échoué pour certaines entrées dans le fichier CSV.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Enterprise Profitability and Cost Management, Narrative Reporting, Strategic Workforce Planning et Sales Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer

Syntaxe

`epmautomate addUsersToGroup` *`FILE_NAME GROUP_NAME`* où :

- *`FILE_NAME`* correspond au nom d'un fichier CSV contenant les noms de connexion des utilisateurs à affecter à un groupe dans Contrôle d'accès.
- *`GROUP_NAME`* correspond au nom d'un groupe existant dans Contrôle d'accès. Cette valeur ne respecte pas la casse.

Exemple

```
epmautomate addUsersToGroup user_file.CSV example_group
```

addUsersToTeam

Ajoute les utilisateurs Oracle Fusion Cloud Enterprise Performance Management répertoriés dans un fichier CSV à une équipe existante.

Si un utilisateur figurant dans le fichier CSV est déjà membre de l'équipe, cette commande l'ignore. Les valeurs figurant dans ce fichier ne respectent pas la casse. Un administrateur de service utilise la commande uploadFile pour charger ce fichier dans l'environnement. Le format de fichier CSV est le suivant :

```
User Login, primary_user
jdoe, yes
jane.doe@example.com,no
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Remarque**
>
> Par défaut, un utilisateur principal a pour rôle d'effectuer les tâches affectées à l'équipe.

Champ d'application

Financial Consolidation and Close, Tax Reporting et Account Reconciliation.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Equipes - Gérer
- Tout rôle prédéfini et rôle d'application Utilisateurs - Gérer

Syntaxe

`epmautomate addUsersToTeam` *`FILE TEAM_NAME`* où :

- `FILE` identifie un fichier CSV au format UTF-8 contenant la liste des ID de connexion des utilisateurs à ajouter à l'équipe. Avant d'exécuter cette commande, utilisez la commande uploadFile pour charger les fichiers vers un environnement.
- `TEAM_NAME` identifie un nom d'équipe tel qu'il est défini dans Contrôle d'accès. Cette valeur ne respecte pas la casse.

Exemple

```
epmautomate addUsersToTeam example_users.csv example_team
```

addUserToGroups

Ajoute un utilisateur en tant que membre des groupes Contrôle d'accès identifiés dans un fichier CSV codé en ANSI ou UTF-8.

Un administrateur de service utilise la commande uploadFile pour charger des fichiers dans un environnement. Le format de fichier est le suivant :

```
Group Name
Group1
Group2
```

Les valeurs de nom de groupe ne respectent pas la casse.

Une fois l'exécution terminée, la commande imprime les informations relatives à chaque entrée en échec dans la console. Reportez-vous à ces informations afin de comprendre pourquoi l'exécution de la commande a échoué pour certaines entrées dans le fichier CSV.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Strategic Workforce Planning et Sales Planning.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer

Syntaxe

`epmautomate addUserToGroups` *`FILE_NAME User_Login`*, où :

- `FILE_NAME` est le nom d'un fichier CSV contenant les noms de groupe Contrôle d'accès auxquels affecter l'utilisateur.
- `User_Login` est l'ID de connexion d'un utilisateur Oracle Fusion Cloud Enterprise Performance Management à affecter à des groupes Contrôle d'accès. Cet ID de connexion utilisateur, qui ne respecte pas la casse, doit exister dans le domaine d'identité qui gère l'environnement et doit être affecté à un rôle prédéfini.

Exemple

```
epmautomate addUserToGroups groups.CSV jdoe@examle.com
```

applicationAdminMode

Passe l'application en mode d'administration afin que l'accès à l'application soit limité aux administrateurs de service uniquement.

Cette commande est utile pour empêcher les utilisateurs de travailler sur l'application lorsque les administrateurs de service effectuent des opérations d'administration. L'application reste en mode d'administration jusqu'à ce que vous rétablissiez le paramétrage initial afin de permettre à tous les utilisateurs d'y accéder.

> **Remarque**
>
> Cette commande a été désapprouvée mais n'a pas été enlevée d'EPM Automate. Oracle vous recommande d'utiliser la commande setApplicationAdminMode à la place.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate applicationAdminMode` *`VALUE`* où `VALUE` détermine si l'application doit passer en mode d'administration. Les valeurs acceptables sont les suivantes :

- `true` pour passer l'application en mode d'administration
- `false` pour repasser l'application en mode normal afin que tous les utilisateurs puissent y accéder

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exemples

- Passer l'application en mode d'administration : `epmautomate applicationAdminMode true`
- Repasser l'application en fonctionnement normal : `epmautomate applicationAdminMode false`

applyDataGrants

Actualise les autorisations d'accès aux données, qui contrôlent l'accès aux secteurs de données Oracle Essbase, de sorte qu'elles correspondent aux autorisations d'accès aux données définies dans une application Profitability and Cost Management.

Les autorisations d'accès aux données de niveau utilisateur et groupe que vous effectuez dans l'application Profitability and Cost Management sont automatiquement synchronisées dans Essbase. Utilisez cette commande pour synchroniser l'accès aux données Essbase si vous suspectez une discordance entre l'autorisation de données dans l'application et les filtres dans Essbase.

Le temps requis pour effectuer cette opération dépend de la taille de l'application. Assurez-vous que l'opération d'actualisation des autorisations de données prend fin avant que l'application soit sauvegardée lors de la prochaine fenêtre de maintenance. Oracle vous recommande de planifier cette opération à une heure où les utilisateurs n'utilisent pas l'application, car cette dernière ne doit pas être utilisée pendant que cette opération est en cours.

Champ d'application

Profitability and Cost Management

Rôles requis

Administrateur de service, Super utilisateur

Syntaxe

`epmautomate applyDataGrants` *`APPLICATION_NAME`*, où *`APPLICATION_NAME`* est le nom de l'application Profitability and Cost Management pour laquelle les autorisations d'accès aux données doivent être recréées.

Exemple

```
epmautomate applyDataGrants BksML12
```

archiveTmTransactions

Archive les transactions mises en correspondance, y compris les détails annexes et les détails d'ajustement, égales ou antérieures à une certaine ancienneté. Les transactions mises en correspondance sont enregistrées dans un fichier ZIP.

Utilisez cette commande afin de conserver une taille d'application Account Reconciliation optimale grâce à l'archivage et à la purge des anciennes transactions mises en correspondance en fonction des stratégies de conservation des transactions de votre organisation.

Champ d'application

Account Reconciliation

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service, Super utilisateur, Utilisateur, Visualiseur
Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmautomate archiveTmTransactions matchType age [filterOperator=VALUE] [filterValue=VALUE] [logFilename=FILE_NAME] [filename=FILE_NAME]` où :

- `matchType` est l'identificateur (`TextID`) du type de correspondance dont les transactions mises en correspondance doivent être archivées.
- `age` indique le nombre de jours écoulés depuis la mise en correspondance de la transaction. Toute transaction mise en correspondance antérieure ou égale à cette valeur sera archivée.
- `filterOperator` (facultatif) est l'une des conditions de filtre suivantes permettant d'identifier les comptes qui contiennent des transactions mises en correspondance pour archivage. Cette valeur est associée à `filterValue` pour identifier les comptes dont les transactions mises en correspondance doivent être archivées :
  - `equals`
  - `not_equals`
  - `starts_with`
  - `ends_with`
  - `contains`
  - `not_contains`
- `filterValue` (facultatif) est une valeur de filtre permettant d'identifier les transactions à archiver. Si `filterOperator` est `equals` ou `not_equals`, vous pouvez utiliser une liste de valeurs séparées par des espaces pour indiquer plusieurs valeurs. Par exemple : `filterValue=101-120 filterValue=102-202`. Si plusieurs valeurs sont indiquées, les transactions provenant des comptes correspondant à toute combinaison d'opérateur de filtre et de valeur de filtre sont sélectionnées afin d'être archivées.

  > **Note**
  >
  > Si `filterOperator` et `filterValue` ne sont pas indiqués, toutes les transactions mises en correspondance antérieures ou égales à `age` provenant de tous les comptes pour la valeur `matchType` indiquée sont archivées.

- `logFilename` (facultatif) est le nom d'un fichier journal permettant d'enregistrer des informations à propos de l'activité de commande. Si aucun nom de fichier n'est indiqué, un fichier journal nommé `Archive_Transactions_matchType_JOBID.log` est automatiquement généré.
- `filename` (facultatif) est le nom d'un fichier ZIP destiné à contenir les transactions archivées. S'il n'est pas indiqué, la commande crée par défaut le fichier `Archived_Transactions_matchType_JOBID.zip`. Employez la commande downloadFile pour télécharger ce fichier sur un ordinateur local.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Note**
>
> Cette commande exécute le job d'archivage des transactions Correspondance de transactions en fonction des paramètres que vous indiquez. L'ID de job est renvoyé dans la sortie de commande afin de pouvoir l'utiliser facilement avec la commande purgeArchivedTmTransactions. Vous pouvez surveiller le job à partir de la console de jobs.

Exemples

- Archivez les anciennes transactions mises en correspondance sans utiliser les filtres, mais à l'aide d'un fichier journal et d'un nom de fichier ZIP personnalisés :
  `epmautomate archiveTmTransactions cashrecon 180 logFile=tmlogs.log filename=trans.zip`
- Archivez les anciennes transactions mises en correspondance à l'aide des filtres :
  - `epmautomate archiveTmTransactions cashrecon 180 filterOperator=equals filterValue=101-120 FilterValue=102-202`
  - `epmautomate archiveTmTransactions cashrecon 180 filterOperator=contains filterValue=11`

assignRole

Affecte un rôle à des utilisateurs (y compris l'utilisateur qui exécute cette commande) dont les ID de connexion sont inclus dans un fichier CSV encodé en ANSI ou UTF-8. Cette commande permet d'affecter des utilisateurs à un rôle prédéfini ou à un rôle d'application.

Avant d'utiliser cette commande, un administrateur de service utilise la commande uploadFile pour charger le fichier d'entrée dans un environnement. Le format de fichier est le suivant :

```
User Login
jane.doe@example.com
jdoe
```

Reportez-vous à la section Affectation d'un rôle à plusieurs utilisateurs dans le guide *Mise en route d'Oracle Cloud*.

> **Remarque**
>
> - Les valeurs de connexion utilisateur incluses dans le fichier ne respectent pas la casse.
> - Placez entre guillemets doubles les noms de rôle qui contiennent un espace.

Une fois l'exécution terminée, la commande imprime les informations relatives à chaque entrée en échec dans la console. Reportez-vous à ces informations afin de comprendre pourquoi l'exécution de la commande a échoué pour certaines entrées dans le fichier CSV.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Strategic Workforce Planning et Sales Planning.

Rôles requis

Pour affecter des rôles prédéfinis :

- Pour affecter des rôles prédéfinis :
  - Administrateur de service
  - Administrateur de domaine d'identité et tout rôle prédéfini
- Pour affecter des rôles d'application :
  - Administrateur de service
  - Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer

Syntaxe

`epmautomate assignRole` *`FILE_NAME ROLE`* où :

- *`FILE_NAME`* est le nom d'un fichier CSV contenant des ID de connexion utilisateur. Indiquez l'extension CSV en minuscules.
- `ROLE` est l'une des valeurs suivantes. Cette valeur ne respecte pas la casse :
  - Si vous affectez des rôles de domaine d'identité prédéfinis à des utilisateurs, `ROLE` doit identifier un rôle prédéfini applicable au service. Reportez-vous à la section Présentation des rôles prédéfinis du guide *Guide de mise en route pour les administrateurs*.
    Pour obtenir une description de ces rôles, reportez-vous à la section Gestion des affectations de rôle au niveau de l'application du guide *Administration du contrôle d'accès*
  - Si vous affectez des rôles d'application à des utilisateurs, `ROLE` doit identifier un rôle appartenant à l'application dans l'environnement en cours. Les rôles d'application sont répertoriés dans l'onglet **Rôles** de Contrôle d'accès. Afin d'obtenir une description des rôles d'application pour chaque processus métier, reportez-vous aux rubriques suivantes dans le guide *Administration du contrôle d'accès pour Oracle Enterprise Performance Management Cloud* :
    * Account Reconciliation
    * Enterprise Profitability and Cost Management
    * Planning, FreeForm, Financial Consolidation and Close et Tax Reporting
    * Profitability and Cost Management
    * Oracle Enterprise Data Management
    * Narrative Reporting

Exemples

- Affectation d'utilisateurs à un rôle de domaine d'identité prédéfini :
  `epmautomate assignRole admin_role_file.csv "Administrateur de service"`
- Affection d'utilisateurs à un rôle d'application :
  `epmautomate assignRole example_file.csv "Gestionnaire d'accès à la liste des tâches"`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

autoPredict

Génère des prévisions de performances en fonction d'une définition de prévision automatique existante dans Planning ou les modules Planning.

Cette commande lance un job qui utilise les données historiques pour chaque membre identifié dans la définition de prévision automatique indiquée dans l'application. Pour obtenir des informations détaillées sur les applications qui utilisent la fonctionnalité de prévision automatique et sur la configuration des prévisions, reportez-vous à la section Configuration de l'exécution automatique des prévisions grâce à la prévision automatique du guide *Administration de Planning*.

Champ d'application

Planning, modules Planning, si les cubes Oracle Essbase hybrides sont activés dans l'application.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate autoPredict` *`PREDICTION_DEFINITION`* `[forceRun=true|false] [paginatedDim=`*`DIMENSION_NAME`*`]` où :

- *`PREDICTION_DEFINITION`* est le nom d'une définition de prévision automatique disponible dans l'application.
- `forceRun` (facultatif) indique si la prévision doit être exécutée si la définition sous-jacente n'a pas été modifiée après l'exécution initiale. La valeur par défaut est `false`. Définissez la valeur de ce paramètre sur `true` pour exécuter le job de prévision automatique même si aucune modification n'a été apportée à la définition du job. Utilisez la valeur par défaut (`false`) pour exécuter la prévision une fois, lors de la toute première exécution du job.
- `paginatedDim` (facultatif) indique une dimension utilisée pour accélérer le job de prévision automatique en exécutant des prévisions en parallèle dans des threads distincts. Afin que ces threads parallèles soient efficaces, indiquez une dimension qui permettra de répartir les données à parts égales pour chaque thread de prévision.

Exemple

```
epmautomate autoPredict ASOtoBSO forceRun=true paginatedDim=Entity
```

calculateModel

Exécute le processus de calcul dans les applications Enterprise Profitability and Cost Management.

Champ d'application

Enterprise Profitability and Cost Management

Rôles requis

Administrateur de service

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Syntaxe

> **Note**
>
> Si des variables de substitution existent, vous pouvez les utiliser à la place des valeurs de paramètre réelles. Pour obtenir des informations détaillées, reportez-vous à la section Utilisation de variables de substitution avec certaines commandes Enterprise Profitability and Cost Management.

`epmautomate calculateModel` *`POV_NAME MODEL_NAME EXECUTION_TYPE`* `[povDelimiter=`*`DELIMITER`*`] [optimizeForReporting=true|false] [captureDebugScripts=true|false] [comment=`*`COMMENT`*`] [PARAMETER=`*`VALUE`*`]`, où :

- *`POV_NAME`* est le nom du PDV de données à calculer. Pour calculer plusieurs PDV, répertoriez les noms de PDV séparés par une virgule. N'utilisez aucun autre séparateur pour séparer les noms de PDV. Mettez la liste des noms de PDV entre guillemets lorsque les noms de membre contiennent des espaces.
- *`MODEL_NAME`* est le nom du modèle à calculer. Mettez le nom de modèle entre guillemets s'il contient des espaces.
- *`EXECUTION_TYPE`* correspond à l'une des valeurs suivantes, qui identifie le type d'exécution de règle.
  - `ALL_RULES`, pour utiliser toutes les règles afin de calculer le PDV.
    Si vous indiquez cette valeur, ne précisez pas les paramètres d'exécution liés à une seule règle ou à un sous-ensemble de règles, tels que `rulesetSeqNumStart`, `rulesetSeqNumEnd` et `ruleName`.
  - `RULESET_SUBSET`, pour utiliser un sous-ensemble d'un ensemble de règles afin de calculer le PDV.
    Si vous utilisez cette valeur, vous devez indiquer les valeurs `rulesetSeqNumStart` et `rulesetSeqNumEnd` en tant que paramètres d'exécution.
  - `SINGLE_RULE`, pour exécuter une règle spécifique afin de calculer le PDV.
    Si vous utilisez cette valeur, vous devez uniquement indiquer `ruleName` en tant que paramètre d'exécution.
  - `RUN_FROM_RULE`, pour exécuter des calculs sur un PDV à partir d'une règle spécifique.
    Si vous utilisez cette valeur, vous devez uniquement indiquer `ruleName` en tant que paramètre d'exécution.
  - `STOP_AFTER_RULE`, pour arrêter de calculer le PDV une fois qu'une règle spécifique a terminé les calculs.
    Si vous utilisez cette valeur, vous devez uniquement indiquer `ruleName` en tant que paramètre d'exécution.
- `povDelimiter` (facultatif) est le séparateur utilisé dans les valeurs de PDV. Le séparateur par défaut est :: (deux-points doubles). Le séparateur doit être placé entre guillemets doubles. Seuls les séparateurs suivants sont pris en charge :
  - _ (trait de soulignement)
  - # (dièse)
  - ~ (tilde)
  - % (pourcentage)
  - ; (point-virgule)

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

  - : (deux-points)
  - - (tiret)
- `optimizeForReporting=true|false` (facultatif) indique si les calculs doivent être exécutés avec ou sans optimisation à des fins de reporting. La valeur par défaut est `false`. Définissez cette valeur sur `false` pour gagner du temps lors du traitement en ignorant l'étape de création d'agrégation, par exemple lors de l'exécution d'une règle unique ou d'une série séquentielle de PDV. En cas d'exécution de plusieurs jobs de calcul simultanés, définissez `optimizeForReporting=true` pour tous les jobs. Le dernier job terminé effectue l'agrégation, ce qui évite un traitement redondant et un ralentissement de l'exécution des jobs.
- `captureDebugScripts=true|false` (facultatif) identifie s'il est nécessaire de générer des scripts de débogage dans la boîte de réception. Oracle peut avoir besoin de ces scripts pour résoudre les problèmes de calcul. La valeur par défaut est `false`.
- `comment="COMMENT"` (facultatif) indique un commentaire à propos du processus entre guillemets doubles.
- `PARAMETER=`*`VALUE`* (facultatif) indique les paramètres d'exécution et leurs valeurs pour exécuter le calcul. Indiquez autant de paires paramètre-valeur que le processus l'exige. Paramètres valides et leurs valeurs :
  - `rulesetSeqNumStart` indique le numéro de séquence de la première règle de l'ensemble de règles à exécuter. Valide uniquement si `EXECUTION_TYPE=RULESET_SUBSET` est utilisé.
  - `rulesetSeqNumEnd` indique le numéro de séquence de la dernière règle de l'ensemble de règles à exécuter. Valide uniquement si `EXECUTION_TYPE=RULESET_SUBSET` est utilisé.
  - `ruleName` indique le nom de la règle à exécuter. Mettez la valeur entre guillemets si elle contient un espace. Valide uniquement si la valeur du paramètre `EXECUTION_TYPE` est définie sur `SINGLE_RULE`, `RUN_FROM_RULE` ou `STOP_AFTER_RULE`.
  - `clearCalculatedData=true|false` indique si les calculs existants doivent être effacés. La valeur par défaut est `false`.
  - `executeCalculations=true|false` indique si des calculs doivent être exécutés. La valeur par défaut est `false`.

> **Note**
>
> Les valeurs de paramètre (`true` et `false`) doivent être indiquées entièrement en minuscules.

Exemples

- Exécutez toutes les règles pour calculer un PDV unique :
```
epmautomate calculateModel FY22::Jan::Actual::Working "10 Actuals Allocation
Process" ALL_RULES clearCalculatedData=true executeCalculations=true
optimizeForReporting=true comment="Running all rules to calculate a POV"
```
- Exécutez toutes les règles pour calculer plusieurs PDV à l'aide d'un séparateur personnalisé :
```
epmautomate calculateModel
"FY22_Jan_Actual_Working,FY22_Feb_Actual_Working,FY22_Mar_Actual_Working" "10
Actuals Allocation Process" ALL_RULES clearCalculatedData=true
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
executeCalculations=true optimizeForReporting=true captureDebugScripts=true
comment="Runing calculation for multiple POVs" povDelimiter="_"
```

- Exécutez une plage d'ensembles de règles pour calculer un PDV :
```
epmautomate calculateModel FY22::Jan::Actual::Working "10 Actuals Allocation
Process" RULESET_SUBSET rulesetSeqNumStart=10 rulesetSeqNumEnd=20
clearCalculatedData=true executeCalculations=true comment="Running a subset of
rule sets"
```
- Exécutez une règle spécifique pour calculer un PDV :
```
epmautomate calculateModel FY22::Jan::Actual::Working "10 Actuals Allocation
Process" SINGLE_RULE ruleName="Rent and Utilities Reassignment"
clearCalculatedData=true executeCalculations=true comment="Running a specific
rule"
```

clearCube

Supprime des données spécifiques dans les cubes d'entrée et de reporting à l'aide des paramètres indiqués dans un job de type `clearcube`.

Cette commande ne supprime pas la définition d'application dans les tables relationnelles de l'application. Reportez-vous à la section Effacement des cubes du guide *Administration de Planning*.

Champ d'application

Planning, modules Planning, FreeForm, Enterprise Profitability and Cost Management, Strategic Workforce Planning Cloud et Sales Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate clearCube` *`JOB_NAME`*, où *`JOB_NAME`* est le nom d'un job défini dans l'application.

Exemple

```
epmautomate clearCube ClearPlan1
```

clearDataByPointOfView

Efface les données relatives à un PDV spécifique pour un cube Enterprise Profitability and Cost Management.

Champ d'application

Enterprise Profitability and Cost Management

Rôles requis

Administrateur de service

Syntaxe

`epmAutomate clearDataByPointOfView` *`POV_NAME`* `[cubeName=`*`CUBE_NAME`*`]` `[`*`PARAMETER=VALUE`*`]` où :

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- *POV_NAME* est le nom d'un PDV dans l'application.
- `cubeName` (facultatif) est le nom du cube dans lequel les données doivent être effacées. La valeur par défaut est `PCM_CLC`.
- `PARAMETER=`*VALUE* indique des paramètres d'exécution facultatifs et leurs valeurs. Indiquez autant de paires paramètre-valeur que le processus l'exige. Paramètres valides et leurs valeurs :
  - `povDelimiter` est le séparateur utilisé dans les valeurs de PDV. La valeur par défaut est `::` (deux-points doubles). Cette valeur doit être placée entre guillemets doubles. Exemple : `povDelimiter="_"`.
    Outre le séparateur par défaut, seuls ceux-ci sont pris en charge : _ (trait de soulignement), # (dièse), ~ (tilde), % (pourcentage), ; (point-virgule), : (deux-points), - (tiret).
  - `clearInput=true|false` indique si des données d'entrée doivent être effacées. La valeur par défaut est `false`.
  - `clearAllocatedValues=true|false` indique si des valeurs allouées doivent être effacées. La valeur par défaut est `false`.
  - `clearAdjustmentValues=true|false` indique si des valeurs d'ajustement doivent être effacées. La valeur par défaut est `false`.

> **Note**
>
> * Les valeurs de paramètre (`true` ou `false`) doivent être indiquées entièrement en minuscules.
> * Au moins l'un des paramètres `clearInput`, `clearAllocatedValues` et `clearAdjustmentValues` doit être défini sur `true`.

Exemples

- Effacez les données d'un PDV dans le cube PCM_CLC par défaut à l'aide du séparateur de PDV par défaut :
```
epmAutomate clearDataByPointOfView FY21::Jan::Actual::Working clearInput=true
clearAllocatedValues=true clearAdjustmentValues=true
```
- Effacez les données de saisie et les valeurs affectées d'un PDV dans un cube spécifique à l'aide d'un séparateur de PDV personnalisé :
```
epmAutomate clearDataByPointOfView FY21_Jan_Actual_Working cubeName=PCM_REP
povDelimiter="_" clearInput=true clearAllocatedValues=true
```
- Effacez les données de saisie d'un PDV dans un cube spécifique à l'aide d'un séparateur de PDV personnalisé :
```
epmAutomate clearDataByPointOfView FY21:Jan:Actual:Working cubeName=PCM_REP
povDelimiter=":" clearInput=true
```

clearDataByProfile

Efface les données des éléments (par exemple, les régions) identifiés dans un profil d'effacement de données défini dans Financial Consolidation and Close et Tax Reporting.

Champ d'application

Financial Consolidation and Close, Tax Reporting

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate clearDataByProfile` *`PROFILE_NAME`*, où *`PROFILE_NAME`* est le nom d'un profil d'effacement de données.

Exemple

```
epmautomate clearDataByProfile clearDataProfile_01
```

clearPOV

Efface les données et les artefacts de modèle d'une combinaison de points de vue (PDV) ou d'une région de données du PDV dans une application Profitability and Cost Management.

Champ d'application

Profitability and Cost Management

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

> **Remarque**
>
> Si des variables de substitution existent, vous pouvez les utiliser à la place des valeurs de paramètre réelles. Pour obtenir des informations détaillées, reportez-vous à la section Utilisation de variables de substitution avec certaines commandes Enterprise Profitability and Cost Management.

`epmautomate clearPOV` *`APPLICATION_NAME POV_NAME [QUERY_NAME]`* `PARAMETER=`*`VALUE`* `stringDelimiter=`*`"DELIMITER"`* où :

- *`APPLICATION_NAME`* est le nom d'une application Profitability and Cost Management.
- *`POV_NAME`* est un PDV dans l'application. Cette valeur est obligatoire.
- `QUERY_NAME` (facultatif) est le nom d'une requête exactement tel que défini dans Profitability and Cost Management. Si elle est indiquée, cette requête sera utilisée pour effacer la région de données dans le PDV.

> **Remarque**
>
> Si vous indiquez un nom de requête, vous devez définir la valeur de tous les paramètres d'exécution (voir ci-dessous) sur False.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `PARAMETER=VALUE` indique les paramètres d'exécution et leurs valeurs pour effacer le PDV. Indiquez autant de paires paramètre-valeur que le processus l'exige. Paramètres valides (au moins l'un d'entre eux est obligatoire) et leurs valeurs :
  - `isManageRule=true|false` indique si les règles doivent être effacées
  - `isInputData=true|false` indique si des données d'entrée doivent être effacées
  - `isAllocatedValues=true|false` indique si des valeurs d'allocation doivent être effacées
  - `isAdjustmentValues=true|false` indique si des valeurs d'ajustement doivent être effacées

> **Remarque**
>
> Les valeurs de paramètre (`true` ou `false`) doivent être indiquées entièrement en minuscules.
>
> Pour effacer les régions de données dans un PDV (si `QUERY_NAME` est indiqué), vous devez définir la valeur des paramètres d'exécution (`isManageRule`, `isInputData`, `isAllocatedValues` et `isAdjustmentValues`) sur `false`.

- `stringDelimiter="DELIMITER"` indique le séparateur utilisé dans les valeurs de PDV. Le séparateur doit être placé entre guillemets doubles. La valeur par défaut est _ (trait de soulignement).

Exemples

- Effacement de tous les artefacts et des données de modèle d'un PDV : `epmautomate clearPOV BksML12 2012_Jan_Actual isManageRule=true isInputData=true isAllocatedValues=true isAdjustmentValues=true stringDelimiter="_"`
- Effacement d'une région de données dans un PDV : `epmautomate clearPOV BksML12 2012_Jan_Actual queryName=BksML12_2012_Jan_clear_query isManageRule=false isInputData=false isAllocatedValues=false isAdjustmentValues=false stringDelimiter="_"`

cloneEnvironment

Clone l'environnement en cours et éventuellement les artefacts de domaine d'identité (utilisateurs et affectations de rôle prédéfini), les enregistrements Data Management, les enregistrements d'audit, les enregistrements de la console de jobs, les propriétés d'application, le contenu des boîtes de réception et d'envoi, ainsi que les instantanés stockés. Cette commande constitue une alternative à l'utilisation de la fonctionnalité Cloner l'environnement dans Migration.

Lancez le clonage après la maintenance quotidienne planifiée des environnements source et cible. Si la maintenance quotidienne de l'environnement source commence alors que le clonage est en cours, le processus de clonage prendra fin. Le processus de clonage sur l'environnement cible n'est pas concerné, même si le clonage est en cours au moment où la maintenance quotidienne démarre. Dans ce scénario, la maintenance quotidienne sera exécutée une fois le clonage terminé.

Si le clonage de l'environnement prend beaucoup de temps, replanifiez l'heure de début de la maintenance quotidienne sur l'environnement source afin d'éviter que le processus de clonage ne soit prenne fin. Pour savoir comment réinitialiser l'heure de début de la maintenance quotidienne, consultez les sources d'information suivantes.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- setDailyMaintenanceStartTime
- Gestion de la maintenance quotidienne dans le *Guide de mise en route pour les administrateurs*
- Affichage et définition de l'heure de la fenêtre de maintenance quotidienne dans le guide *API REST pour Oracle Fusion Cloud EPM*

> **Note**
>
> - **Data Management :** le clonage des enregistrements Data Management peut prendre un certain temps si les tables intermédiaires contiennent beaucoup d'enregistrements. De même, le clonage du contenu des boîtes de réception et d'envoi, ainsi que des instantanés stockés, peut prendre un temps considérable, surtout s'ils contiennent une grande quantité de données.
> - **Planning :** le clonage peut échouer si le processus métier Planning contient un membre de période prédéfini renommé qui a été remplacé par un membre de période libre. Par exemple, vous avez renommé le membre de période *YearTotal* prédéfini en *unused_YearTotal*, puis vous avez ajouté un membre de période d'un autre type avec le nom du membre prédéfini d'origine (*YearTotal* dans cet exemple). Dans ce scénario, le clonage de l'environnement peut échouer.

Pour obtenir des informations détaillées à ce sujet, reportez-vous à la section Clonage d'environnements Cloud EPM du guide *Administration de Migration*.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Syntaxe

`epmAutomate cloneEnvironment` *`TARGET_USERNAME TARGET_PASSWORD TARGET_URL`* `[SnapshotName=`*`NAME`*`] [UsersAndPreDefinedRoles=true|false] [DataManagement=true|false] [appAudit=true|false] [jobConsole=true|false] [storedSnapshotsAndFiles=true|false] [DailyMaintenanceStartTime=true|false] [ApplicationProperties=true|false]`, où :

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Note

- Le paramètre `dataManagement` ne s'applique pas aux environnements Cloud EDM et Narrative Reporting.
  Ne clonez les enregistrements de gestion des données que si les environnements source et cible utilisent la même mise à jour mensuelle, ou si l'environnement cible repose sur une mise à jour plus récente que l'environnement source. Par exemple, vous pouvez cloner les enregistrements de gestion des données 22.01 vers un autre environnement 22.01 ou un environnement 22.02 uniquement.
- Le paramètre `jobConsole` s'applique uniquement à Planning, aux modules Planning, à FreeForm, à Financial Consolidation and Close, à Tax Reporting, à Enterprise Profitability and Cost Management, à Sales Planning et à Strategic Workforce Planning.
- Le paramètre `appAudit` s'applique uniquement à Planning, aux modules Planning, à FreeForm, à Enterprise Profitability and Cost Management, à Sales Planning et à Strategic Workforce Planning.
  Les informations d'audit pour Financial Consolidation and Close et Tax Reporting sont par défaut incluses dans l'instantané.
- Si le paramètre `dataManagement`, `jobConsole` ou `appAudit` n'est pas applicable à un environnement, la commande ignore la valeur indiquée.
- Le paramètre `ApplicationProperties` s'applique uniquement à Account Reconciliation.

- *`TARGET_USERNAME`* est l'ID d'un administrateur de service dans l'environnement cible. Vous devez utiliser le nom d'utilisateur de domaine d'identité cible (pas le nom d'utilisateur d'authentification unique).

  Note

  Pour cloner des affectations d'utilisateur et de rôle, cet utilisateur doit être un administrateur de service disposant du rôle Administrateur de domaine d'identité.

- *`TARGET_PASSWORD`* est le nom et l'emplacement du mot de passe crypté ou du jeton d'actualisation OAuth 2.0 de l'utilisateur identifié par *`TARGET_USERNAME`*.
- *`TARGET_URL`* est l'URL `CLOUD-EPM_BASE_URL` de l'environnement qui deviendra l'environnement cloné.
- *`SnapshotName`* (facultatif) est le nom de l'instantané à utiliser pour le clonage. Cet instantané doit être présent dans l'environnement source. La valeur par défaut est `Artifact Snapshot`, qui utilise le dernier instantané de maintenance pour cloner l'environnement.

  Note

  Il doit s'agir d'un instantané créé par le processus de maintenance quotidienne. Il peut s'agir d'un instantané de ce type copié vers l'environnement source à l'aide de la commande restoreBackup.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- *UsersAndPreDefinedRoles* (facultatif) indique si les utilisateurs et leur affectation de rôle prédéfini doivent être clonés (les groupes de contrôle d'accès sont toujours clonés). La valeur par défaut est `false`.
  Pour que cette option fonctionne, l'utilisateur identifié par *TARGET_USER_NAME* doit avoir le rôle Administrateur de domaine d'identité dans l'environnement cible.

  L'import des utilisateurs et de leur rôle prédéfini peut échouer si un utilisateur qui n'est pas administrateur de domaine d'identité clone un environnement après avoir sélectionné cette case. L'erreur suivante est enregistrée dans le rapport sur le statut de la migration : `Echec de l'import de l'artefact d'annuaire externe <nom_artefact>. L'utilisateur <nom_d'utilisateur> n'est pas autorisé à effectuer cette opération. L'utilisateur doit être doté du rôle Administrateur de domaine d'identité pour effectuer cette opération.`

  - Si vous n'importez pas les utilisateurs et qu'un utilisateur dans l'instantané source n'est pas affecté à un rôle prédéfini dans l'environnement cible, une erreur (`EPMIE-00070 : Utilisateur introuvable lors de l'import des rôles affectés.`) s'affiche.
  - Les affectations du rôle Administrateur de domaine d'identité ne sont pas clonées. Les utilisateurs dotés uniquement du rôle Administrateur de domaine d'identité ne sont pas clonés dans l'environnement cible.
    Les utilisateurs dotés du rôle Administrateur de domaine d'identité et de rôles prédéfinis dans l'environnement source sont clonés, mais seuls les rôles prédéfinis leur seront respectivement affectés dans l'environnement cible. Ces utilisateurs ne seront pas dotés du rôle Administrateur de domaine d'identité dans l'environnement cible.
  - Les modifications apportées aux rôles prédéfinis de l'utilisateur seront mises à jour en fonction des rôles affectés dans l'instantané source. Toutefois, les affectations de rôle dans la cible ne seront pas enlevées pour correspondre à celles de l'instantané source. Par exemple, supposons que `jdoe` se voit affecter le rôle prédéfini de super utilisateur dans l'environnement cible, mais ne dispose que du rôle d'utilisateur dans l'instantané source. Dans ce cas, cette commande affecte à `jdoe` le rôle d'utilisateur et n'enlève pas l'affectation de rôle de super utilisateur dans l'environnement cible.
  - Cette commande ne supprime pas les utilisateurs existants de l'environnement cible s'ils n'existent pas dans l'instantané source. Par exemple, `jdoe` dispose d'un compte dans l'environnement cible, mais ce compte n'est pas présent dans l'instantané source. Dans ce cas, le compte de `jdoe` dans l'environnement cible n'est pas supprimé.
  - Cette commande ajoute des utilisateurs qui n'existent pas dans l'environnement cible. Elle ne met pas à jour les propriétés de l'utilisateur actuel dans l'environnement cible, même si celles-ci sont différentes dans l'instantané source. Par exemple, si l'orthographe du nom de famille de `jdoe` est différente dans l'instantané source et dans l'environnement cible, aucune modification ne sera apportée dans l'environnement cible. Un mot de passe aléatoire est affecté aux nouveaux utilisateurs dans l'environnement cible. Les nouveaux utilisateurs recevront des courriels d'activation de compte les invitant à modifier leur mot de passe.
  - Cette commande ne modifie pas les mots de passe des utilisateurs existants dans l'environnement cible, même s'ils sont différents dans l'instantané source.
- `dataManagement=true|false` (facultatif) clone les enregistrements de gestion des données de l'environnement source dans l'environnement cible. La valeur par défaut est `true`, qui clone les enregistrements de gestion des données. Définissez cette valeur sur `false` si vous ne voulez pas cloner les enregistrements de gestion des données.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `appAudit=true|false` (facultatif) clone les enregistrements d'audit de l'environnement source dans l'environnement cible. La valeur par défaut est `true`, qui clone les données d'audit d'application. Définissez cette valeur sur `false` si vous ne voulez pas cloner les données d'audit d'application dans l'environnement cible.
- `jobConsole=true|false` (facultatif) clone les enregistrements de la console de jobs de l'environnement source dans l'environnement cible. La valeur par défaut est `true`. Définissez cette valeur sur `false` si vous ne voulez pas cloner les enregistrements de la console de jobs.
- `storedSnapshotsAndFiles` (facultatif) indique si la commande doit cloner le contenu de la boîte de réception et de la boîte d'envoi, ainsi que les instantanés stockés. La valeur par défaut est `false`.

> **Note**
>
> Seuls les dossiers de niveau supérieur dans la boîte de réception et la boîte d'envoi sont clonés. Les sous-dossiers ne le sont pas. Si vous devez conserver le contenu des sous-dossiers, sauvegardez ces sous-dossiers sur un ordinateur local, puis chargez-les vers l'environnement cible.

- *`DailyMaintenanceStartTime`* (facultatif) réinitialise l'heure de début de maintenance de l'environnement cible cloné de sorte qu'elle corresponde à celle de l'environnement source. La valeur par défaut est `true`. Pour conserver l'heure de début de maintenance actuelle de l'environnement cible, définissez cette valeur sur `false`.
- `ApplicationProperties` (facultatif) clone les paramètres d'application Account Reconciliation (paramètre d'activation/de désactivation des notifications par courriel, expérience Redwood, thème, nom de processus métier, image de logo et image d'arrière-plan). La valeur par défaut est `true`.
  Pour que l'environnement cible conserve ses paramètres d'application actuels, définissez cette valeur sur `false`.

Exemples

- Clonez l'environnement, les utilisateurs et les affectations de rôle prédéfini, les données d'audit, les enregistrements de la console de jobs et les enregistrements de gestion des données. Modifiez également l'heure de début de maintenance de l'environnement cible de sorte qu'elle corresponde à celle de l'environnement source :
  ```
  epmAutomate cloneEnvironment serviceAdmin Password.epw https://test-cloudpln.pbcs.us1.oraclecloud.com UsersAndPreDefinedRoles=true
  ```
- Clonez l'environnement, y compris les instantanés stockés et le contenu des boîtes de réception et d'envoi, mais pas les utilisateurs et les affectations de rôle prédéfini, les enregistrements de gestion des données, les données d'audit et les enregistrements de la console de jobs, sans modifier l'heure de début de maintenance de l'environnement cible.
  ```
  epmAutomate cloneEnvironment serviceAdmin Password.epw https://test-cloudpln.pbcs.us1.oraclecloud.com DataManagement=false appAudit=false jobConsole=false storedSnapshotsAndFiles=true DailyMaintenanceStartTime=false
  ```
- Clonez l'ensemble de l'environnement (utilisateurs et affectations de rôle prédéfini, données d'audit, enregistrements de la console de jobs, contenu des boîtes de réception et d'envoi, instantanés stockés et enregistrements de gestion des données) à l'aide d'un instantané personnalisé. Modifiez également l'heure de début de maintenance de l'environnement cible de sorte qu'elle corresponde à celle de l'environnement source :

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
epmAutomate cloneEnvironment serviceAdmin Password.epw https://test-
cloudpln.pbcs.us1.oraclecloud.com UsersAndPreDefinedRoles=true
storedSnapshotsAndFiles=true SnapshotName=SampleSnapshot
```

- **Pour Account Reconciliation uniquement :** clonez l'environnement, les utilisateurs, les affectations de rôle prédéfini et les enregistrements Data Management, mais conservez les paramètres d'application de l'environnement cible :
  `epmAutomate cloneEnvironment serviceAdmin Password.epw https://test-cloudarcs.arcs.epm.us.oraclecloud.com UsersAndPreDefinedRoles=true ApplicationProperties=false`

compactCube

Compresse le fichier d'outline d'un cube en mode "aggregate storage" (ASO) à l'aide des paramètres définis dans un job de type *Comprimer le cube*. La compression de l'outline permet de garder une taille optimale pour le fichier d'outline de cube. La compression n'efface pas les données du cube et ne transmet pas non plus les modifications du processus métier vers Oracle Essbase.

Champ d'application

Planning, modules Planning, FreeForm, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning

Rôles requis

Administrateur de service

Syntaxe

`epmautomate compactCube` *`ASO_CUBE_NAME`*, où *`ASO_CUBE_NAME`* est le nom du cube ASO, dont vous voulez compresser l'outline.

Exemple

```
epmautomate compactCube Vis1Aso
```

copyDataByPointOfView

Copie les données d'un PDV source dans un cube vers un PDV de destination dans le même cube Enterprise Profitability and Cost Management ou dans un autre.

Champ d'application

Enterprise Profitability and Cost Management

Rôles requis

Administrateur de service

Syntaxe

`epmAutomate copyDataByPointOfView` *`SOURCE_POV_NAME TARGET_POV_NAME`*
`copyType=ALL_DATA|INPUT` *`SOURCE_CUBE_NAME TARGET_CUBE_NAME`* [*`PARAMETER=VALUE`*] où :

- *`SOURCE_POV_NAME`* est le nom du PDV source à partir duquel les données doivent être copiées.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- *TARGET_POV_NAME* est le nom d'un PDV cible valide vers lequel les données de la source doivent être copiées.
- copyType identifie les données à copier à partir du PDV source. Les valeurs valides sont les suivantes :
  - ALL_DATA, pour copier toutes les données d'entrée et calculées vers le PDV de destination.
  - INPUT, pour copier l'ensemble des données d'entrée, y compris les données d'inducteur, vers le PDV de destination.
- *SOURCE_CUBE_NAME* est le nom du cube contenant le PDV source.
- *TARGET_CUBE_NAME* est le nom du cube contenant le PDV cible.
- PARAMETER=*VALUE* indique des paramètres d'exécution facultatifs et leurs valeurs. Indiquez autant de paires paramètre-valeur que le processus l'exige. Paramètres valides et leurs valeurs :
  - povDelimiter (facultatif) est le séparateur utilisé dans les valeurs de PDV. La valeur par défaut est : : (deux-points doubles). Cette valeur doit être placée entre guillemets doubles. Exemple : povDelimiter="_".
    Outre le séparateur par défaut, seuls ceux-ci sont pris en charge : _ (trait de soulignement), # (dièse), ~ (tilde), % (pourcentage), ; (point-virgule), : (deux-points), - (tiret).
  - createDestPOV=true|false indique si le PDV cible doit être créé s'il n'existe pas. La valeur par défaut est false. Vous devez définir cette valeur de paramètre sur true si le PDV de destination n'existe pas.

Exemples

- Copiez toutes les données vers un autre PDV dans le même cube :
```
epmAutomate copyDataByPointOfView FY21_Jan_Actual_Working
FY22_Jan_Actual_Working ALL_DATA PCM_CLC PCM_CLC povDelimiter="_"
createDestPOV=true
```
- Copiez toutes les données vers un autre PDV dans un autre cube :
```
epmAutomate copyDataByPointOfView FY21_Jan_Actual_Working
FY22_Jan_Actual_Working ALL_DATA PCM_CLC PCM_REP povDelimiter="_"
createDestPOV=true
```
- Copiez les données de saisie vers un autre PDV dans le même cube :
```
epmAutomate copyDataByPointOfView FY21_Jan_Actual_Working
FY22_Jan_Actual_Working INPUT PCM_CLC PCM_CLC povDelimiter="_"
createDestPOV=true
```
- Copiez les données de saisie vers un autre PDV dans un autre cube :
```
epmAutomate copyDataByPointOfView FY21_Jan_Actual_Working
FY22_Jan_Actual_Working INPUT PCM_CLC PCM_REP povDelimiter="_"
createDestPOV=true
```

copyDataByProfile

Copie les données des éléments (par exemple, les régions) identifiés dans un profil de copie des données.

Champ d'application

Financial Consolidation and Close, Tax Reporting

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate copyDataByProfile` *`PROFILE_NAME`*, où *`PROFILE_NAME`* est le nom d'un profil de copie de données défini dans Financial Consolidation and Close et Tax Reporting.

Exemple

```
epmautomate copyDataByProfile copyDataProfile_01
```

copyFileFromInstance

Copie un fichier à partir d'un environnement source vers l'environnement à partir duquel vous exécutez cette commande.

Avant d'exécuter cette commande, à l'aide d'EPM Automate, connectez-vous à l'environnement dans lequel vous souhaitez copier le fichier.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Strategic Workforce Planning et Sales Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Syntaxe

`epmautomate copyFileFromInstance` *`SOURCE_FILE_NAME USERNAME PASSWORD_FILE URL TARGET_FILE_NAME`*, où :

- *`SOURCE_FILE__NAME`* est le nom du fichier (extension comprise) que vous voulez copier à partir de l'environnement source.
- *`USERNAME`* est le nom d'utilisateur d'un administrateur de service dans l'environnement source.
- *`PASSWORD_FILE`* représente le nom et l'emplacement du fichier qui contient le mot de passe crypté de l'administrateur de service de l'environnement source.
- *`URL`* est l'URL de l'environnement source.
- *`TARGET_FILE_NAME`* est le nom unique du fichier (extension comprise) dans l'environnement à partir duquel vous exécutez cette commande.

Exemple

```
epmautomate copyFileFromInstance "my data file.zip" serviceAdmin
C:\mySecuredir\password.epw https://test-cloud-pln.pbcs.us1.oraclecloud.com "my
target data file.zip"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

copyFromObjectStorage

Copie un fichier ou un instantané de sauvegarde d'un bucket Oracle Object Storage vers l'environnement actuel.

Si vous copiez un instantané de sauvegarde, cette commande le copie à partir du bucket Object Storage et en extrait le contenu vers Oracle Fusion Cloud Enterprise Performance Management.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate copyFromObjectStorage` *`USERNAME PASSWORD URL TARGET_FILE_NAME`*, où :

- *`USERNAME`* est l'ID d'un utilisateur qui dispose des droits d'accès requis dans Oracle Object Storage Cloud.
  Pour les utilisateurs créés dans un fournisseur d'identités fédérées, indiquez le nom qualifié complet de l'utilisateur (par exemple, `exampleIdP/jdoe` ou `exampleIdP/john.doe@example.com`, où `exampleIdP` est le nom du fournisseur d'identités fédérées). Pour les autres utilisateurs, fournissez l'ID utilisateur.
- *`PASSWORD`* est le mot de passe Swift ou le jeton d'authentification associé à l'utilisateur. Ce mot de passe est différent de celui que vous utilisez pour vous connecter à la console Object Storage. Le jeton d'authentification est un jeton généré par Oracle que vous utilisez pour l'authentification auprès d'API tierces, par exemple auprès d'un client Swift. Afin d'obtenir des instructions pour créer ce jeton, reportez-vous à la section Procédure de création d'un jeton d'authentification du guide *Documentation Oracle Cloud Infrastructure* .
- *`URL`* est l'URL du bucket Oracle Object Storage Cloud comprenant le nom de bucket et le nom de l'objet à copier.
  Format d'URL :

  `https://`*`swiftobjectstorage.region_identifier`*`.oraclecloud.com/v1/`*`namespace/bucket_name/object_name`*

  Composants de cet URL :

  - `region_identifier` est une région Oracle Cloud.
  - `namespace` est le conteneur de niveau supérieur pour tous les buckets et les objets. Un nom unique d'espace de noms Object Storage généré par le système et immuable est affecté à chaque locataire Oracle Cloud Infrastructure au moment de la création de compte. Le nom d'espace de noms de votre locataire, par exemple, `axaxnpcrorw5`, est effectif dans toutes les régions.
  - `bucket_name` est le nom du conteneur logique dans lequel vous stockez vos données et vos fichiers. Les buckets sont organisés et tenus à jour dans des compartiments. Un nom de bucket généré par le système, par exemple, `bucket-20210301-1359` reflète l'année, le mois, le jour et l'heure en cours.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

  - `object_name` est le nom de l'instantané ou du fichier que vous voulez copier à partir d'Oracle Object Storage Cloud. Cette valeur doit correspondre exactement au nom complet de l'objet dans Object Storage Cloud. N'utilisez pas d'extension telle que `.zip`, sauf si elle figure dans le nom de l'objet.

  Pour plus d'informations, reportez-vous à ces rubriques dans la *documentation Oracle Cloud Infrastructure*

  - Régions et domaines de disponibilité
  - Présentation des espaces de noms Object Storage
  - Gestion des buckets

- *`TARGET_FILE_NAME`* est le nom unique du fichier ou de l'instantané dans l'environnement Cloud EPM. Lors de la copie d'instantanés, n'indiquez pas l'extension ZIP afin que le nom de fichier puisse être utilisé avec la commande importSnapshot.
  Les fichiers dont la taille est supérieure à 100 Mo sont stockés dans Oracle Object Storage au sein d'un répertoire logique avec le fichier manifeste qui identifie les segments. Indiquez le nom du répertoire logique au format *`TARGET_FILE_NAME`*.

Exemples

Dans ces exemples, remplacez *`URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET`* par une URL fonctionnelle au format suivant : `https://swiftobjectstorage.`*`identificateur_région`*`.oraclecloud.com/v1/`*`espacedenoms/nom_bucket/`*.

- Copie d'un instantané nommé `backup_Snapshot_12_05_20.zip` d'un bucket Oracle Object Storage vers Cloud EPM et changement de son nom :
  ```
  epmautomate copyFromObjectStorage oracleidentitycloudservice/jDoe example_pwd URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET/backup_Snapshot_12_05_20.zip snapshot_from_osc
  ```
- Copie d'un instantané nommé `backup_Snapshot_12_05_20` d'un bucket Oracle Object Storage vers Cloud EPM et changement de son nom :
  ```
  epmautomate copyFromObjectStorage oracleidentitycloudservice/jDoe example_pwd URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET/backup_Snapshot_12_05_20 snapshot_from_osc
  ```
- Copie d'un instantané nommé `backup_Snapshot_12_05_20` d'un bucket Oracle Object Storage vers Cloud EPM sans changer son nom :
  ```
  epmautomate copyFromObjectStorage oracleidentitycloudservice/jDoe example_pwd URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET/backup_snapshot_12_05_20 backup_snapshot_12_05_20
  ```
- Copie d'un fichier vers Cloud EPM à partir d'un bucket Oracle Object Storage :
  ```
  epmautomate copyFromObjectStorage oracleidentitycloudservice/jDoe example_pwd URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET/example_file.txt copied_from_osc.txt
  ```

copyFromSFTP

Copie un fichier directement d'un serveur SFTP sécurisé dans un environnement Oracle Fusion Cloud Enterprise Performance Management. Grâce à cette commande, vous pouvez copier un fichier directement à partir d'un serveur SFTP, sans avoir à le télécharger d'abord vers un autre serveur à l'aide d'un client SFTP, puis à utiliser EPM Automate ou l'API REST pour le charger vers Cloud EPM.

Si une liste d'autorisation d'adresses IP est activée pour le serveur SFTP, vous devez y ajouter l'adresse IP sortante de la région OCI qui héberge les environnements Cloud EPM. Reportez-

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

vous à la section Adresses IP sortantes des centres de données et des régions Cloud EPM du *Guide des opérations* pour connaître les adresses IP sortantes des régions OCI.

> **Note**
>
> Cette commande prend en charge les serveurs SFTP exécutés sur le port 22 uniquement.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Strategic Workforce Planning et Sales Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Syntaxe

`epmautomate copyFromSftp` *`SFTP_SERVER_URL FILE_NAME`* `username=`*`USERNAME`* `password=`*`PASSWORD`* où :

- *`SFTP_SERVER_URL`* est l'URL du serveur SFTP, y compris l'emplacement et le nom du fichier à copier à partir du serveur.
- *`FILE_NAME`* est le nom du fichier dans l'environnement Cloud EPM. Indiquez le chemin d'accès, par exemple, si vous n'effectuez pas la copie vers l'emplacement de chargement par défaut.
- *`USERNAME`* est l'ID d'un utilisateur qui dispose des droits d'accès requis sur le serveur SFTP.
- *`PASSWORD`* est le mot de passe de l'utilisateur SFTP.

Exemple

```
epmautomate copyFromSftp sftp://mySFTP_Server/someDirectory/someFile.csv
myFile.csv username=jdoe password=Sftpwelcome
```

copyOwnershipDataToNextYear

Copie les données de participation de la dernière période d'une année vers la première de l'année suivante.

Les paramètres de participation initiaux par défaut et de remplacement sont automatiquement reportés de période en période dans une même année, mais pas aux périodes des années suivantes. Pour reporter les paramètres de participation les plus récents de la dernière période d'une année vers la première de l'année suivante, vous devez les copier de la dernière période de l'année dans le PDV vers la première de l'année suivante.

Champ d'application

Financial Consolidation and Close et Tax Reporting.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur

Syntaxe

`epmautomate copyOwnershipDataToNextYear` *`Scenario Year`* où :

- `Scenario` est le nom du scénario à partir duquel les données de participation doivent être copiées.
- `Year` est l'année à partir de laquelle les données de participation doivent être copiées vers la première période de l'année suivante.

Exemple

```
epmautomate copyOwnershipDataToNextYear FCCS_total_Actual FY18
```

copyPOV

Copie les artefacts de modèle et les données de cube Oracle Essbase d'un PDV source vers un PDV de destination.

Champ d'application

Profitability and Cost Management.

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

> **Remarque**
>
> Si des variables de substitution existent, vous pouvez les utiliser à la place des valeurs de paramètre réelles. Pour obtenir des informations détaillées, reportez-vous à la section Utilisation de variables de substitution avec certaines commandes Enterprise Profitability and Cost Management.

`epmautomate copyPOV` *`APPLICATION_NAME SOURCE_POV_NAME TARGET_POV_NAME`* `PARAMETER=`*`VALUE`* `stringDelimiter=`*`"DELIMITER"`* `[isInputData=true|false isAllInputData=true|false]`, où :

- *`APPLICATION_NAME`* est le nom de l'application Profitability and Cost Management contenant le PDV source.
- *`SOURCE_POV_NAME`* est le nom du PDV source dans l'application indiquée.
- *`TARGET_POV_NAME`* est le nom d'un PDV cible valide ayant le statut `Brouillon`.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `PARAMETER=VALUE` indique les paramètres d'exécution et leurs valeurs pour copier le PDV. Indiquez autant de paires paramètre-valeur que le processus l'exige. Paramètres valides et leurs valeurs :
  - `isManageRule=true|false` indique si les règles doivent être copiées.
  - `isInputData=true | isAllData=true | isAllInputData=true` (facultatif) indique comment copier les données. Pour ces paramètres, la valeur par défaut est False. Définissez uniquement l'un des paramètres suivants sur True :
    * Indiquez `isInputData=true` pour copier les données d'entrée vers le PDV de destination.
    * Indiquez `isAllData=true` pour copier toutes les données d'entrée et calculées vers le PDV de destination.
    * Indiquez `AllInputData=true` pour copier l'ensemble des données d'entrée, y compris les données d'inducteur, vers le PDV de destination.
  - `modelViewName=NAME` indique le nom du secteur de données qui sera copié du PDV source vers le PDV cible.
  - `createDestPOV=true|false` indique si le PDV cible doit être créé s'il n'existe pas.
  - `nonEmptyTupleEnabled=true|false` indique si les tuples non vides doivent être activés afin que la commande prenne en compte uniquement les croisements contenant des données. La valeur par défaut est `true`, ce qui peut entraîner, dans de rares cas, une exécution incorrecte de la commande pour la copie des données Essbase. Dans ce cas, remplacez la valeur par défaut par `nonEmptyTupleEnabled=false` afin d'améliorer les performances.

  > **Remarque**
  >
  > Les valeurs de paramètre (`true` ou `false`) doivent être indiquées entièrement en minuscules.

- `stringDelimiter="DELIMITER"` indique le séparateur utilisé dans les valeurs de PDV. Le séparateur doit être placé entre guillemets doubles.

Exemples

- `epmautomate copyPOV BksML12 2012_Jan_Actual 2012_Feb_Actual isManageRule=true isInputData=true modelViewName="Balancing - 5 Customer Costs" createDestPOV=true stringDelimiter="_"`
- `epmautomate copyPOV BksML12 2012_Jan_Actual 2012_Feb_Actual isManageRule=true isAllInputData=true createDestPOV=true stringDelimiter="_"`
- `epmautomate copyPOV BksML12 2012_Jan_Actual 2012_Feb_Actual isManageRule=true isAllData=true createDestPOV=true stringDelimiter="_"`

copySnapshotFromInstance

Copie l'instantané en cours à partir d'un environnement source vers l'environnement (cible) à partir duquel vous exécutez cette commande.

Cette commande est principalement utilisée comme première étape de la migration d'un environnement en copiant l'instantané en cours à partir d'un autre environnement, par exemple, à partir d'un environnement de test vers un environnement de production. Utilisez ensuite la commande importSnapshot pour terminer le processus de migration.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Avant d'exécuter cette commande, démarrez une session EPM Automate et connectez-vous à l'environnement cible.

Si cette commande est exécutée pour copier l'instantané actuel lors de la génération de l'instantané de l'environnement source, par exemple lors de la maintenance quotidienne, vous obtiendrez le message d'erreur `Fichier introuvable` ou `Fichier non valide`.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Strategic Workforce Planning et Sales Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Syntaxe

`epmautomate copySnapshotFromInstance` *`SNAPSHOT_NAME USERNAME PASSWORD_FILE URL`*, où :

- *`SNAPSHOT_NAME`* est le nom d'un instantané existant dans l'environnement source.
- *`USERNAME`* est le nom d'utilisateur d'un administrateur de service de l'environnement source.
- *`PASSWORD_FILE`* représente le nom et l'emplacement du fichier qui contient le mot de passe crypté de l'administrateur de service de l'environnement source.
- *`URL`* est l'URL de l'environnement source.

Exemple

```
epmautomate copySnapshotFromInstance "Artifact Snapshot" serviceAdmin
C:\mySecuredir\password.epw https://test-cloud-pln.pbcs.us1.oraclecloud.com
```

copyToObjectStorage

Copie un fichier ou un instantané de l'environnement actuel vers un bucket Oracle Object Storage Cloud.

Si vous copiez un instantané, cette commande en compresse le contenu avant de le copier vers Oracle Object Storage.

Pour permettre de copier rapidement des fichiers, cette commande divise les gros fichiers (dont la taille dépasse 100 Mo) en segments de 10 Mo (nommés *`FILE_NAME`*`/`*`FILE_NAME`*`_object_store_bytes_seg_0`, jusqu'à *`FILE_NAME`*`/`*`FILE_NAME`*`_object_store_bytes_seg_`*`n`*) et crée un fichier manifeste (nommé *`FILE_NAME`*`/`*`FILE_NAME`*`.manifest`). Les segments de fichier sont stockés dans Oracle Object Storage avec le fichier manifeste. Dans la console Object Storage, le fichier est affiché en tant que répertoire logique contenant les segments de fichier et le fichier manifeste.

Les fichiers dont la taille est inférieure à 100 Mo ne sont pas segmentés et sont stockés avec leur nom de fichier d'origine.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate copyToObjectStorage` *`SOURCE_FILE_NAME USERNAME PASSWORD URL`*, où :

- *`SOURCE_FILE_NAME`* est le nom du fichier ou de l'instantané dans Oracle Fusion Cloud Enterprise Performance Management. Si vous copiez un instantané, n'indiquez pas l'extension ZIP.
- *`USERNAME`* est l'ID d'un utilisateur qui dispose des droits d'accès en écriture requis dans Oracle Object Storage Cloud.
  Pour les utilisateurs créés dans un fournisseur d'identités fédérées, indiquez le nom qualifié complet de l'utilisateur (par exemple, `exampleIdP/jdoe` ou `exampleIdP/john.doe@example.com`, où `exampleIdP` est le nom du fournisseur d'identités fédérées). Pour les autres utilisateurs, fournissez l'ID utilisateur.
- *`PASSWORD`* est le mot de passe Swift ou le jeton d'authentification associé à l'utilisateur. Ce mot de passe est différent de celui que vous utilisez pour vous connecter à la console Object Storage. Le jeton d'authentification est un jeton généré par Oracle que vous utilisez pour l'authentification auprès d'API tierces, par exemple auprès d'un client Swift. Afin d'obtenir des instructions pour créer ce jeton, reportez-vous à la section Procédure de création d'un jeton d'authentification du guide *Documentation Oracle Cloud Infrastructure* .
- *`URL`* est l'URL du bucket Oracle Object Storage Cloud à laquelle est ajouté un nom d'objet facultatif.
  Format d'URL sans nom d'objet :

  `https://swiftobjectstorage.`*`region_identifier`*`.oraclecloud.com/v1/`*`namespace/bucket_name`*

  Format d'URL avec nom d'objet :

  `https://`*`swiftobjectstorage.region_identifier`*`.oraclecloud.com/v1/`*`namespace/bucket_name/object_name`*

  Composants de cet URL :

  - `region_identifier` est une région Oracle Cloud.
  - `namespace` est le conteneur de niveau supérieur pour tous les buckets et les objets. Un nom unique d'espace de noms Object Storage généré par le système et immuable est affecté à chaque locataire Oracle Cloud Infrastructure au moment de la création de compte. Le nom d'espace de noms de votre locataire, par exemple, `axaxnpcrorw5`, est effectif dans toutes les régions.
  - `bucket_name` est le nom du conteneur logique dans lequel vous stockez vos données et vos fichiers. Les buckets sont organisés et tenus à jour dans des compartiments. Un nom de bucket généré par le système, par exemple, `bucket-20210301-1359` reflète l'année, le mois, le jour et l'heure en cours.
  - `object_name` (facultatif) est le nom que vous utilisez pour le fichier sur Oracle Object Storage Cloud. Si vous n'indiquez aucun nom d'objet, le fichier sera copié avec son nom d'origine.

  Pour plus d'informations, reportez-vous à ces rubriques dans la *documentation Oracle Cloud Infrastructure*

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Régions et domaines de disponibilité
- Présentation des espaces de noms Object Storage
- Gestion des buckets

Exemples

Dans ces exemples, remplacez *URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET* par une URL fonctionnelle au format suivant : `https://swiftobjectstorage.`*`identificateur_région`*`.oraclecloud.com/v1/`*`espacedenoms/nom_bucket/`*.

- Copie d'un instantané vers un bucket Oracle Storage et changement de son nom :
  `epmautomate copyToObjectStorage "Artifact Snapshot" oracleidentitycloudservice/jDoe example_pwd` *`URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET`*`/Snapshot_04_30_21`
- Copie d'un fichier vers un bucket Oracle Storage :
  `epmautomate copyToObjectStorage example_file.txt oracleidentitycloudservice/jDoe example_pwd` *`URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET`*
- Copie d'un fichier vers un bucket Oracle Storage et changement de son nom :
  `epmautomate copyToObjectStorage example_file.txt eoracleidentitycloudservice/jDoe example_pwd` *`URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET`*`/epm_text_file.txt`

copyToSFTP

Copie un fichier à partir d'un emplacement dans un environnement Oracle Fusion Cloud Enterprise Performance Management vers un emplacement sur un serveur FTP sécurisé. Grâce à cette commande, vous pouvez copier un fichier directement à partir d'un environnement vers un serveur SFTP sans d'abord le télécharger vers un autre serveur à l'aide d'EPM Automate ou de l'API REST, puis en utilisant un client SFTP pour le charger vers le serveur SFTP.

Si une liste d'autorisation d'adresses IP est activée pour le serveur SFTP, vous devez y ajouter l'adresse IP sortante de la région OCI qui héberge les environnements Cloud EPM. Reportez-vous à la section Adresses IP sortantes des centres de données et des régions Cloud EPM du *Guide des opérations* pour connaître les adresses IP sortantes des régions OCI.

> **Note**
>
> Cette commande prend en charge les serveurs SFTP exécutés sur le port 22 uniquement.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de service

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Syntaxe

`epmautomate copyToSftp SFTP_SERVER_URL EPM_FILE_NAME username=USERNAME password=PASSWORD` où :

- *SFTP_SERVER_URL* est l'URL du serveur SFTP, y compris le répertoire dans lequel le fichier doit être copié et le nom du fichier.
- *EPM_FILE_NAME* est le nom et l'emplacement (s'il ne s'agit pas de la boîte d'envoi par défaut) du fichier à copier à partir de l'environnement Cloud EPM. N'indiquez pas l'extension ZIP si vous copiez un instantané.
- *USERNAME* est l'ID d'un utilisateur qui dispose des droits d'accès requis sur le serveur SFTP.
- *PASSWORD* est le mot de passe de l'utilisateur SFTP.

Exemple

```
epmautomate copyToSftp sftp://mySFTP_Server/someDirectory/some_file.csv
myFile.csv username=jDoe password=SftpWelcome
```

createGroups

Ajoute des groupes dans Contrôle d'accès à l'aide d'un fichier CSV ANSI ou UTF-8 qui a été chargé vers l'environnement.

Un administrateur de service utilise la commande uploadFile pour charger des fichiers dans un environnement. Le format de fichier est le suivant :

```
Group Name,Description
Example_grp1,My test group
Example_grp2,My other test group
```

Les noms de groupe ne sont pas sensibles à la casse. Une fois l'exécution terminée, la commande imprime les informations relatives à chaque entrée en échec dans la console. Reportez-vous à ces informations afin de comprendre pourquoi l'exécution de la commande a échoué pour certaines entrées dans le fichier CSV.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer

Syntaxe

`epmautomate createGroups FILE_NAME`, où *FILE_NAME* représente le nom d'un fichier CSV contenant les noms des groupes et leur description.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exemple

```
epmautomate createGroups group_file.CSV
```

createNRSnapshot

Créez un instantané à la demande, nommé `EPRCS_Backup.tar.gz`, d'un environnement Narrative Reporting.

Vous pouvez télécharger `EPRCS_Backup.tar.gz` ainsi qu'un fichier d'erreurs sur un ordinateur local à l'aide de la commande downloadFile, ou le copier dans un autre environnement à l'aide de la commande copyFileFromInstance.

Les données d'application dans `EPRCS_Backup.tar.gz` correspondent à celles de la dernière maintenance quotidienne. Si vous devez sauvegarder des données plus récentes, utilisez la fonctionnalité d'export de données Narrative Reporting.

Champ d'application

Narrative Reporting

Rôles requis

Administrateur de service

Syntaxe

`epmautomate createNRSnapshot [errorFile=Error_File.txt]`, où `errorFile` (facultatif) identifie le nom d'un fichier texte unique pour enregistrer les erreurs, si la commande en détecte.

Exemple

```
epmautomate createNRSnapshot errorFile=EPRCS_backup_Error.txt
```

createReconciliations

Copie les profils dans la période spécifiée à l'aide d'un filtre public.

Champ d'application

Account Reconciliation.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini (le rôle de visualiseur nécessite le rôle d'application Rapprochement - Préparateur)

Syntaxe

`epmautomate createreconciliations` *`PERIOD SAVED_FILTER`*, où :

- *`PERIOD`* est le nom d'une période.
- *`SAVED_FILTER`* est le nom d'un filtre public enregistré. Si vous n'indiquez pas de filtre enregistré, la commande copie tous les profils applicables à l'aide de filtres publics.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Remarque**
>
> Si vous indiquez le nom d'un filtre privé, la commande ignore le filtre et affiche le message `Filtre introuvable`.

Exemples

- Copie de tous les profils de la période : `epmautomate createReconciliations "January 2015"`
- Copie des profils d'un filtre donné : `epmautomate createReconciliations "January 2015" "Corporate Recs"`

deleteFile

Supprime un fichier ou un instantané de l'emplacement de chargement par défaut, de la boîte de réception ou d'envoi, d'un dossier de gestion des données ou de profitinbox/profitoutbox.

Pour supprimer un fichier d'un emplacement autre que l'emplacement de chargement par défaut, vous devez indiquer l'emplacement du fichier.

Si cette commande est exécutée pour supprimer un instantané en cours de génération ou d'archivage, vous obtiendrez l'un des messages d'erreur suivants :

- Erreur `Fichier introuvable` ou `Fichier non valide` si l'instantané est en cours de génération
- `Le processus d'archivage est en cours. Impossible de renommer ou de supprimer X` si l'instantané est en cours d'archivage

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Syntaxe

```
epmautomate deleteFile FILE_NAME
```

> **Remarque**
>
> Vous devez indiquer le nom du fichier avec l'extension, par exemple, `data.csv`, `data.zip`, si applicable. Vous pouvez supprimer un instantané sans indiquer son extension de fichier (.zip). Cette pratique est cependant désapprouvée. Vous devez indiquer l'emplacement du fichier s'il ne se trouve pas à l'emplacement par défaut. Pour obtenir des informations détaillées, reportez-vous à Emplacements de fichier par défaut. Les emplacements pris en charge incluent `inbox`, `profitinbox`, `outbox`, `profitoutbox`, `to_be_imported` et `inbox/nom_répertoire`.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exemples

- Suppression d'un fichier de l'emplacement de chargement par défaut :
  `epmautomate deleteFile data.csv`
- Suppression d'un fichier de la boîte de réception :
  `epmautomate deleteFile inbox/data.csv`
- Suppression de la boîte d'envoi :
  `epmautomate deleteFile outbox/data.csv`
- Suppression d'un instantané que vous avez créé à l'aide de la migration :
  - `epmautomate deleteFile "Backup 18-06-12.zip"` ou
  - `epmautomate deleteFile "Backup 18-06-12"` (désapprouvé)
- Suppression de profitinbox (Profitability and Cost Management) :
  `epmautomate deleteFile profitinbox/data.csv`
- Suppression de profitoutbox (Profitability and Cost Management) :
  `epmautomate deleteFile profitoutbox/data.csv`
- Suppression d'un dossier de chargement de gestion des données :
  `epmautomate deleteFile inbox/dm_data/data.csv`
- Suppression d'un dossier de gestion des données :
  `epmautomate deleteFile outbox/dm_data/data.csv`

deleteGroups

Enlève des groupes de Contrôle d'accès en fonction des informations disponibles dans un fichier CSV ANSI ou UTF-8 qui a été chargé vers l'environnement.

Un administrateur de service utilise la commande uploadFile pour charger des fichiers dans un environnement. Le format de fichier est le suivant :

```
Group Name
Example_grp1
Example_grp2
```

Les valeurs de nom de groupe dans le fichier ne respectent pas la casse. Une fois l'exécution terminée, la commande imprime les informations relatives à chaque entrée en échec dans la console. Reportez-vous à ces informations afin de comprendre pourquoi l'exécution de la commande a échoué pour certaines entrées dans le fichier CSV.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Syntaxe

`epmautomate deleteGroups FILE_NAME`, où *`FILE_NAME`* représente le nom d'un fichier CSV contenant les noms des groupes à enlever du contrôle d'accès.

Exemple

```
epmautomate deleteGroups group_file.CSV
```

deletePointOfView

Supprime les artefacts et les données de cube Oracle Essbase d'un PDV dans une application Enterprise Profitability and Cost Management.

Champ d'application

Enterprise Profitability and Cost Management

Rôles requis

Administrateur de service

Syntaxe

`epmautomate deletePointOfView POV_NAME [povDelimiter="DELIMITER"]` où :

- *`POV_NAME`* identifie le nom du PDV à supprimer.
- `povDelimiter` est le séparateur utilisé dans les valeurs de PDV. La valeur par défaut est `::` (deux-points doubles). Cette valeur doit être placée entre guillemets doubles. Exemple : `povDelimiter="_"`.
  Outre le séparateur par défaut, seuls ceux-ci sont pris en charge : _ (trait de soulignement), # (dièse), ~ (tilde), % (pourcentage), ; (point-virgule), : (deux-points), - (tiret).

Exemple

- Suppression d'un PDV qui utilise un séparateur de PDV personnalisé
  ```
  epmAutomate deletePointOfView FY21_Jan_Actual_Working povDelimiter="_"
  ```
- Suppression d'un PDV qui utilise le séparateur de PDV par défaut
  ```
  epmAutomate deletePointOfView FY21::Jan::Actual::Working
  ```

deletePOV

Supprime les artefacts de modèle et les données de cube Oracle Essbase d'un PDV dans Profitability and Cost Management.

Champ d'application

Profitability and Cost Management

Rôles requis

- Administrateur de service
- Super utilisateur

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Syntaxe

> **Remarque**
>
> Si des variables de substitution existent, vous pouvez les utiliser à la place des valeurs de paramètre réelles. Pour obtenir des informations détaillées, reportez-vous à la section Utilisation de variables de substitution avec certaines commandes Enterprise Profitability and Cost Management.

`epmautomate deletePOV APPLICATION_NAME POV_NAME stringDelimiter="DELIMITER"` où :

- *APPLICATION_NAME* est le nom de l'application Profitability and Cost Management contenant le PDV à supprimer.
- *POV_NAME* est le nom du PDV à supprimer. Cette valeur est obligatoire.
- `stringDelimiter="DELIMITER"` indique le séparateur utilisé dans les valeurs de PDV. Le séparateur doit être placé entre guillemets doubles.

Exemple

```
epmautomate deletePOV BksML12 2012_Jan_Actual stringDelimiter="_"
```

deployCube

Déploie ou redéploie le cube de calcul d'une application Profitability and Cost Management.

Champ d'application

Profitability and Cost Management

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate deployCube APPLICATION_NAME PARAMETER=VALUE comment="comment"` où :

- *APPLICATION_NAME* est le nom d'une application Profitability and Cost Management.
- `PARAMETER=VALUE` indique les paramètres d'exécution et leurs valeurs pour déployer le cube. Indiquez autant de paires paramètre-valeur que le processus l'exige. Paramètres valides et leurs valeurs :

  > **Remarque**
  >
  > Les valeurs de paramètre (`true` ou `false`) doivent être indiquées entièrement en minuscules.

  – `isKeepData=true|false`

    indique si les données existantes doivent être conservées, le cas échéant

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

  - `isReplaceCube=true|false` indique si le cube existant doit être remplacé

    > **Remarque**
    >
    > Les valeurs de `isKeepData` et `isReplaceCube` ne peuvent pas être toutes les deux définies sur `true`.

  - `isRunNow=true|false` indique si le processus doit être exécuté dès à présent
- `comment` est un commentaire facultatif placé entre guillemets doubles.

Exemple

```
epmautomate deployCube BksML12 isKeepData=true isReplaceCube=false isRunNow=true
comment="Test cube deployment"
```

deployEJTemplates

Déploie des modèles de journal Enterprise finalisés vers des périodes ouvertes dans Financial Consolidation and Close. Le déploiement de modèles de journal Enterprise crée des journaux récurrents associés au modèle pour la période sélectionnée. Il permet également de créer des journaux ad hoc à l'aide des modèles déployés.

Cette commande constitue une alternative à l'utilisation d'écrans Financial Consolidation and Close pour le déploiement de nouveaux modèles de journal Enterprise en début de mois.

Champ d'application

Financial Consolidation and Close

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate deployEJTemplates` *`YEAR PERIOD`* `[Template=`*`TEPMPLATE_NAME`*`] [ResetJournals=true|false]`, où :

- `Year` correspond à l'année du journal.
- `Period` correspond à la période du journal. Cette valeur ne peut être renseignée que si l'année est indiquée.
- `Template=`*`TEMPLATE_NAME`* identifie les noms des journaux à déployer. Pour déployer plusieurs journaux, indiquez chaque nom de modèle unique au format `Template=`*`TEMPLATE_NAME`*, par exemple : `Template="Loan Details" Template="Housing Details" Template="Repayment Details"`.
  Si la valeur de ce paramètre n'est pas indiquée, la commande déploie tous les modèles de la combinaison année/période spécifiée.
- `ResetJournals` (facultatif) indique si la première étape de tous les journaux doit être rétablie après le redéploiement des modèles. La valeur par défaut est `false`.
  Financial Consolidation and Close valide cette valeur en interne en fonction des modifications apportées aux modèles et peut remplacer la valeur indiquée si besoin.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exemple

```
epmautomate deployEJTemplates 2021 May Template="Loan Details" Template="Housing Details" ResetJournals=true
```

deployFormTemplates

Déploie les modèles de formulaire finalisés vers les nouvelles périodes de collecte de données afin de créer des formulaires Supplemental Data et de garantir un processus de collecte de données cohérent pouvant être répété.

Champ d'application

Financial Consolidation and Close, Tax Reporting.

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate deployFormTemplates COLLECTION_INTERVAL [DIMENSION] [Template] [resetWorkFlows=true|false]` où :

- `COLLECTION_INTERVAL` est le nom de l'intervalle de collecte vers lequel le modèle doit être déployé.
- `DIMENSION` (facultatif) indique les dimensions de fréquence du processus de collecte des données au format *`DIMENSION=MEMBER_NAME`*. Indiquez autant de dimensions que celles définies dans l'intervalle de collecte (quatre au maximum y compris Year et Period), par exemple : `"Year=2020" "Period=July" "Product=Oracle EPM" "Consolidation=entity Input"`. Si vous n'indiquez aucune valeur pour ce paramètre, aucune valeur par défaut n'est utilisée.
- `Template` (facultatif) identifie les noms uniques des modèles de formulaire à déployer au format `Template=`*`TEMPLATE_NAME`*. Vous pouvez indiquer autant de noms uniques que nécessaire dans ce format. Exemple : `Template="Loan Details Template" Template="Housing Details Template" Template="Repayment Detals Template"`. Si cette valeur de propriété n'est pas indiquée, la commande déploie tous les modèles de l'intervalle spécifié.
- `resetWorkFlows` (facultatif) indique si la première étape est à rétablir pour tous les formulaires après leur déploiement. La valeur par défaut est `false`.

Exemple

```
epmautomate deployFormTemplates "Journal Collection Interval" "Year=2020" "Period=July" "Product=Oracle EPM" "Consolidation=entity Input" Template="Loan Details Template" Template="Housing Details Template" resetWorkFlows=true
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

deployTaskManagerTemplate

Déploie des tâches d'un modèle Task Manager vers un échéancier de tâches pour garantir l'exécution cohérente des processus métier répétitifs.

Champ d'application

Enterprise Profitability and Cost Management, Financial Consolidation and Close et Tax Reporting

Rôles requis

Administrateur de service

Syntaxe

`epmAutomate deployTaskManagerTemplate` *`TEMPLATE_NAME SCHEDULE_NAME YEAR PERIOD DAY_ZERO_DATE`* `[dateFormat=`*`DATE_FORMAT`*`] [orgUnit=`*`ORGANIZATION UNIT`*`]`, où :

- *`TEMPLATE_NAME`* est le nom du modèle Task Manager à déployer.
- *`SCHEDULE_NAME`* est le nom de l'échéancier à créer à partir du modèle.
- *`YEAR`* est le membre de dimension Année vers lequel le modèle doit être déployé.
- *`PERIOD`* est le membre de dimension Période vers lequel le modèle doit être déployé.
- *`DAY_ZERO_DATE`* est la date du jour zéro, au format valide, à utiliser pour créer l'échéancier.
- `dateFormat` (facultatif) est le format de date à utiliser pour la date du jour zéro. Le format par défaut est AAAA-MM-JJ.
- `orgUnit` (facultatif) est le nom de l'unité organisationnelle. Si aucune valeur n'est indiquée, l'échéancier sera créé à l'aide du mapping de date standard. Les règles de jour férié ne seront pas utilisées.

Exemple

- Déployez un modèle Task Manager pour l'unité organisationnelle *Ind* à l'aide du format de date par défaut (YYYY-MM-DD) pour la date du jour zéro :
  ```
  epmautomate deployTaskManagerTemplate "Vision Monthly Close" "Qtr 2 Close"
  2021 July 2021-07-10 orgUnit=Ind
  ```
- Déploiement du modèle Task Manager pour l'unité organisationnelle Ind en utilisant le format jj/mm/aaaa pour le jour zéro :
  ```
  epmautomate deployTaskManagerTemplate "Vision Monthly Close" "Qtr 2 Close"
  2021 July 02/07/2021 dateFormat=dd/MM/yyyy orgUnit=Ind
  ```

dismissIPMInsights

Automatise le rejet des données Intelligent Performance Management (IPM) Insights avant d'exécuter de nouveaux jobs IPM Insights. Le rejet des données ferme toutes les informations clés ouvertes sur lesquelles vous n'envisagez aucune action. Cette commande est une alternative au rejet manuel des données à l'aide du tableau de bord IPM Insights.

Champ d'application

Planning, modules Planning , Strategic Workforce Planning, Sales Planning.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate dismissIPMInsights [comment="comment"]` où comment représente la justification du rejet des informations clés, le cas échéant.

Exemple

```
epmautomate dismissIPMInsights comment="dismissing unusable insights"
```

downloadFile

Télécharge un fichier d'un environnement vers un ordinateur local.

Cette commande vous permet de télécharger des données, des métadonnées et des instantanés de sauvegarde pour un stockage en local. Ce fichier est téléchargé dans le dossier à partir duquel vous exécutez EPM Automate.

Si cette commande est exécutée pour télécharger l'instantané actuel lors de la génération de l'instantané de l'environnement, par exemple lors de la maintenance quotidienne, vous obtiendrez le message d'erreur `Fichier introuvable` ou `Fichier non valide`.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Syntaxe

`epmautomate downloadFile "[FILE_PATH]/FILE_NAME]"`

> **Conseil**
>
> Utilisez la commande listFiles pour identifier l'emplacement du fichier à télécharger.

Exemples

- Téléchargement d'un instantané de maintenance : `epmautomate downloadFile "Artifact Snapshot"`
- Téléchargement d'un instantané personnalisé : `epmautomate downloadFile "mySnapshot.zip"`
- Téléchargement d'un instantané de maintenance Narrative Reporting : `epmautomate downloadFile "EPRCS_Backup.tar.gz"`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Téléchargement d'un fichier à partir de l'emplacement de téléchargement par défaut : `epmautomate downloadFile data.csv`
- Téléchargement à partir d'un dossier de gestion des données : `epmautomate downloadfile outbox/dm_data/data.csv`
- Téléchargement à partir de profitoutbox : `epmautomate downloadFile profitOutbox/data.csv`
- Téléchargement du rapport d'activité *2025-06-01 20_38_43.html* à partir du dossier *apr/2025-06-01* dans un environnement Account Reconciliation : `epmautomate downloadfile "apr/2025-06-01 20_38_43/2025-06-01 20_38_43.html"`

enableApp

Active une application.

Champ d'application

Profitability and Cost Management

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate enableapp` *`APPLICATION_NAME`*, où *`APPLICATION_NAME`* est le nom de l'application Profitability and Cost Management à activer.

Exemple

```
epmautomate enableApp BksML12
```

nenableQueryTracking

Active le suivi des requêtes sur les cubes ASO pour commencer à capturer des modèles d'extraction de données utilisateur (requêtes).

Utilisez les modèles d'extraction de données capturés pour optimiser l'agrégation de cubes ASO, lancée à l'aide de la commande executeAggregationProcess.

Champ d'application

Planning, modules Planning, FreeForm, Enterprise Profitability and Cost Management, Strategic Workforce Planning Cloud et Sales Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate enableQueryTracking` *`ASO_CUBE_NAME`*, où *`ASO_CUBE_NAME`* est le nom du cube ASO dans lequel le suivi des requêtes doit être activé.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exemple

```
epmautomate enableQueryTracking VISION_ASO
```

encrypt

Utilise l'algorithme AES (`AES/CBC/PKCS5Padding(256)`) afin de crypter le mot de passe Oracle Fusion Cloud Enterprise Performance Management (ou le jeton d'actualisation OAuth 2.0 et l'ID client permettant d'accéder aux environnements) et, éventuellement, le mot de passe de serveur proxy Internet employé pour la connexion aux environnements, et le stocke dans un fichier de mots de passe.

Le cryptage des clés secrètes permet aux administrateurs de service de partager leur fichier de mots de passe cryptés avec les développeurs qui écrivent les scripts EPM Automate pour qu'ils puissent exécuter les scripts. Vous évitez ainsi de partager le mot de passe de l'administrateur de service ou de créer un compte Cloud EPM partagé générique spécifiquement pour l'exécution des scripts.

Le processus de cryptage du mot de passe n'est effectué qu'une seule fois.

> **Remarque**
>
> Reportez-vous à Gestion des caractères spéciaux pour en savoir plus sur le cryptage des mots de passe qui contiennent des caractères spéciaux.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Syntaxe

`epmautomate encrypt` *`PASSWORD|REFRESH_TOKEN KEY PASSWORD_FILE`* `[ClientID=`*`CLIENT_ID`*`] [ProxyServerPassword=`*`PROXY_PASSWORD`*`]`, où :

- *`PASSWORD|REFRESH_TOKEN PASSWORD`* est le mot de passe ou le jeton d'actualisation OAuth à crypter. Vous ne pouvez pas utiliser d'informations d'identification d'entreprise avec EPM Automate.
- *`KEY`* est la clé privée à utiliser pour crypter le mot de passe.
- *`PASSWORD_FILE`* représente le nom et l'emplacement du fichier qui stocke le mot de passe crypté ou le jeton d'actualisation. Le fichier de mots de passe doit utiliser l'extension `.epw`.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `ClientID` (facultatif) est l'identificateur client créé lors de la configuration d'OAuth 2.0. Cette valeur doit être indiquée lors du cryptage d'un jeton d'actualisation OAuth 2.0. N'indiquez pas cette valeur lors du cryptage d'un mot de passe.
- `ProxyServerPassword` est le mot de passe qui permet d'authentifier l'utilisateur auprès du serveur proxy HTTP. Nécessaire uniquement si l'authentification auprès du serveur proxy est activée pour votre réseau.

Exemples

- **Crypter uniquement le mot de passe Cloud EPM :** `epmautomate encrypt P@ssword1 myKey C:\mySecuredir\password.epw`
- **Crypter le mot de passe Cloud EPM et celui du serveur proxy Internet :** `epmautomate encrypt E@xample1 myKey C:\mySecuredir\password.epw ProxyServerPassword=Proxy_Pwd1`
- **Crypter le jeton d'actualisation et l'ID client :** `epmautomate encrypt AAyyilYBAWD4....FVkxefd8kjoJr6HJPA= myEncyprtion42Key C:\mySecuredir\oauthfile1.epw ClientID=6fdf2e72fd343430ABR22394C`

essbaseBlockAnalysisReport

Crée le rapport d'analyse de bloc Essbase qui vous aide à analyser les données Oracle Essbase pour prendre en charge l'ajustement des cubes de l'option "block storage" (BSO) (généralement utilisés pour les calculs) dans votre application.

Le rapport d'analyse de bloc Essbase est utile pour résoudre les problèmes de performances résultant de modèles de données. Par exemple, des nombres répétés dans les cubes BSO d'Essbase. Ce rapport fournit des informations sur les trois domaines suivants :

1. **Pourcentage de blocs contenant uniquement des zéros :** affiche les blocs contenant uniquement des zéros en tant que pourcentage de tous les blocs figurant dans le fichier d'export.
2. **10 principales valeurs de cellule numérique répétées par pourcentage de cellules numériques :** ce tableau présente les 10 principales valeurs répétées en tant que pourcentage de toutes les valeurs du fichier d'export.
3. **100 principales combinaisons de membres denses avec des valeurs répétées :** ce tableau présente les 100 principales combinaisons denses avec des valeurs répétées dans le cube. La colonne **Valeur de cellule** affiche une valeur pour chaque membre, dans l'ordre dans lequel il apparaît dans la hiérarchie, dans une colonne différente. Par exemple, si la période se répète sur l'ensemble de la colonne, il existera une colonne différente pour janvier, février, etc. D'autres dimensions denses apparaissent dans les lignes. Cela doit vous aider à identifier les emplacements des valeurs répétées.

Avant d'exécuter ce rapport, utilisez la commande exportEssbaseData afin d'exporter les données du cube pour lequel vous voulez créer le rapport d'analyse de bloc vers un fichier ZIP. Vous pouvez exporter les données de niveau 0 ou toutes les données selon vos besoins. Exécutez cette commande afin de créer le rapport d'analyse de bloc pour ce fichier ZIP. Le rapport est créé dans la boîte d'envoi. Vous pouvez utiliser la commande downloadFile pour le télécharger sur un ordinateur local ou la commande sendMail afin de l'envoyer par courriel.

Champ d'application

Financial Consolidation and Close, Planning, modules Planning, FreeForm, Strategic Workforce Planning, Sales Planning et Tax Reporting.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate essbaseBlockAnalysisReport` *`EXPORT_DATA_FILE`*`.zip` *`REPORT_FILE`*, où

- `EXPORT_DATA_FILE.zip` est le nom du fichier ZIP qui contient les données Essbase qui ont été précédemment exportées d'un cube BSO à l'aide de la commande exportEssbaseData.
- `REPORT_FILE` est le nom du fichier de rapport d'analyse de bloc au format HTML.

Exemple

```
epmautomate essbaseBlockAnalysisReport Plan1_cube_data.zip block_report.html
```

executeAggregationProcess

Lance le processus d'agrégation, en utilisant éventuellement les statistiques de suivi des requêtes, pour améliorer les performances des cubes ASO. Il s'agit d'une étape importante pour l'optimisation des cubes ASO.

Avant d'exécuter cette commande :

- utilisez la commande nenableQueryTracking afin de capturer des statistiques d'extraction de données pour optimiser l'agrégation ASO ;
- prévoyez un délai suffisant afin de permettre au processus métier de capturer des modèles d'extraction de données utilisateur (requêtes) qui peuvent être utilisés pour créer des vues agrégées.

Champ d'application

Planning, modules Planning, FreeForm, Enterprise Profitability and Cost Management, Strategic Workforce Planning Cloud et Sales Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate executeAggregationProcess` *`ASO_CUBE_NAME`* `[useQueryData=true|false] [includeAlternateRollups=disable|enable] [growthSizeRatio=`*`VALUE`*`]`, où :

- `useQueryData` permet d'utiliser des données de requête enregistrées, collectées à l'aide du suivi des requêtes, pour sélectionner l'ensemble de vues agrégées le plus approprié. La valeur par défaut est `false`.
- `includeAlternateRollups` permet d'inclure les hiérarchies secondaires (avec utilisation de niveau par défaut) dans le processus de sélection de vues. La valeur par défaut est `disable`.
- `growthSizeRatio` (facultatif) est le taux de croissance maximale du cube pour agréger les vues sélectionnées par le serveur. La croissance du cube s'arrête lorsque la croissance maximale atteint le taux que vous indiquez. Le paramètre par défaut permet au cube de croître sans limite de taux de croissance.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Note**
>
> Pour créer des vues agrégées par défaut, exécutez cette commande sans indiquer de paramètres facultatifs.

Exemples

- Création d'une vue agrégée en fonction des données de requête capturées à l'aide de la commande nenableQueryTracking :
  ```
  epmautomate executeAggregationProcess VISION_ASO useQueryData=true
  includeAlternateRollups=enable
  ```
- Création d'une vue agrégée par défaut :
  ```
  epmautomate executeAggregationProcess Vis1ASO
  ```

executeBurstDefinition

Exécute une définition d'éclatement qui indique les artefacts, PDV et autres paramètres requis afin d'exécuter des rapports ou des liasses pour plusieurs membres d'une dimension unique pour une source de données.

Champ d'application

Narrative Reporting

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Un niveau de sécurité supplémentaire doit être affecté aux utilisateurs dotés du rôle Super utilisateur, Utilisateur ou Visualiseur via une liste de contrôle d'accès.

Syntaxe

`epmAutomate executeBurstDefinition` *`ARTIFACT_NAME`*, où *`ARTIFACT_NAME`* représente le nom et le chemin de la définition d'éclatement.

Exemple

```
epmAutomate executeBurstDefinition "library/Reports/Example BurstDef1"
```

executeReportBurstingDefinition

A l'aide d'une définition d'éclatement, exécute un éclatement pour un seul rapport ou une seule liasse pour plusieurs membres d'une même dimension et publie une sortie sous forme de fichier PDF ou Excel statique (non actualisable dans Oracle Smart View for Office) pour chaque membre.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service

Syntaxe

`epmAutomate executeReportBurstingDefinition` *`BURST_DEFINITION_NAME`* `[jobName=`*`JOB_NAME`*`]` où :

- *`BURST_DEFINITION_NAME`* est le chemin et le nom d'une définition d'éclatement.
- *`JOB_NAME`* (facultatif) est le nom du job à utiliser pour exécuter la définition d'éclatement. La valeur par défaut est `Execute Bursting Definition`.

Exemple

```
epmAutomate executeReportBurstingDefinition /Library/MonthlySalesBurstDef
```

exportAccessControl

Exporte dans un fichier CSV ou XLS le rapport détaillé sur les utilisateurs, qui contient des informations sur les utilisateurs dotés de rôles prédéfinis dans l'environnement et répertorie les attributs de chaque utilisateur (par exemple, leur nom et leur adresse électronique) ainsi que des informations sur leur accès (par exemple, leur affectation à des groupes, à des équipes ou à des organisations).

Exemple de rapport :

	A	B	C	D	E	F	G	H	I	J	K	L
1	Name	User Login	Status	Teams	Email	Role	Workflow Roles	Preparer	Reviewer	Organizations	Power User Filter	Last Login
2	John Doe	john.doe@example.com	Available	AP Preparers AR Preparers	john.doe@example.com	Administrator		Yes	Yes			
3	Jane Doe	jane.doe@example.com	Available	AP Reviewers AR Reviewers	jane.doe@example.com	Power User		No	No			
4	ats_power_user4	ats_power_user4	Available		example1@example.com	User		No	No			
5	ats_user1	ats_user1	Available		example2@example.com	User		No	No			
6	ats_ view_user1	ats_view_user1	Available		view.user@exaple.com	Viewer		No	No			

Vous pouvez télécharger ce rapport à l'aide de la commande downloadFile.

Champ d'application

Account Reconciliation

Rôles requis

Administrateur de service

Syntaxe

`epmAutomate exportAccessControl` *`REPORT_NAME`* `[reportFormat=XLS|CSV]`, où :

- *`REPORT_NAME`* est le nom du fichier d'export dans lequel figurera le rapport.
- `reportFormat` (facultatif) est le format du fichier. Les valeurs valides sont XLS et CSV (par défaut).

Exemple

```
epmAutomate exportAccessControl aclreport.xls reportFormat=XLS
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

exportAppAudit

Exporte les enregistrements d'audit de données dans un fichier ZIP, que vous pouvez télécharger et archiver sur un ordinateur local. Les informations d'audit sont disponibles pendant 365 jours dans l'environnement. Vous pouvez générer ce rapport pour un nombre de jours ou une plage de dates.

Le premier caractère du fichier de sortie CSV est l'indicateur d'ordre des octets `\ufeff`, suivi d'un identificateur d'application crypté entouré de guillemets. L'en-tête du fichier CSV suit l'identificateur d'application.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Strategic Workforce Planning et Sales Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Audit - Gérer
- Tout rôle prédéfini et rôle d'application Audit - Afficher

Syntaxe

`epmautomate exportAppAudit` *`EXPORT_FILE_NAME`* `[userNames=`*`USER_NAMES`*`] [nDays=`*`Number_of_Days`*`] [startDate=`*`START_DATE`* `endDate=`*`END_DATE`*`] [excludeApplicationId=true|false]`, où :

- *`EXPORT_FILE_NAME`* est le nom du fichier ZIP dans lequel seront stockées les données d'audit exportées. Vous utilisez la commande downloadFile pour télécharger des fichiers vers un environnement.
- `userNames` (facultatif) est la liste des noms de connexion utilisateur, séparés par des virgules. Si cette option est définie, seules les données d'audit créées par ces utilisateurs seront exportées. N'indiquez pas cette valeur si vous voulez exporter les données d'audit pour l'ensemble des utilisateurs.
- `nDays` (facultatif) indique le nombre de jours pour lequel exporter les enregistrements d'audit. La valeur par défaut est de sept jours. Valeurs possibles : `all` afin d'exporter les données d'audit disponibles pour les 365 derniers jours, `1`, `2`, `7`, `30`, `60` et `180`. N'indiquez pas cette valeur si vous générez le rapport pour une plage de dates.
- `startDate` et `enddate` (facultatifs) identifient les dates de début et de fin auxquelles les enregistrements d'audit doivent être exportés. Ces valeurs doivent être spécifiées au format *YYYY-MM-DD*.
  - La date de début doit précéder la date de fin. Pour générer un rapport, vous devez préciser une date de début et une date de fin.
  - Ces dates ne doivent pas être des dates futures.
  - Ces dates sont ignorées si la valeur facultative `nDays` est spécifiée.
- `excludeApplicationId` (facultatif) indique si l'identificateur d'application doit être écrit dans le fichier d'export. La valeur par défaut est `false`.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Remarque**
>
> Les données issues de fichiers exportés qui ne contiennent pas l'identificateur d'application ne peuvent pas être importées dans des environnements Oracle Fusion Cloud Enterprise Performance Management.

Exemples

- Export des données d'audit des 30 derniers jours des utilisateurs spécifiés avec l'identificateur d'application :
  ```
  epmautomate exportAppAudit auditData userNames=johnDoe,jane.doe@example.com
  ndays=30
  ```
- Export des données d'audit des 30 derniers jours des utilisateurs spécifiés sans l'identificateur d'application :
  ```
  epmautomate exportAppAudit auditData userNames=johnDoe,jane.doe@example.com
  ndays=30 excludeApplicationId=true
  ```
- Export des données d'audit d'août 2024 de tous les utilisateurs avec l'identificateur d'application :
  ```
  epmautomate exportAppAudit auditData startDate=2024-08-01 endDate=2024-08-31
  ```

exportAppSecurity

Exporte les affectations d'accès de niveau artefact dans un fichier CSV, que vous pouvez télécharger et stocker en local.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate exportAppSecurity` *`EXPORT_FILE_NAME`*`.CSV`, où *`EXPORT_FILE_NAME`* est le nom du fichier dans lequel seront stockées les données de sécurité exportées. Ce fichier sera créé dans la boîte d'envoi, à partir de laquelle vous pourrez le télécharger vers votre ordinateur.

Exemple

```
epmautomate exportAppSecurity app_security.CSV
```

exportARApplicationProperties

Exporte dans un fichier JSON les paramètres d'application Account Reconciliation (en lien avec l'expérience Redwood, le thème, les notifications par courriel et le nom de processus métier) ainsi que l'image d'arrière-plan et l'image de logo pour pouvoir les importer dans le même environnement ou dans un autre.

Cette commande est utile lors de l'import d'une application d'un environnement de production vers un environnement de test. Si vos paramètres d'application diffèrent dans les environnements de production et de test, vous pouvez les exporter à partir de l'environnement

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

de test avant d'importer l'application à partir de l'environnement de production, puis importer les paramètres dans l'environnement de test afin de conserver les paramètres d'origine.

Champ d'application

Account Reconciliation

Rôles requis

Administrateur de service

Syntaxe

```
epmautomate exportARApplicationProperties FILE_NAME
[Properties=PROPERTIES_TO_EXPORT]
```

- *FILE_NAME* est le nom du fichier JSON qui stockera les valeurs de propriété exportées. Vous pouvez télécharger le fichier d'export à l'aide de la commande downloadFile. Utilisez la commande uploadFile pour le charger vers l'environnement cible, puis exécutez la commande importARApplicationProperties pour restaurer les paramètres dans l'environnement cible.
- `Properties` (facultatif) représente la liste des propriétés à exporter, séparées par des virgules. Vous pouvez exporter l'ensemble des propriétés suivantes, ou uniquement certaines d'entre elles. Si cette propriété est omise, l'ensemble des propriétés suivantes sont exportées :
  - `Theme` : exporte le thème d'affichage utilisé dans l'environnement.
  - `EmailNotification` : exporte les paramètres de notification par courriel définis dans l'environnement.
  - `DisplayBusinessProcessName` : procède à l'export si le nom du processus métier doit être affiché sur la page dans l'environnement.
  - `RedwoodExperience` : exporte le paramètre d'expérience Redwood de l'environnement.
  - `BackgroundImage` : exporte l'image d'arrière-plan utilisée dans l'environnement.
  - `LogoImage` : exporte l'image de logo utilisée dans l'environnement.

Exemple

Export de l'image de logo et des paramètres de notification par courriel et d'expérience Redwood uniquement à partir d'un environnement :

```
epmautomate exportARApplicationProperties myProp.JSON
Properties=EmailNotification,RedwoodExperience,LogoImage
```

exportBackgroundImage

Exporte dans un fichier JPG l'image d'arrière-plan utilisée dans un environnement Account Reconciliation pour pouvoir l'importer dans un autre environnement.

Champ d'application

Account Reconciliation

Rôles requis

Administrateur de service

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Syntaxe

`epmautomate exportBackgroundImage IMAGE_NAME.jpg`, où *IMAGE_NAME* est le nom du fichier de l'image d'arrière-plan.
Vous pouvez télécharger le fichier de l'image à l'aide de la commande downloadFile. Utilisez la commande uploadFile pour le charger vers l'environnement cible, puis exécutez la commande importBackgroundImage pour l'importer.

Exemple

```
epmautomate exportBackgroundImage corpImage.jpg
```

exportCellLevelSecurity

Exporte les paramètres de sécurité de niveau cellule du processus métier dans un fichier ZIP que vous pouvez télécharger sur un ordinateur local à l'aide de la commande downloadFile.

Champ d'application

Planning, modules Planning, FreeForm, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate exportCellLevelSecurity FILE_NAME.ZIP [names=SECURITY_RECORD_NAMES]` où :

- `FILE_NAME` est le nom du fichier ZIP créé pour le fichier Excel contenant les informations de sécurité de niveau cellule.
- `names` (facultatif) identifie la liste des définitions de sécurité de niveau cellule, séparées par une virgule, dans l'application. Si cette option n'est pas définie, toutes les définitions de sécurité de niveau cellule de l'application sont exportées.

Exemples

- **Export de définitions de sécurité de niveau cellule spécifiques**
  ```
  epmautomate exportCellLevelSecurity ExportCLSDRecordsFile.zip
  names=CLSDAccountPeriod,CLSDEntityPeriod,CLSDProductPeriod
  ```
- **Export de toutes les définitions de sécurité de niveau cellule**
  ```
  epmautomate exportCellLevelSecurity ExportCLSDRecordsFile.zip
  ```

exportConsolidationJournals

Exporte les journaux de consolidation à l'aide d'un job défini dans Financial Consolidation and Close.

Champ d'application

Financial Consolidation and Close

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate exportConsolidationJournals jobName [fileName=`*`FILE_NAME`*`]` où

- `jobName` est le nom du job d'export de journal créé dans Financial Consolidation and Close.
- `fileName` (facultatif) est le nom d'un fichier JLF dans lequel les entrées de journal doivent être exportées. Employez la commande downloadFile pour télécharger ce fichier sur un ordinateur local.

Exemple

```
epmautomate exportConsolidationJournals "JEXPORT1" fileName=Export_Test.jlf
```

exportData

Exporte des données d'application dans un fichier ZIP à l'aide des paramètres d'export de données, y compris le nom de fichier, indiqués dans un job de type `Exporter les données`.

Le fichier de données exporté est stocké à l'emplacement de téléchargement par défaut à partir duquel vous pouvez le télécharger vers votre ordinateur. Utilisez l'explorateur de boîte de réception/d'envoi pour visualiser les détails du fichier exporté.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate exporData` *`JOB_NAME [FILE_NAME]`*, où *`JOB_NAME`* est le nom du job défini dans l'application et *`FILE_NAME`* est le nom du fichier ZIP (facultatif) dans lequel les données doivent être exportées.

Exemple

```
epmautomate exportData dailydataexport dailyData.zip
```

exportDataManagement

Exporte les enregistrements de gestion des données d'un environnement dans un fichier ZIP.

Cette commande exporte dans un fichier ZIP un ensemble complet de données de table intermédiaire et de configuration, y compris les colonnes d'ID, de sorte que les données puissent être importées sans nuire à l'intégrité référentielle.

Le fichier exporté, par exemple `dataFile.zip`, est stocké dans la boîte d'envoi. Vous pouvez télécharger le fichier exporté à l'aide de la commande downloadFile, par exemple : `epmAutomate`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

`downloadFile outbox/dataFile.zip`. Vous pouvez utiliser ce fichier ZIP pour importer les données à l'aide de la commande importDataManagement.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate exportDataManagement` *`FILE_NAME`*`.zip`, où *`FILE_NAME`* est le nom du fichier ZIP dans lequel les données doivent être exportées.

Exemple

```
epmautomate exportDataManagement dataFile.zip
```

exportDimension

Exporte une dimension d'une application Oracle Fusion Cloud Enterprise Data Management vers un fichier de la zone intermédiaire ou, éventuellement, vers un environnement cible défini dans une connexion.

Champ d'application

Cloud EDM

Rôles requis

- Administrateur de service
- Utilisateur (nécessite l'autorisation Gestionnaire de données)

Syntaxe

`epmautomate exportDimension` *`APPLICATION DIMENSION FILE_NAME`* `[connection=`*`NAME`*`]` où :

- `APPLICATION` est le nom d'une application Cloud EDM.
- `DIMENSION` est le nom d'une dimension d'application.
- `FILE_NAME` est le nom du fichier (CSV pour exporter vers un fichier ou ZIP pour exporter vers Oracle Financials Cloud) stockant les données exportées. Si la valeur du paramètre connection n'est pas définie, ce fichier est créé dans la zone intermédiaire. Vous pouvez le télécharger sur un ordinateur local à l'aide de la commande downloadFile ou le copier dans un autre environnement Oracle Fusion Cloud Enterprise Performance Management à l'aide de la commande copyFileFromInstance.
- `connection=`*`NAME`* (facultatif) identifie un nom de connexion (emplacement de l'instance) défini dans Cloud EDM. Si ce paramètre est indiqué, le fichier d'export est chargé vers l'environnement cible (boîte de réception pour Cloud EPM et emplacement de chargement par défaut pour Oracle Financials Cloud).

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Remarque**
>
> Les informations d'identification indiquées dans la définition de connexion doivent disposer des droits d'accès en écriture sur l'environnement cible.

Exemples

- Export vers la zone intermédiaire Cloud EDM : `epmautomate exportDimension USOperations Entity EntityData.CSV`
- Export et chargement vers Oracle Financials Cloud: `epmautomate exportDimension USOperations Entity EntityData.zip Connection=ora_fusion_gl`
- Export et chargement vers la boîte de réception Cloud EPM cible : `epmautomate exportDimension USOperations Entity EntityData.CSV Connection=EPM_cloud_pln`

exportDimensionMapping

Exporte les règles de mapping d'une dimension Oracle Fusion Cloud Enterprise Data Management donnée pour un emplacement afin de créer un fichier de règles de mapping et, éventuellement, charge le fichier exporté vers la boîte de réception Data Management d'un autre environnement Oracle Fusion Cloud Enterprise Performance Management.

Champ d'application

Cloud EDM

Rôles requis

- Administrateur de service
- Utilisateur (avec autorisation de gestionnaire de données)

Syntaxe

`epmautomate exportDimensionMapping` *`APPLICATION DIMENSION LOCATION FILE_NAME`* `[connection=`*`NAME`*`]` où :

- `APPLICATION` est le nom d'une application Cloud EDM.
- `DIMENSION` est le nom d'une dimension d'application.
- `LOCATION` est l'emplacement spécifique pour lequel des règles de mapping doivent être exportées.
- `FILE_NAME` est le nom du fichier CSV où les mappings exportés doivent être stockés. Ce fichier est créé dans la zone intermédiaire si le paramètre `connection` n'est pas défini. Vous pouvez le télécharger sur un ordinateur local à l'aide de la commande downloadFile ou utiliser la commande copyFileFromInstance pour copier le fichier dans un autre environnement Cloud EPM.
- `connection=`*`NAME`* (facultatif) identifie un nom de connexion (emplacement de l'instance) défini dans Cloud EDM. Si ce paramètre est indiqué, la commande charge le fichier exporté vers l'emplacement de chargement par défaut de l'environnement cible.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Remarque**
>
> Les informations d'identification indiquées dans la connexion doivent disposer des droits d'accès en écriture sur l'environnement cible.

Exemples

- Export vers la zone intermédiaire : `epmautomate exportDimensionMapping USOperations Entity Loc1 Loc1Mappings.CSV`
- Export et chargement vers la boîte de réception de l'environnement Cloud EPM cible : `epmautomate exportDimensionMapping USOperations Entity Loc1 Loc1Mappings.CSV Connection=EPM_cloud_pln`

exportEJJournals

Exporte les journaux Enterprise prêts à être imputés de Financial Consolidation and Close dans un fichier ZIP. Ce fichier peut ensuite être utilisé pour importer des données de journal dans un système ERP.

Une fois les journaux exportés dans un fichier d'export, cette commande remplace le statut d'imputation `Ready To Post` de chaque journal exporté par `Post In Progress`.

Champ d'application

Financial Consolidation and Close

Rôles requis

Administrateur de service

Syntaxe

`epmautomate exportEJJournals` *`FILE_NAME`*`.zip [year=`*`YEAR`*`] [period=`*`PERIOD`*`] [OPERATION=posting|validation]`, où :

- *`FILE_NAME`* identifie le fichier ZIP dans lequel les fichiers CSV d'export de journal doivent être archivés. La commande génère un fichier CSV (format de dénomination : `YEAR_PERIOD_JOURNALID_YYYYDDMMHHMMSS.csv`) pour chaque journal et les compresse pour créer ce fichier ZIP.
- *`YEAR`* (facultatif) correspond à l'année de collecte de données pour laquelle les données de journal doivent être exportées. Si aucune valeur n'est indiquée, les données de toutes les années sont exportées.
- *`PERIOD`* (facultatif) correspond à la période de collecte de données pour laquelle les données de journal doivent être exportées. La période ne peut être définie que si une année de collecte de données est indiquée. Si aucune valeur n'est indiquée, les données de toutes les périodes sont exportées.

> **Note**
>
> Si *`YEAR`* et *`PERIOD`* ne sont pas indiqués, cette commande exporte tous les journaux dont le statut d'imputation est `Ready To Post` pour toutes les années et les périodes.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `Operation` (facultatif) est l'opération à exécuter. Les valeurs possibles sont `posting` et `validation`. La valeur par défaut est `posting`.
  `operation=posting` fournit un fichier CSV qui contient les résultats de l'imputation des journaux avec les colonnes suivantes : Année, Période, ID de journal, Statut d'imputation (Imputé/Echec), Message, Code d'erreur et Message d'erreur. Les colonnes Message, Code d'erreur et Message d'erreur sont facultatives. Le statut d'imputation des journaux est mis à jour en fonction du statut fourni dans le fichier. Seuls les journaux dont le statut d'imputation est *Imputation en cours* sont mis à jour.

  `operation = validation` fournit un fichier CSV qui contient les résultats de la validation des journaux avec les colonnes suivantes : Année, Période, ID de journal, Statut de validation (Valide/Echec), Message, Code d'erreur et Message d'erreur. Les colonnes Message, Code d'erreur et Message d'erreur sont facultatives. Le statut de validation des journaux est mis à jour en fonction du statut fourni dans le fichier. Seuls les journaux dont le statut de validation est *Validation en cours* seront mis à jour.

Exemples

- Export de données de journal pour toutes les années et périodes :
  ```
  epmautomate exportEJJournals Journal_Export.zip
  ```
- Export de données de journal pour une année donnée :
  ```
  epmautomate exportEJJournals Journal_Export.zip year=2020
  ```
- Export de données de journal pour une combinaison année/période donnée :
  ```
  epmautomate exportEJJournals Journal_Export.zip year=2024 period=March
  ```
- Export de données de journal pour une combinaison année/période donnée, avec exécution de l'opération de validation :
  ```
  epmautomate exportEJJournals Journal_Export.zip year=2024 period=March
  Operation=validation
  ```

exportEssbaseData

Exporte des données d'un cube d'application vers une archive. Vous pouvez exporter uniquement les données de niveau 0 (cubes ASO et BSO) ou toutes les données du cube (cubes BSO).

Utilisez l'archive exportée afin d'analyser les données Oracle Essbase à la recherche de motifs, par exemple pour améliorer les performances.

> **Note**
>
> - Les fichiers de données compris dans l'archive ont une extension `.txt`.
> - Les listes dynamiques et les objets de commentaire de cellule sont exportés en tant qu'UUID.

L'exécution d'un export de données place le cube en mode lecture seule et empêche toute activité d'écriture pendant que l'opération d'export est en cours.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Strategic Workforce Planning et Sales Planning.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate exportEssbaseData` *`CUBE_NAME FILE_NAME`* `[level=0|All]`, où :

- *`CUBE_NAME`* identifie le cube à partir duquel les données doivent être exportées.
- *`FILE_NAME`* est le nom du fichier .zip qui contiendra les données exportées. Vous pouvez télécharger cette archive en exécutant la commande downloadFile.
- `level` (facultatif) indique le niveau de données à exporter. La valeur par défaut est `0`.
  - **Cubes ASO :** indiquez `0` pour exporter les données de niveau 0. Vous ne pouvez pas utiliser l'option `All`.
  - **Cubes BSO :** indiquez `0` pour exporter les données de niveau 0 ou `All` pour exporter toutes les données.

Exemples

- Exporter toutes les données à partir d'un cube BSO :
  `epmautomate exportEssbaseData Report1 Report1_all_data.zip level=All`
- Exporter les données de niveau 0 à partir d'un cube :
  `epmautomate exportEssbaseData Plan1 Plan1_lvl0_data.zip`

exportJobConsole

Exporte les enregistrements de la console de jobs dans un fichier CSV et crée un fichier ZIP d'export.

Le premier caractère du fichier de sortie CSV est l'indicateur d'ordre des octets `\ufeff`, suivi d'un identificateur d'application crypté entouré de guillemets. L'en-tête du fichier CSV suit l'identificateur d'application.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Strategic Workforce Planning et Sales Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Jobs - Gérer
- Tout rôle prédéfini et rôle d'application Jobs - Afficher

Syntaxe

`epmautomate exportJobConsole` *`FILE_NAME`*`.zip [nDays=`*`NUMBER_OF_DAYS`*`] [jobtypes=`*`JOB_TYPE`*`] [jobStatusCodes=`*`STATUS_CODE`*`] [exportErrorDetails=true|false] [excludeApplicationId=true|false]`, où :

- *`FILE_NAME`* est le nom du fichier ZIP dans lequel seront stockés les enregistrements de la console de jobs. Vous utilisez la commande downloadFile pour télécharger ce fichier à partir d'un environnement.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `nDays` (facultatif) indique le nombre de jours pour lequel exporter les enregistrements de la console de jobs. Valeurs possibles : `all` (tout en minuscules) pour exporter tous les enregistrements de la console de jobs disponibles, `1`, `2`, `7`, `30` et `60`. La valeur par défaut est `7`.
- `jobTypes` (facultatif) correspond à la liste des codes de job, séparés par une virgule, pour lesquels les enregistrements de la console doivent être exportés. La valeur par défaut est `Rules`. Les valeurs valides sont les suivantes :
  - `all` (tout en minuscules)
  - `RULES`
  - `RULESET`
  - `CLEAR_CELL_DETAILS`
  - `COPY_DATA`
  - `INVALID_INTERSECTION_RPT`
  - `COPY_VERSIONS`
  - `CONTENT_UPGRADE`
  - `PLAN_TYPE_MAP`
  - `IMPORT_DATA`
  - `EXPORT_DATA`
  - `EXPORT_METADATA`
  - `IMPORT_METADATA`
  - `CUBE_REFRESH`
  - `CLEAR_CUBE`
  - `ADMIN_MODE`
  - `COMPACT_CUBE`
  - `RESTRUCTURE_CUBE`
  - `MERGE_DATA_SLICES`
  - `OPTIMIZE_AGGREGATION`
  - `SECURITY_IMPORT`
  - `SECURITY_EXPORT`
  - `AUDIT_EXPORT`
  - `JOBCONSOLE_EXPORT`
  - `SORT_MEMBERS`
  - `SMART_PUSH`
  - `IMPORT_EXCHANGE_RATES`
- `jobStatusCodes` (facultatif) correspond à la liste des codes de statut de job, séparés par une virgule, pour lesquels les enregistrements doivent être exportés. La valeur par défaut est `2` (Terminé sans erreurs). Valeurs possibles :
  - `all` (tout en minuscules) pour tous les jobs, quel que soit leur statut
  - `1` - Traitement

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

  - 2 - Terminé sans erreurs
  - 3 - Echec avec des erreurs
  - 4 - Terminé avec un statut inconnu
  - 5 - Terminé avec un statut Violation de seuil
  - 6 - Annulation en attente
  - 7 - Annulé
  - 8 - Terminé avec des erreurs
  - 9 - Terminé avec des avertissements
- *exportErrorDetails* (facultatif) exporte les détails des jobs ayant échoué ou ayant signalé des erreurs dans des fichiers journaux s'il est défini sur `true`. Ce fichier journal d'erreurs est inclus dans le fichier ZIP de sortie. La valeur par défaut est `false`. Si la valeur est définie sur `true`, les détails de statut des jobs présentant l'un des statuts suivants sont exportés.
  - Echec avec des erreurs
  - Terminé avec un statut inconnu
  - Terminé avec un statut Violation de seuil
  - Terminé avec des erreurs
  - Terminé avec des avertissements
- `excludeApplicationId` (facultatif) indique si l'identificateur d'application doit être écrit dans le fichier d'export. La valeur par défaut est `false`.

> **Note**
>
> Les données issues de fichiers exportés qui ne contiennent pas l'identificateur d'application ne peuvent pas être importées dans des environnements Oracle Fusion Cloud Enterprise Performance Management.

Exemples

- Export de tous les enregistrements de la console de jobs disponibles :
  ```
  epmautomate exportJobConsole jobs.zip nDays=all jobTypes=all
  jobStatusCodes=all
  ```
- Export de tous les enregistrements de la console de jobs de règles disponibles :
  ```
  epmautomate exportJobConsole jobs.zip nDays=all jobStatusCodes=all
  ```
- Export de tous les enregistrements de la console de jobs de règles disponibles sans identificateur d'application :
  ```
  epmautomate exportJobConsole jobs.zip nDays=all jobStatusCodes=all
  excludeApplicationId=true
  ```
- Export des enregistrements des jobs de règles terminés au cours des 14 derniers jours uniquement :
  ```
  epmautomate exportJobConsole jobs.zip nDays=14
  ```
- Export des enregistrements de la console et des erreurs des jobs d'import de métadonnées et d'effacement de cube ayant échoué ou réussi avec des erreurs, exécutés au cours des sept derniers jours :

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
epmautomate exportJobConsole jobs.zip jobtypes=IMPORT_METADATA,CLEAR_CUBE
jobStatusCodes=3,8 exportErrorDetails=true
```

exportLibraryArtifact

Exporte les artefacts de la bibliothèque Narrative Reporting. Pour les artefacts de rapport uniquement, cette commande peut éventuellement convertir l'export en fichier LCM que vous pouvez importer dans Financial Consolidation and Close, Planning, les modules Planning ou Tax Reporting.

Une fois l'export effectué, utilisez la commande downloadFile pour télécharger les fichiers d'export et d'erreurs sur un ordinateur local.

Champ d'application

Narrative Reporting

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Un niveau de sécurité supplémentaire doit être accordé aux utilisateurs dotés du rôle Super utilisateur, Utilisateur ou Visualiseur via une liste de contrôle d'accès.

Syntaxe

`epmautomate exportLibraryArtifact` *`ARTIFACT_PATH EXPORT_FILE`* `[exportFormat=Native|File|LCM] [applicationName=`*`APP_NAME`*`] [errorFile=`*`ERROR_FILE`*`.txt]`, où :

- *`ARTIFACT_PATH`* est l'emplacement de l'artefact dans la bibliothèque Narrative Reporting.
- *`EXPORT_FILE`* est le nom unique du fichier dans lequel les artefacts doivent être exportés.
- `exportFormat` (facultatif) est l'un des formats suivants :
  - `Native` exporte des artefacts sous forme de fichier ZIP pouvant être utilisé avec d'autres environnements Narrative Reporting. Il s'agit de la valeur par défaut.
  - `File` exporte les fichiers au format binaire d'origine (PDF, DOCX, ZIP, JPEG, etc.) auquel ils sont disponibles dans Narrative Reporting. Ce paramètre peut être utilisé pour exporter des fichiers binaires uniquement. Il ne doit pas être utilisé avec des artefacts Reports.
  - `LCM` convertit les rapports au format utilisé par Migration et les exporte dans un fichier ZIP pouvant être importé dans des environnements Financial Consolidation and Close, Planning, de modules Planning ou Tax Reporting.
- `applicationName` (facultatif) est le nom de l'application cible dans laquelle vous prévoyez d'importer les rapports. Cette valeur n'est requise que si vous utilisez `LCM` en tant que valeur du paramètre `exportFormat`.
- *`errorFile`* (facultatif) est le nom unique du fichier texte qui stockera les erreurs liées à l'export.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exemples

- Export d'un rapport à son format natif pour qu'il puisse être importé dans un autre environnement Narrative Reporting :
```
epmautomate exportLibraryArtifact "Library/Samples/Sample Report 1"
exp_SampleReport1.doc errorFile=export_errors.txt
```
- Export d'une feuille de calcul à son format binaire d'origine :
```
epmautomate exportLibraryArtifact "Library/Spreadsheets/Sheet1.xlsx"
exp_Sheet1.xlsx exportFormat=File errorFile=export_errors.txt
```
- Export et formatage de rapports pour import dans Financial Consolidation and Close, Planning, les modules Planning ou Tax Reporting :
```
epmautomate exportLibraryArtifact "Library/Samples/Sample Report 1"
exp_SampleReport1.zip exportFormat=LCM applicationName=Vision
errorFile=report_exp_errors.txt
```

exportLibraryDocument

Exporte les documents disponibles dans la bibliothèque Reports vers un fichier.

Vous pouvez télécharger le fichier exporté sur un ordinateur local à l'aide de la commande downloadFile.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate exportLibraryDocument` *`ARTIFACT_PATH`* `[jobName=`*`JOB_NAME`*`] [exportFile=`*`FILE_NAME`*`] [exportFormat=file|zip] [errorFile=`*`FILE_NAME`*`.log] [overWrite=true|false]`, où :

- *`ARTIFACT_PATH`* est l'emplacement ou l'UUID (identifiant unique universel) du document dans la bibliothèque Reports. Pour exporter plusieurs documents, spécifiez une liste d'UUID ou d'emplacements de document, séparés par des virgules.
- *`jobName`* (facultatif) est le nom du job d'export de document de bibliothèque à utiliser pour exporter le document. Le nom de job par défaut est `Copy Artifact From Library`.
- *`EXPORT_FILE`* (facultatif) est le nom unique du fichier d'export.
  Si vous utilisez `exportFormat=ZIP`, le fichier ZIP résultant utilise ce nom. Veillez à utiliser l'extension .zip dans cette valeur.

  Si vous utilisez `exportFormat=file` :

  – Export d'un document : le fichier d'export est créé avec ce nom si l'extension de fichier spécifiée correspond à celle du document.

  – Export de plusieurs documents : la commande ignore cette valeur et exporte chaque document en utilisant son nom d'origine.
- `exportFormat` (facultatif) est l'un des formats suivants :

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

  - `File` exporte les documents au format binaire d'origine (PDF, DOCX, ZIP, JPEG, etc.) auquel ils sont disponibles dans la bibliothèque. Chaque fichier exporté est créé dans la boîte d'envoi à l'aide de son nom d'origine. Il s'agit de la valeur par défaut.
  - `zip` crée un fichier ZIP contenant tous les documents exportés au format binaire d'origine. Si aucune valeur n'est spécifiée pour le paramètre facultatif `exportFile`, le nom du fichier ZIP est automatiquement généré à l'aide du nom de l'utilisateur qui exécute la commande et de l'horodatage. Le format du nom de fichier est *username_exported_artifacts_yyyyMMddHHmm.zip*. Par exemple, *JDoe_exported_artifacts_202410201559.zip*.
- *`errorFile`* (facultatif) est le nom unique du fichier qui stockera les erreurs liées à l'export. Aucun fichier d'erreurs n'est créé si vous n'indiquez pas cette valeur.
- `overwrite` (facultatif) détermine si un fichier portant le même nom existant dans l'emplacement de téléchargement par défaut doit être remplacé. La valeur par défaut est `false`, ce qui signifie que la commande échouera si un fichier portant le même nom existe dans la boîte d'envoi.

Exemples

- Export d'un document :
```
epmautomate exportLibraryDocument Library/folder1/WeeklySales.html
jobName="Copy Weekly Sales" exportFile=WeeklySales.zip
errorFile=WeeklySalesError.log overWrite=true exportFormat=zip
```
- Export de plusieurs documents :
```
epmautomate exportLibraryDocument Library/folder1/WeeklySales.html,Library/
folder2/WeeklySalesReport.pdf jobName="Copy Weekly Sales"
exportFile=WeeklySales.zip errorFile=WeeklySalesError.log overWrite=true
exportFormat=zip
```

exportLogoImage

Exporte dans un fichier JPG le logo d'entreprise utilisé dans un processus métier Account Reconciliation pour pouvoir l'importer dans un autre environnement.

Champ d'application

Account Reconciliation

Rôles requis

Administrateur de service

Syntaxe

`epmautomate exportLogoImage` *`IMAGE_NAME`*`.jpg`, où *`IMAGE_NAME`* est le nom du fichier de l'image de logo.
Vous pouvez télécharger le fichier de logo exporté à l'aide de la commande downloadFile. Utilisez la commande uploadFile pour le charger vers l'environnement cible, puis exécutez la commande importLogoImage.

Exemple

```
epmautomate exportLogoImage corpLogo.jpg
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

exportMapping

Exporte les règles de mapping d'une dimension ou d'un emplacement spécifique afin de créer un fichier de règles de mapping. Vous devez indiquer le nom du fichier et un emplacement dans la boîte de réception (par exemple, `inbox/exportedAccountMap.txt` ou `inbox/france sales/exportedAccountMap.txt`) pour exporter les mappings.

Utilisez la commande downloadFile pour télécharger le fichier de mapping exporté sur un ordinateur local.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate exportMapping` *`DIMENSION_NAME|ALL FILE_NAME LOCATION`*, où :

- *`DIMENSION_NAME|ALL`* est la dimension source à partir de laquelle les mappings doivent être exportés. Indiquez le nom de la dimension à partir de laquelle les mappings doivent être exportés ou `ALL` pour exporter les mappings à partir de toutes les dimensions d'un emplacement.
- *`FILE_NAME`* est le nom unique d'un fichier de mapping et d'un emplacement dans la boîte d'envoi.
- *`LOCATION`* est l'emplacement de la gestion des données pour lequel des règles de mapping doivent être exportées.

Exemples

- `epmautomate exportMapping Account inbox/exportedAccountMap.txt "France Sales"`
- `epmautomate exportMapping ALL "inbox/france sales/exportedAccountMap.txt" "France Sales"`

exportMetadata

Exporte des métadonnées dans un fichier à l'aide des paramètres indiqués dans un job de type `export metadata`. Le fichier contenant les données exportées est stocké dans l'emplacement de téléchargement par défaut à partir duquel vous pouvez le télécharger vers un ordinateur local.

Vous pouvez éventuellement indiquer un nom de fichier pour les données exportées, qui remplace le nom de fichier par défaut (nom de job qui est utilisé pour exporter des métadonnées). Les métadonnées sont exportées sous forme de fichier ZIP uniquement.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Strategic Workforce Planning et Sales Planning.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate exportMetadata JOB_NAME [FILE_NAME]`, où *`JOB_NAME`* est le nom d'un job défini dans l'application et *`FILE_NAME`* est le nom du fichier ZIP dans lequel les métadonnées doivent être exportées.
Employez la commande downloadFile pour télécharger ce fichier sur un serveur local.

Exemple

```
epmautomate exportMetadata dailyAccountexport Accountexport.ZIP
```

exportOwnershipData

Exporte les données de participation d'une entité vers un fichier CSV d'éléments séparés par des virgules.

Les données de participation par défaut renseignées par Financial Consolidation and Close ne sont pas incluses dans le fichier d'export. Seules les données saisies par des utilisateurs pour remplacer les paramètres par défaut y sont incluses.

Champ d'application

Financial Consolidation and Close et Tax Reporting.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur

Syntaxe

`epmautomate exportOwnershipData Entity Scenario Year Period FILE_NAME`, où :

- `Entity` est le nom de l'entité à partir de laquelle les données doivent être exportées.
- `Scenario` est le scénario à partir duquel les données doivent être exportées.
- `Year` est l'année à partir de laquelle les données doivent être exportées.
- `Period` est la période de l'année à partir de laquelle les données doivent être exportées.
- `FILE_NAME` est le nom d'un fichier CSV vers lequel les données doivent être exportées. Employez la commande downloadFile pour télécharger ce fichier sur un serveur local.

Exemple

```
epmautomate exportOwnershipData FCCS_TotalActual FY18 Dec exportfile.csv
```

exportQueryResults

Exécute une requête définie dans une application et exporte les résultats dans un fichier texte.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Le fichier de résultats de requête est stocké dans profitoutbox. Vous pouvez le télécharger à l'aide de la commande downloadFile ou de l'explorateur de fichiers Profitability and Cost Management.

Champ d'application

Profitability and Cost Management

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Syntaxe

`epmautomate exportQueryResults` *`APPLICATION_NAME`* `fileName=`*`FILE_NAME`* `[fileOutputOptions=ZIP_ONLY|ZIP_AND_TEXT|TEXT_ONLY] [queryName=`*`QUERY_NAME`*`] [exportOnlyLevel0Flg=true|false] [roundingPrecision=2] [dataFormat=NATIVE|COLUMNAR] [memberFilters=`*`JSON_FILTER`*`] [includeHeader=true|false] [delimiter="`*`DELIMITER`*`"] [keepDuplicateMemberFormat=true|false]`, où :

- *`APPLICATION_NAME`* est le nom de l'application Profitability and Cost Management pour laquelle exécuter la requête.
- `fileName` est le nom du fichier qui stockera les résultats de requête. La valeur de ce paramètre est requise si la valeur du paramètre `queryName` n'est pas indiquée. Elle est facultative si la valeur du paramètre `queryName` est indiquée, auquel cas le nom de requête est utilisé comme nom du fichier de résultats de requête.
  Le format de données indiqué détermine le format du fichier de sortie. Si vous utilisez `dataFormat=NATIVE` (valeur par défaut), le processus d'export crée un fichier texte. Si vous utilisez `dataFormat=COLUMNAR`, le processus d'export crée plusieurs fichiers texte numérotés de façon séquentielle et les compresse ensuite dans un fichier ZIP.
- `fileOutputOptions` (facultatif) indique le format de sortie du fichier de résultats de requête. La valeur par défaut est `ZIP_ONLY`, qui crée `fileName`.ZIP ou `queryName`.ZIP selon qu'une valeur est indiquée ou non pour le paramètre `fileName`. Les autres options sont `TEXT_ONLY`, pour créer le fichier de sortie sous forme de fichier texte, et `ZIP_AND_TEXT`, pour générer un fichier texte et un fichier ZIP.
- `queryName` est un paramètre facultatif qui identifie une requête définie dans l'application. Les noms de requête qui contiennent un espace doivent être placés entre guillemets doubles.
  N'indiquez pas de nom de requête si vous voulez exporter toutes les données Oracle Essbase appartenant à l'application.

  Les conditions suivantes peuvent entraîner la création d'un fichier de données vide par cette commande :

  - une requête dont le format est incorrect qui n'extrait aucune donnée ;
  - une requête qui génère trop de données. Dans ce scénario, envisagez de réduire la portée de la requête de sorte qu'elle extraie moins de données ou divisez la requête en de plus petites requêtes.
    Reportez-vous à la section Gestion des requêtes Oracle Profitability and Cost Management Cloud du guide *Administration de Profitability and Cost Management*.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `exportOnlyLevel0Flg` (facultatif) indique si la requête ne doit extraire que les données de niveau 0. Indiquez cette valeur de paramètre entièrement en minuscules.
  Ce paramètre est ignoré si vous exportez toutes les données d'application en omettant le nom de la requête.
- `roundingPrecision` (facultatif) indique le nombre de décimales (précision à l'arrondi) à utiliser lors de l'export des résultats de requête. Applicable uniquement lorsque `queryName` est indiqué. La valeur par défaut est 2.
- `dataFormat` (facultatif) indique le format de sortie. Les valeurs valides sont les suivantes :
  - `NATIVE`, qui conserve le résultat de la requête sous forme de données au format natif Essbase. Il s'agit de la valeur par défaut.
  - `COLUMNAR`, qui convertit les données utilisant le format natif Essbase et les classe en colonnes pour en faciliter l'interprétation et l'import dans d'autres applications.
    Cette option exporte toutes les données Essbase et ignore la valeur du paramètre `queryName`. Pour filtrer les données, définissez la valeur du paramètre `memberFilters`.

> **Remarque**
>
> La commande ne tient compte des paramètres facultatifs suivants que si `dataFormat` indique `COLUMNAR`.

- `memberFilters` (facultatif) accepte une chaîne au format JSON pour filtrer par dimension et par membre de niveau 0. Exemple : `"{\"Dim1\":[\"Mem1\"],\"Dim2\":[\"Mem21\",\"Mem22\"]}"`
- `includeHeader` (facultatif) ajoute des noms de dimension en tant qu'en-têtes de colonne. Définissez cette valeur sur `false` pour exclure l'en-tête de colonne. La valeur par défaut est `true`.
- `delimiter` (facultatif) identifie le séparateur à utiliser pour séparer les membres de dimension dans le fichier de résultats de requête. Le séparateur doit être placé entre guillemets doubles. La valeur par défaut est un espace (" ").
- `keepDuplicateMemberFormat` (facultatif) indique si le format de membre doit être imprimé au format de membre dupliqué Essbase, par exemple, `[Account]@[Accoun1]`. Définissez cette valeur sur `false` pour n'imprimer que le nom du membre. La valeur par défaut est `true`.

Exemples

- Export de toutes les données d'application :
  ```
  epmautomate exportQueryResults BksML12 fileName="BksML12_MyQuery1.txt"
  fileOutputOptions=TEXT_ONLY
  ```
- Export des résultats d'une requête spécifique :
  ```
  epmautomate exportQueryResults BksML12 queryName="My Product Query"
  roundingPrecision=3
  ```
- Export des données de niveau 0 au format NATIVE :
  ```
  epmautomate exportQueryResults BksML30 fileName="BksML30_ExportLevel0-Data"
  fileOutputOptions=ZIP_AND_TEXT exportOnlyLevel0Flg=true
  ```
- Export des données de niveau 0 au format COLUMNAR avec une seule dimension et un seul filtre de membre :
  ```
  epmautomate exportQueryResults BksML30 fileName="BksML30_Level0-Data"
  dataFormat="COLUMNAR" memberFilters="{\"Period\":[\"December\"]}"
  includeHeader="true" delimiter="," roundingPrecision="3"
  ```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Export des données de niveau 0 au format COLUMNAR avec une seule dimension et plusieurs filtres de membre :
```
epmautomate exportQueryResults BksML30 fileName="BksML30_Level0-Data"
dataFormat="COLUMNAR" memberFilters="{\"Period\":[\"November\",\"December\"]}"
includeHeader="true" delimiter="," roundingPrecision="3"
```
- Export des données de niveau 0 au format COLUMNAR avec plusieurs dimensions et plusieurs filtres de membre :
```
epmautomate exportQueryResults BksML30 fileName="BksML30_Level0-Data"
dataFormat="COLUMNAR" memberFilters="{\"Year\":[\"2016\"],\"Period\":
[\"November\",\"December\"]}" includeHeader="true" delimiter=","
roundingPrecision="3"
```

exportSnapshot

Répète une opération d'export exécutée précédemment afin de créer un instantané du contenu de Migration.

A l'aide de Migration, sélectionnez et exportez les artefacts souhaités vers un instantané, par exemple, `January16FullApp`. Utilisez le nom de l'instantané avec cette commande afin de répéter par la suite l'opération d'export, qui exportera uniquement les artefacts sélectionnés lors de l'opération d'export d'origine. Reportez-vous à la section Export d'artefacts et d'application du guide *Administration de Migration*.

- Les données suivantes ne font pas partie des instantanés d'application Planning, modules Planning et FreeForm :
  - Données d'audit
  - Données de console de job

  Utilisez la commande cloneEnvironment ou la fonctionnalité de clonage d'environnement si vous souhaitez copier les données d'audit et de console de job dans l'environnement cible.
- Les instantanés ne contiennent pas les données de table intermédiaire de gestion des données. Pour importer ces données, utilisez les commandes exportDataManagement et importDataManagement ou l'interface de scripts de maintenance système de gestion des données. Vous pouvez utiliser la commande cloneEnvironment ou la fonctionnalité de clonage d'environnement pour créer une copie identique de l'environnement, y compris des données de table intermédiaire de gestion des données.

Vous pouvez télécharger l'instantané exporté à partir de l'emplacement par défaut à l'aide de la commande downloadFile.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Syntaxe

`epmautomate exportSnapshot` *`SNAPSHOT_NAME`*, où *`SNAPSHOT_NAME`* est le nom d'un instantané existant dans Migration. Cet instantané est remplacé par le nouvel instantané.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exemple

```
epmautomate exportSnapshot January16FullApp
```

exportTemplate

Exporte une application en tant que modèle dans un fichier .ZIP. Le fichier exporté est stocké dans profitoutbox.

Vous pouvez télécharger le fichier exporté sur un ordinateur local à l'aide de la commande downloadFile.

Champ d'application

Profitability and Cost Management

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate exportTemplate` *`APPLICATION_NAME File_Name`*, où :

- *`APPLICATION_NAME`* est le nom de l'application Profitability and Cost Management que vous voulez exporter en tant que modèle.
- `File_Name` est le nom du fichier de modèle.

Exemple

```
epmautomate exportTemplate BksML12 template1
```

exportTaskManagerAccessControl

Exporte le rapport détaillé sur les utilisateurs pour les affectations d'utilisateur Task Manager, Supplemental Data et Journal Enterprise. Le rapport au format Excel ou CSV contient des informations sur les utilisateurs dotés de rôles prédéfinis dans l'environnement et répertorie les attributs de chaque utilisateur (par exemple, leur nom et leur adresse électronique), ainsi que leur statut, leurs équipes, leurs rôles prédéfinis, leurs rôles de workflow, leur organisation, leurs groupes et leur dernier horodatage de connexion.

Exemple de rapport sur le contrôle d'accès à Task Manager :

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Name	User Login	Status	Teams	Email	Role	Workflow	Task Assignee	Task Approver	Organizatic	Last Login	Form Preparer	Form Approver	Form Integrator	Groups
2	John Doe	john.doe@example.com	Available	Admins	john.doe@example.com	Administrator									Service Administrator
3	Jane Doe	jane.doe@example.com	Available		jane.doe@example.com	Power User									Power User
4	user Ats	ats_user	Available		example1@example.com	User									User
5	user2 Ats	ats_user2	Available		example2@example.com	user									User
6	View User	viewUser	Available		view.user@exaple.com	Viewer									Viewer

Champ d'application

Enterprise Profitability and Cost Management, Financial Consolidation and Close et Tax Reporting

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service

Syntaxe

`epmAutomate exportTaskManagerAccessControl` *`REPORT_NAME`*, où *`REPORT_NAME`* est le nom du fichier d'export (extension CSV or XLS valide comprise) dans lequel figurera le rapport.

Ce rapport peut être généré au format CSV ou XLS. Vous pouvez le télécharger à l'aide de la commande downloadFile.

Exemples

- `epmAutomate exportTaskManagerAccessControl aclreport.csv`
- `epmAutomate exportTaskManagerAccessControl aclreport.xls`

exportValidIntersections

Exporte les groupes de croisements valides du processus métier dans un fichier ZIP que vous pouvez télécharger sur un ordinateur local à l'aide de la commande downloadFile. Les croisements valides sont des interactions de cellules filtrées en fonction de règles définies, appelées règles de croisement valide. Celles-ci filtrent certains croisements de cellules sur les utilisateurs lorsqu'ils saisissent des données ou sélectionnent des invites d'exécution.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate exportValidIntersections` *`FILE_NAME`*`.zip [names=`*`INTERSECTION_NAMES`*`]` où :

- *`FILE_NAME`* est le nom du fichier ZIP d'export. Tous les croisements valides identifiés dans la commande sont exportés dans un fichier Microsoft Excel, puis compressés pour créer ce fichier.
- `names` (facultatif) identifie la liste des croisements valides, séparés par une virgule, que vous voulez exporter. Si aucune valeur n'est indiquée pour ce paramètre, la commande exporte tous les croisements valides dans l'application.

Exemples

- **Export de croisements valides spécifiques**
  ```
  epmautomate exportValidIntersections VI_export_File.zip
  names=VIAccountPeriod,VIEntityPeriod,VIProductPeriod
  ```
- **Export de tous les croisements valides**
  ```
  epmautomate exportValidIntersections VI_export_File.zip
  ```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

extractDimension

Extrait une dimension Oracle Fusion Cloud Enterprise Data Management vers un fichier ou vers une connexion globale.

Champ d'application

Cloud EDM

Rôles requis

- Administrateur de service
- Utilisateur (avec autorisation de gestionnaire de données)

Syntaxe

`epmautomate extractDimension` *`APPLICATION DIMENSION EXTRACT_NAME FILE_NAME`* `[connection=`*`NAME`*`]` où :

- `APPLICATION` est le nom d'une application Cloud EDM.
- `DIMENSION` est le nom de la dimension à extraire.
- `EXTRACT_NAME` est le nom du profil d'extraction défini dans l'application. Ce profil est utilisé pour extraire la dimension.
- `FILE_NAME` est le nom du fichier (CSV pour exporter vers un fichier ou ZIP pour exporter vers Oracle Financials Cloud) stockant les données extraites. Si la valeur du paramètre connection n'est pas définie, ce fichier est créé dans la zone intermédiaire. Vous pouvez le télécharger sur un ordinateur local à l'aide de la commande downloadFile ou le copier dans un autre environnement Cloud EDM à l'aide de la commande copyFileFromInstance.
- `connection=`*`NAME`* (facultatif) identifie un nom de connexion globale (emplacement de l'instance) défini dans Cloud EDM comme emplacement du fichier. Si ce paramètre est indiqué, le fichier d'extraction est chargé vers l'environnement cible (boîte de réception pour Oracle Fusion Cloud Enterprise Performance Management et compte de document spécifié pour Oracle ERP).

> **Note**
>
> Les informations d'identification indiquées dans la connexion globale doivent disposer des droits d'accès en écriture sur l'environnement cible.

Exemples

- Extraction vers la zone intermédiaire Cloud EDM : `epmautomate extractDimension USOperations Entity EntityExtProfile EntityData.CSV`
- Extraction et chargement vers Oracle ERP : `epmautomate extractDimension USOperations Entity EntityExtProfile EntityData.zip Connection=ora_fusion_gl`
- Extraction et chargement vers la boîte de réception Cloud EPM cible : `epmautomate extractDimension USOperations Entity EntityExtProfile EntityData.CSV Connection=EPM_cloud_pln`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

extractPackage

Exécute un package d'extractions comportant plusieurs extractions pour une application en une seule opération. Les résultats de chaque extraction du package sont ajoutés à un fichier ZIP, qui peut être écrit dans la zone intermédiaire ou dans une connexion globale.

Champ d'application

Oracle Fusion Cloud Enterprise Data Management

Rôles requis

- Administrateur de service
- Utilisateur (avec autorisation de gestionnaire de données)

Syntaxe

`epmautomate extractPackage` *`APPLICATION_NAME EXTRACT_PACKAGE_NAME FILE_NAME`*`.zip [connection=`*`CONNECTION_NAME`*`]`, où :

- *`APPLICATION_NAME`* est le nom d'une application Cloud EDM.
- *`EXTRACT_PACKAGE_NAME`* est le nom du package défini dans l'application Cloud EDM.
- *`FILE_NAME`* est le nom du fichier ZIP d'extraction qui contiendra les résultats de l'extraction. Ce fichier est créé dans la zone intermédiaire ou écrit dans une connexion globale si un nom de connexion est spécifié.
  Un package peut contenir un ou plusieurs profils d'extraction, chacun portant un nom distinct. Le fichier ZIP du package d'extractions contiendra les fichiers de sortie des extractions du package. Le nombre de fichiers dans le fichier ZIP est déterminé par les paramètres du package.
- *`CONNECTION_NAME`* (facultatif) est le nom de la connexion globale (emplacement de l'instance) défini dans Cloud EDM. Il identifie l'emplacement dans lequel le fichier ZIP contenant les résultats de l'extraction doit être chargé.
  Si un nom de connexion est spécifié, le fichier ZIP contenant les données extraites est chargé vers l'environnement cible (boîte de réception pour un environnement Oracle Fusion Cloud Enterprise Performance Management ou compte de document spécifié pour Oracle ERP). Si aucune valeur n'est précisée, le fichier d'export est créé dans la boîte d'envoi par défaut (zone intermédiaire) de l'environnement Cloud EDM.

Exemples

- Extraction du package COARedesignMaps vers un fichier dans la zone intermédiaire Cloud EDM :

  ```
  epmautomate extractPackage FinancialsCloud COARedesignMaps COARedesign.zip
  ```

- Extraction du package COARedesignMaps vers un fichier et chargement vers une boîte de réception Oracle Fusion Cloud EPM à l'aide d'une connexion :

  ```
  epmautomate extractPackage CorporateAccount COARedesignMaps
  COARedesign.zip Connection=EPM_cloud_pln
  ```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

feedback

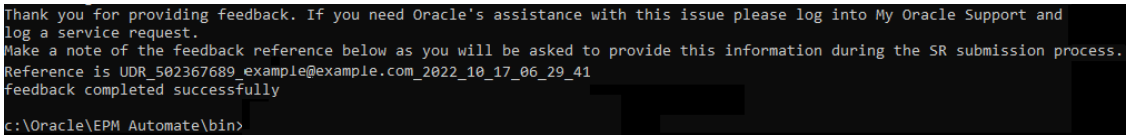
Envoie un commentaire à Oracle et aux administrateurs de service de l'environnement, et charge automatiquement les fichiers journaux EPM Automate créés au cours des dernières 24 heures à partir du répertoire en cours.

Vous pouvez éventuellement charger des fichiers supplémentaires (par exemple des fichiers de trace Fiddler) à partir desquels le support technique Oracle pourra diagnostiquer la raison du problème en cours.

Cette commande, qui imite la fonctionnalité Fournir des commentaires du service, est particulièrement utile pour envoyer des commentaires à Oracle lorsque l'interface utilisateur ne répond plus ou lorsque vous rencontrez un problème lors de l'exécution d'EPM Automate.

Pour plus d'informations sur la fonctionnalité Fournir des commentaires, reportez-vous à la section Aider Oracle à collecter des informations de diagnostic à l'aide de l'utilitaire Fournir des commentaires dans le guide *Guide de mise en route pour les administrateurs*.

La commande feedback renvoie un message semblable au suivant, vous informant qu'elle ne crée pas de demande de service. Si vous avez besoin de l'aide d'Oracle pour résoudre un problème, vous devez créer une demande de service. La commande affiche une référence UDR que vous devez inclure dans la demande de service que vous créez.



Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Syntaxe

`epmautomate feedback "Comment" [Screenshot="FILE_PATH"] [File="FILE_PATH"]` où :

- `Comment` est le texte qui décrit le problème faisant l'objet du commentaire soumis. Les commentaires doivent être placés entre guillemets.
- `Screenshot` (facultatif) identifie le nom d'un fichier graphique qui illustre le problème faisant l'objet du commentaire soumis. Vous pouvez ajouter plusieurs captures d'écran en répétant ce paramètre et la valeur autant de fois que besoin.
- `File` (facultatif) identifie le nom d'un fichier à partir duquel le support technique Oracle pourra résoudre le problème en cours. Utilisez ce paramètre pour soumettre des fichiers

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

de trace Fiddler ou d'autres fichiers à Oracle. Vous pouvez ajouter plusieurs fichiers en répétant ce paramètre et la valeur autant de fois que besoin.

Exemples

- Windows : `epmautomate Feedback "runplantypemap CampaignToReporting ClearData=True did not clear data from aggregate storage" Screenshot=C:/feedback/issue.jpg File=exampleScript.ps1 file=trace.har`
- Linux : `epmautomate Feedback "runplantypemap CampaignToReporting ClearData=True did not clear data from aggregate storage" Screenshot=/scratch/screens/issue.jpg File=/home/feedback/trace.har`

getApplicationAdminMode

Vérifie si l'application est en mode d'administration avec un accès limité uniquement aux administrateurs de service.

Cette commande, qui renvoie `true` si l'application est en mode d'administration et `false` dans le cas contraire, permet de vérifier le statut de l'application avant d'exécuter des scripts d'automatisation. Par exemple, la commande refreshCube nécessite que l'application soit en mode d'administration. Vous pouvez utiliser cette commande dans le script d'automatisation comme suit pour vérifier si l'application est en mode d'administration.

```
adminMode = ‘epmautomate.sh getApplicationAdminMode’
        if ["$adminMode" == "true"]
              epmautomate.sh refreshCube
```

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Account Reconciliation, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de service

Syntaxe

```
epmautomate getApplicationAdminMode
```

Exemple

```
epmautomate getApplicationAdminMode
```

getDailyMaintenanceStartTime

Affiche, dans la console, l'heure UTC à laquelle le démarrage de la maintenance quotidienne de l'environnement est planifié, ou éventuellement le fuseau horaire.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Syntaxe

`epmautomate getDailyMaintenanceStartTime [timezone=true|false]`, où `timezone=true` (facultatif) indique si l'heure de début de la maintenance quotidienne doit être affichée dans le fuseau horaire indiqué lors de sa définition, par exemple : `America/Los_Angeles`. La valeur par défaut est `false`.

Exemples

- Affichage de l'heure de maintenance dans le fuseau horaire indiqué lors de sa définition :
  `epmautomate getDailyMaintenanceStartTime timezone=true`
- Affichage de l'heure de maintenance au format UTC :
  `epmautomate getDailyMaintenanceStartTime`

getEssbaseQryGovExecTime

Affiche le délai maximal en cours, en secondes, qu'une requête Oracle Essbase peut utiliser pour extraire et envoyer les informations avant que la requête soit interrompue.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Profitability and Cost Management, Enterprise Profitability and Cost Management, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de service

Syntaxe

```
epmautomate getEssbaseQryGovExecTime
```

Exemple de sortie de commande :

```
c:\Oracle\EPM Automate\bin>epmautomate getEssbaseQryGovExecTime
300

c:\Oracle\EPM Automate\bin>epmautomate setEssbaseQryGovExecTime 600
setEssbaseQryGovExecTime completed successfully


c:\Oracle\EPM Automate\bin>epmautomate getEssbaseQryGovExecTime
600
```

Exemple

```
epmautomate getEssbaseQryGovExecTime
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

getIdleSessionTimeout

Affiche le délai d'expiration de session (en minutes) de l'environnement Oracle Fusion Cloud Enterprise Performance Management. Lorsqu'une session atteint ce délai d'inactivité, les utilisateurs sont redirigés vers la page de connexion.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

```
epmautomate getIdleSessionTimeout
```

Exemple de sortie de commande :

```
c:\Oracle\EPM Automate\bin>epmautomate getIdleSessionTimeout
75
```

getIPAllowlist (désapprouvé)

Affiche les adresses IP et les CIDR (Classless Inter-Domain Routings) inclus dans la liste d'autorisation actuelle.

> **Note**
>
> Cette commande est désapprouvée et sera enlevée dans une mise à jour future. Au lieu d'utiliser une liste d'autorisation d'adresses IP par environnement, configurez un périmètre réseau dans Oracle Identity Cloud Service afin de limiter les connexions à des adresses IP ou des plages spécifiques pour l'ensemble du domaine. Reportez-vous aux sources d'information suivantes :
>
> - Configuration d'un périmètre réseau dans le guide *Guide de mise en route pour les administrateurs*
> - Gestion des périmètres réseau Oracle Identity Cloud Service dans le guide *Administration d'Oracle Identity Cloud Service*

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service

Syntaxe

```
epmAutomate getIPAllowlist
```

Exemple

Affichez les adresses IP et les CIDR inclus dans la liste d'autorisation actuelle :

```
epmAutomate getIPAllowlist
```

getRestrictedDataAccess

Indique si l'environnement est configuré pour empêcher les administrateurs de service de consentir à la soumission d'un instantané d'application à Oracle lorsqu'ils se servent de l'utilitaire Fournir des commentaires.

Cette commande indique `true` si l'environnement est configuré pour empêcher les administrateurs de service de consentir à la soumission de l'instantané d'application, et `false` dans le cas contraire.

Champ d'application

Account Reconciliation, Oracle Fusion Cloud Enterprise Data Management, Enterprise Profitability and Cost Management, Financial Consolidation and Close, FreeForm, Planning, modules Planning, Profitability and Cost Management, Narrative Reporting, Sales Planning, Strategic Workforce Planning et Tax Reporting.

Rôles requis

Administrateur de service

Syntaxe

Pour modifier le statut en cours de restriction d'accès aux données, utilisez setRestrictedDataAccess.

Exemple

```
epmautomate getRestrictedDataAccess
```

getSubstVar

Extrait les valeurs des variables de substitution et les affiche à l'écran.

Le format d'affichage est le suivant : *`CUBE_NAME.SUBSTVAR`*`=value`, par exemple, `Plan2.CurYear=2016`. Les valeurs des variables de substitution de niveau application sont affichées au format *`ALL.SUBSTVAR`*`=value`, par exemple `ALL.CurYear=2016`

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate getSubstVar` *`CUBE_NAME`*`|ALL [`*`name=VARIABLE_NAME`*`]` où :

- *`CUBE_NAME`* est le cube (par exemple, Plan1, Plan2) duquel vous voulez extraire la variable de substitution. Utilisez `ALL` pour extraire des variables de substitution au niveau de l'application.
- *`name=VARIABLE_NAME`* identifie éventuellement la variable de substitution pour laquelle vous voulez extraire une valeur. Si vous n'indiquez pas de nom de variable, la commande extrait la valeur de toutes les variables de substitution.

Exemples

- Obtention de la valeur de toutes les variables de substitution au niveau du cube et de l'application : `epmautomate getSubstVar ALL`
- Obtention de la valeur d'une variable de substitution spécifique au niveau de l'application : `epmautomate getSubstVar ALL name=CurYear`
- Obtention de la valeur de toutes les variables de substitution au niveau du cube : `epmautomate getSubstVar Plan2`
- Obtention de la valeur d'une variable de substitution spécifique au niveau du cube : `epmautomate getSubstVar Plan2 name=CurYear`

getVirusScanOnFileUploads

Vérifie si votre environnement est autorisé à analyser tous les fichiers en cours de chargement afin de s'assurer qu'ils ne contiennent aucun virus.

Cette commande vérifie si l'analyse antivirus est appliquée avant que les fichiers soient chargés vers l'environnement.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate getVirusScanOnFileUploads`
Cette commande imprime `true` si l'environnement est autorisé à analyser les fichiers chargés à la recherche de virus. Sinon, elle imprime `false`.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

groupAssignmentAuditReport

Crée un rapport répertoriant les utilisateurs et les groupes qui ont été affectés aux groupes de contrôle d'accès ou enlevés de ceux-ci (dont l'affectation a été annulée) pour une plage de dates indiquée. Ce rapport peut être généré pour 120 jours au maximum.

Ce rapport, qui est généré sous forme de fichier CSV, peut être utilisé afin de prendre en charge les opérations d'audit de sécurité. Chaque ligne du fichier CSV généré indique l'utilisateur ou le groupe qui a été ajouté ou enlevé, le groupe auquel l'utilisateur ou le groupe a été ajouté ou duquel il a été enlevé, le type de groupe, l'administrateur de service qui a réalisé l'action, ainsi que la date et l'heure auxquelles l'action a été effectuée.

	A	B	C	D	E	F	G
1	**User/Group Name**	**User/Group Type**	**Action**	**Group**	**Group Type**	**Performed By**	**Date and Time**
2	Reviewer	EPM Group	Assigned	Strategic Planner	EPM	john.smith@example.com	April 07, 2025 09:40:03 UTC
3	Super User	EPM Group	Assigned	exampleGroup	EPM	john.smith@example.com	April 07, 2025 09:41:03 UTC
4	IDCS_Group1	IDCS Group	Assigned	Group1	EPM	john.smith@example.com	April 07, 2025 09:42:02 UTC
5	Power User	Predefined Group	Unassigned	Group1	EPM	john.smith@example.com	April 07, 2025 09:55:05 UTC
6	jDoe	User	Unassigned	Group1	EPM	john.smith@example.com	April 07, 2025 09:50:05 UTC

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Strategic Workforce Planning et Sales Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Afficher

Syntaxe

`epmAutomate groupAssignmentAuditReport` *`FROM_DATE TO_DATE REPORT_NAME`*, où :

- *`FROM_DATE`* est la date de début de la période, au format `YYYY-MM-DD`, pour laquelle le rapport doit être généré.
- *`TO_DATE`* est la date de fin de la période, au format `YYYY-MM-DD`, pour laquelle le rapport doit être généré.
- *`REPORT_NAME`* est le nom du fichier CSV pour le rapport. Vous pouvez télécharger ce rapport généré à l'aide de la commande downloadFile.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Note**
>
> Oracle Fusion Cloud Enterprise Performance Management s'assure que seule une plage de dates valide est utilisée lors de la génération du rapport. Les validations suivantes sont effectuées pour les dates *de début* et *de fin* :
>
> - La date *FROM_DATE* ne peut pas être antérieure à la période de conservation maximale autorisée (120 jours) à compter de la date actuelle.
> - La date *TO_DATE* ne peut pas être postérieure à la période de conservation maximale à compter de la date *FROM_DATE*.
> - La date *TO_DATE* ne peut pas être antérieure à la date *FROM_DATE*.

Exemple

```
epmAutomate groupAssignmentAuditReport 2022-03-01 2022-05-01
GroupAssignmentReport.CSV
```

help

Affiche l'aide pour toutes les commandes EPM Automate ou une commande spécifique.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Syntaxe

`epmautomate [COMMAND_NAME] help`, où *COMMAND_NAME* (facultatif) est le nom de la commande pour laquelle afficher l'aide dans la console.

Exemples

- Affichage d'une page Web à partir de laquelle obtenir de l'aide pour toutes les commandes :
  `epmautomate help`
- Affichage de l'aide sur la commande de cryptage dans la console :
  `epmautomate encrypt help`

importAppAudit

Importe les enregistrements d'audit de données d'un fichier ZIP créé en exportant les données d'audit d'un environnement.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Vous créez le fichier d'import à l'aide de la commande exportAppAudit (`epmautomate exportAppAudit auditData ndays=All`). Utilisez cette commande pour cloner des enregistrements d'audit d'un environnement vers un autre lors de la migration ou du clonage à des fins de récupération après sinistre.

Champ d'application

Planning, modules Planning, FreeForm, Enterprise Profitability and Cost Management, Strategic Workforce Planning Cloud et Sales Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Audit - Gérer

Syntaxe

`epmautomate importAppAudit FILE_NAME [logFilename=LOG_FILE_NAME]`, où :

- `FILE_NAME` est le nom du fichier ZIP contenant les enregistrements d'audit de données que vous voulez importer dans l'application. Avant d'exécuter cette commande, utilisez la commande uploadFile pour charger le fichier vers l'environnement.
- `logFileName` (facultatif) identifie le journal d'erreurs dans lequel seront enregistrées les erreurs rencontrées au cours de l'import. Si cette valeur n'est pas indiquée, la commande génère un fichier d'erreurs nommé selon la convention suivante : `username_date_timestamp`. Vous pouvez télécharger ce fichier à l'aide de la commande downloadFile.

Exemple

```
epmautomate importAppaudit Audit_data.zip logFileName=auditImportLog
```

importAppSecurity

Charge les autorisations d'accès des utilisateurs ou des groupes d'une application à partir d'un fichier CSV présent dans la boîte de réception.

L'import des autorisations d'accès remplace les affectations existantes uniquement pour les membres, formulaires, dossiers de formulaires, listes de tâches, règles métier et dossiers de règles métier de Calculation Manager ayant été importés. Toutes les autres autorisations d'accès restent inchangées.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate importAppSecurity ACL_FILE_NAME ERROR_FILE [clearall=true|false]` où :

- `ACL_FILE_NAME` est le nom du fichier CSV contenant les autorisations d'accès que vous souhaitez importer dans l'application. Avant d'exécuter cette commande, utilisez la

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

commande uploadFile pour charger ce fichier vers la boîte de réception. L'image suivante représente un exemple du contenu d'un fichier d'entée :

	A	B	C	D	E	F	G	H
1	Object Name	Name	Parent	Is User	Object Type	Access Type	Access Mode	Remove
2	AllAB	Interactive User		N	SL_DIMENSION	READWRITE	@IDESCENDANTS	N
3	FormsB	Interactive User	/	N	SL_FORMFOLDER	READWRITE	@IDESCENDANTS	N
4	Statistics	Interactive User		N	SL_DIMENSION	READWRITE	@IDESCENDANTS	N
5	TD	Interactive User		N	SL_DIMENSION	READWRITE	@IDESCENDANTS	N
6	No Entity	Interactive User		N	SL_DIMENSION	READWRITE	MEMBER	N

Pour obtenir une description des en-têtes de colonne et des valeurs possibles, reportez-vous à la section Import de la sécurité du guide *API REST pour Oracle Fusion Cloud EPM*.

- `ERROR_FILE` est le nom d'un fichier CSV qu'EPM Automate crée dans la boîte d'envoi pour enregistrer les erreurs détectées lors de cette opération. Vous pouvez télécharger ce fichier vers un ordinateur local pour analyser et corriger les erreurs signalées. L'image suivante représente un exemple du contenu d'un fichier d'erreur. Les colonnes de ce fichier correspondent aux colonnes d'en-tête du fichier d'entrée :

	A	B	C	D	E	F	G	H
1	AllAB	Interactive User		N	SL_DIMENSION	READWRITE	@IDESCENDANTS	N
2	FormsB	Interactive User	/	N	SL_FORMFOLDER	READWRITE	@IDESCENDANTS	N

- `clearall` (facultatif) indique si les autorisations d'accès existantes doivent être supprimées avant le chargement des nouvelles autorisations à partir du fichier. La valeur par défaut est `false`.

Exemple

```
epmautomate importAppSecurity Acl_file.CSV Acl_import_error.CSV clearall=true
```

importARApplicationProperties

Importe dans un environnement Account Reconciliation les paramètres d'application (expérience Redwood, thème, notifications par courriel et nom du processus métier) ainsi que les images de logo et d'arrière-plan disponibles dans un fichier d'export JSON.

Champ d'application

Account Reconciliation

Rôles requis

Administrateur de service

Syntaxe

`epmautomate importARApplicationProperties` *`FILE_NAME`*, où *`FILE_NAME`* est le nom du fichier JSON exporté à partir d'un environnement.

Ce fichier, exporté à partir d'un autre environnement à l'aide de la commande exportARApplicationProperties, doit être disponible dans l'environnement dans lequel vous restaurez les paramètres d'application.

Exemple

```
epmautomate importARApplicationProperties myProp.JSON
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

importBackgroundImage

Importe l'image d'arrière-plan dans un environnement Account Reconciliation à partir d'un fichier d'export.

Champ d'application

Account Reconciliation

Rôles requis

Administrateur de service

Syntaxe

`epmautomate importBackgroundImage` *`FILE_NAME`*`.jpg`, où *`FILE_NAME`* est le nom du fichier d'image d'arrière-plan exporté à partir d'un autre environnement.

Exemple

```
epmautomate importBackgroundImage image_file.jpg
```

importBalances

Utilise la gestion des données pour importer les données de soldes à partir d'une définition de chargement des données.

Champ d'application

Account Reconciliation.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Un niveau de sécurité supplémentaire doit être accordé aux utilisateurs dotés du rôle Super utilisateur, Utilisateur ou Visualiseur via une liste de contrôle d'accès.

Syntaxe

`epmautomate importBalances` *`DL_DEFINITION PERIOD`*, où :

- *`DL_DEFINITION`* est une définition de chargement des données existante dans Account Reconciliation.
- *`PERIOD`* est le nom d'une période.

Exemple

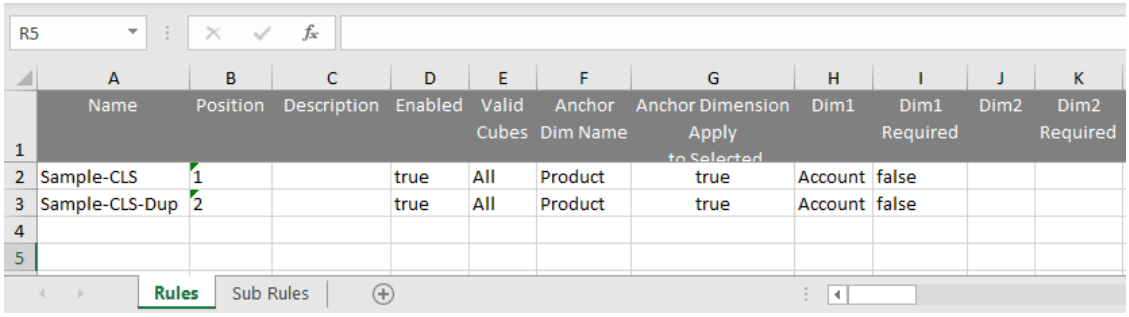
```
epmautomate importBalances DailyLoad "January 2020"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

importCellLevelSecurity

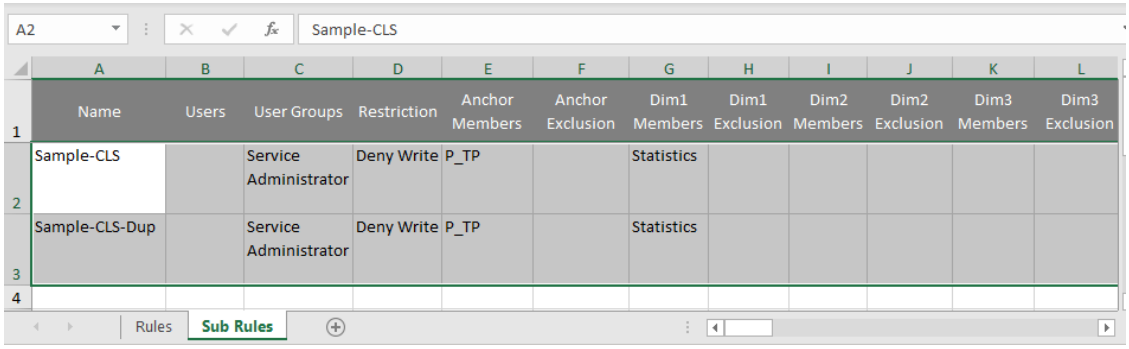
Importe des paramètres de sécurité de niveau cellule à partir d'un fichier ZIP contenant un fichier Excel avec des enregistrements de sécurité de niveau cellule vers le processus métier. Avant d'exécuter cette commande, utilisez la commande uploadFile pour charger le fichier d'import vers l'environnement.

Le fichier ZIP d'import doit contenir un fichier Excel avec deux feuilles de calcul (Rules et Sub Rules) pour pouvoir importer la sécurité de niveau cellule. La feuille Rules doit comporter des définitions de sécurité de niveau cellule, dimensions comprises, et des propriétés telles que Les membres non spécifiés sont valides et Dimensions supplémentaires requises. La feuille Sub Rules doit comporter des sélections de membres et des exclusions. La meilleure méthode pour obtenir le modèle de format de fichier d'import est d'exporter la sécurité de niveau cellule à partir de l'application. Les illustrations suivantes présentent un exemple de format.



	A	B	C	D	E	F	G	H	I	J	K
1	Name	Position	Description	Enabled	Valid Cubes	Anchor Dim Name	Anchor Dimension Apply to Selected	Dim1	Dim1 Required	Dim2	Dim2 Required
2	Sample-CLS	1		true	All	Product	true	Account	false		
3	Sample-CLS-Dup	2		true	All	Product	true	Account	false		
4											
5											

Rules | Sub Rules



	A	B	C	D	E	F	G	H	I	J	K	L
1	Name	Users	User Groups	Restriction	Anchor Members	Anchor Exclusion	Dim1 Members	Dim1 Exclusion	Dim2 Members	Dim2 Exclusion	Dim3 Members	Dim3 Exclusion
2	Sample-CLS		Service Administrator	Deny Write	P_TP		Statistics					
3	Sample-CLS-Dup		Service Administrator	Deny Write	P_TP		Statistics					
4												

Rules | Sub Rules

Champ d'application

Planning, modules Planning, FreeForm, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate importCellLevelSecurity` *`FILE_NAME`*`.ZIP [ErrorFile=`*`FILE_NAME`*`.txt]` où :

- `FILE_NAME` est le nom du fichier ZIP contenant le fichier Excel qui comporte les informations de sécurité de niveau cellule.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `ErrorFile` (facultatif) identifie le nom du fichier texte dans lequel les enregistrements d'erreurs sont écrits. Si aucune valeur n'est indiquée pour ce paramètre, la commande génère automatiquement un fichier d'erreur dont vous pouvez voir le nom dans la console de jobs.
  Utilisez la commande downloadFile pour télécharger le fichier d'erreurs sur un ordinateur local.

Exemple

```
epmautomate importCellLevelSecurity ImportCLSDRecordsFile.zip
ErrorFile=ImportCLSDRecords_errors.txt
```

importConsolidationJournals

Importe les journaux de consolidation à partir d'un fichier JLF dans Financial Consolidation and Close.

- Utilisez la commande exportConsolidationJournals afin de créer le fichier JLF utilisé comme entrée pour cette commande.
- Avant d'exécuter cette commande, utilisez la commande uploadFile pour charger le fichier d'entrée dans l'environnement.

Champ d'application

Financial Consolidation and Close

Rôles requis

Administrateur de service

Syntaxe

`epmautomate importConsolidationJournals jobName [fileName=`*`FILE_NAME`*`] [errorFileName=`*`ERROR_FILE_NAME`*`]` où

- `jobName` est le nom d'un job d'import de journal créé dans Financial Consolidation and Close.
- `fileName` (facultatif) est le nom d'un fichier JLF à partir duquel les entrées de journal doivent être importées.
- `errorFileName` (facultatif) est le nom du fichier journal dans lequel les messages générés lors du processus d'import doivent être enregistrés.

Exemple

```
epmautomate importConsolidationJournals "JIMPORT1" fileName="TestImport1.jlf"
errorFileName="TestImport1_error.log"
```

importData

Importe des données à partir d'un fichier dans l'application à l'aide des paramètres d'import de données indiqués dans un job de type `import data`.

Utilisez la commande uploadFile pour charger le fichier contenant les données d'application vers l'emplacement de chargement par défaut.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate importData` *`JOB_NAME [FILE_NAME]`* `errorFile=`*`ERROR_FILE`*`.zip` où :

- *`JOB_NAME`* représente le nom d'un job défini dans l'application.
- *`FILE_NAME`* (facultatif) identifie le nom du fichier ZIP, CSV ou TXT (fichier de données au format Essbase) à partir duquel les données doivent être importées. Si vous indiquez un nom de fichier, le nom de fichier d'import dans le job n'est pas pris en compte.
  Si le job est défini pour importer des données au format Essbase, le fichier ZIP doit contenir un fichier TXT au format Essbase. Pour d'autres jobs d'import, le fichier ZIP peut contenir des fichiers CSV qui identifient la séquence d'import dans les noms de fichier (par exemple, `data1-3.csv`, `data2-3.csv` et `data3-3.csv`).
- `errorFile` (facultatif) identifie le nom du fichier ZIP dans lequel les enregistrements rejetés sont enregistrés, le cas échéant, lors des opérations d'import. S'il existe un fichier ZIP du même nom dans la boîte d'envoi, il sera remplacé. Vous pouvez télécharger ce fichier en local à l'aide de la commande downloadFile.

Exemple

```
epmautomate importData dailydataload dailydata.zip errorFile=dataImport_error.zip
```

importDataManagement

Importe les enregistrements de gestion des données d'un fichier ZIP dans un environnement.

Cette commande importe des données dans des tables intermédiaires et de configuration à partir d'un fichier ZIP créé à l'aide de la commande exportDataManagement. Un administrateur de service utilise la commande uploadFile (par exemple, `epmAutomate uploadFile "C:/datafile/datafile.zip" inbox`) pour charger le fichier ZIP d'import dans la boîte de réception de gestion des données ou dans l'un des dossiers de celle-ci.

> **Note**
>
> Cette commande ne peut importer que les enregistrements de gestion des données exportés à partir d'un autre environnement exécutant la même mise à jour mensuelle. Par exemple, les enregistrements exportés à partir d'un environnement Oracle Fusion Cloud Enterprise Performance Management 21.11 ne peuvent être importés que dans un autre environnement 21.11.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Account Reconciliation, Tax Reporting, Profitability and Cost Management, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate importDataManagement` *`FILE_NAME`*`.zip`, où *`FILE_NAME`* est le nom du fichier ZIP qui contient les données Data Management à importer.

Exemples

- Import à partir de la boîte de réception de gestion des données :
  `epmautomate importDataManagement inbox/dataFile.zip`
- Import à partir d'un dossier de la boîte de réception :
  `epmautomate importDataManagement inbox/dm_data/dataFile.zip`

importDimension

Importe une dimension à partir d'un fichier dans une application Oracle Fusion Cloud Enterprise Data Management.

Cette commande peut importer un fichier d'entrée à partir d'une connexion définie dans Cloud EDM ou de la zone intermédiaire.

Si le fichier doit être importé à partir de la zone intermédiaire Cloud EDM, un administrateur de service utilise la commande uploadFile pour le charger vers l'environnement Cloud EDM cible. Un administrateur de service peut également utiliser la commande copyFileFromInstance pour copier le fichier à partir d'un autre environnement Oracle Fusion Cloud Enterprise Performance Management.

Champ d'application

Cloud EDM.

Rôles requis

- Administrateur de service
- Utilisateur (avec autorisation de gestionnaire de données)

Syntaxe

`epmautomate importDimension` *`APPLICATION DIMENSION IMPORT_TYPE FILE_NAME`* `[connection=`*`NAME`*`]` où :

- `APPLICATION` est le nom d'une application Cloud EDM.
- `DIMENSION` est le nom de la dimension d'application importée.
- `IMPORT_TYPE` indique la façon d'exécuter l'import. Les types d'import valides sont les suivants :
  - `ResetDimension` permet de supprimer toutes les données de dimension existantes et d'importer les nouvelles données.
  - `ReplaceNodes` permet d'ajouter ou de mettre à jour les noeuds et de remplacer les hiérarchies existantes lors de l'import.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

  - `Merge` permet de traiter les modifications incrémentielles apportées aux noeuds et aux hiérarchies à l'aide d'une demande d'import.
- `FILE_NAME` est le nom du fichier (CSV ou ZIP) contenant les données de dimension à importer. Le nom du fichier doit se terminer par le nom de la dimension préfixé par _ (trait de soulignement), par exemple : `import_Entity.csv`. Si vous effectuez l'import à partir d'un fichier ZIP contenant plusieurs fichiers d'import, cette commande dépend du nom du fichier dans le fichier ZIP pour identifier le bon fichier d'import.
  Si vous indiquez une valeur pour `connection`, vous devez importer la dimension à partir d'un fichier ZIP file, par exemple : `importdata_Entity.zip`.
- `connection=`*`NAME`* (facultatif) identifie un nom de connexion (emplacement de l'instance) défini dans Cloud EDM comme emplacement du fichier d'import. Si ce paramètre n'est pas indiqué, le processus d'import recherchera le fichier d'import dans la zone intermédiaire locale.

Exemples

- Import à partir d'un fichier chargé vers la zone intermédiaire : `epmautomate importDimension USOperations Entity ReplaceNodes data_Entity.CSV`
- Import à partir de la boîte d'envoi d'un autre environnement Cloud EPM : `epmautomate importDimension USOperations Entity ReplaceNodes data_Entity.ZIP Connection=Cloud-EPM_pln`

importEJDimensionMembers

Importe les membres de dimension de journaux Enterprise dans une application Financial Consolidation and Close.

Cette commande crée un job d'import à l'aide des paramètres de ligne de commande que vous spécifiez. Les informations relatives à ce job sont disponibles dans la console de jobs.

Champ d'application

Financial Consolidation and Close

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate importEJDimensionMembers` *`Dimension_Name File_Name`* `[ImportMode=Replace|Update] [delimiter="`*`DELIMITER`*`"] [dateFormat=`*`DATE_FORMAT`*`]`, où :

- *`DIMENSION_NAME`* est le nom de la dimension de journaux Enterprise pour laquelle les membres sont importés.
- *`FILE_NAME`* est le nom du fichier CSV source contenant les données de membre de la dimension indiquée par le paramètre *`DIMENSION_NAME`*. Utilisez la commande uploadFile pour charger ce fichier vers l'emplacement de chargement par défaut.
- `ImportMode` (facultatif) est le mode à utiliser pour l'opération d'import. Indiquez les éléments suivants :
  - `Update` pour modifier uniquement les membres inclus dans le fichier CSV source. Les autres membres ne sont pas modifiés.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

  - `Replace` pour effacer les membres de dimension du formulaire et les remplacer par les membres du fichier CSV source. Dans ce mode, les valeurs non disponibles dans le fichier source seront laissées vides après l'import. Il s'agit du mode par défaut.
- `delimiter` (facultatif) identifie le séparateur utilisé pour séparer les membres de dimension dans le fichier CSV source. La valeur par défaut est la virgule (,). Pour utiliser la tabulation comme séparateur, spécifiez *t*.
- `dateFormat`, (facultatif) indique le format de date à utiliser pour convertir les valeurs de date ou de date/heure, le cas échéant, dans le fichier CSV source. Formats de date pris en charge : `MM/dd/yyyy`, `dd/MM/yyyy`, `d-M-yyyy`, `dd-MMM-yy`, `MM-dd-yyyy` et `MMM d, yyyy`, qui est le format de date par défaut.

Exemple

```
epmautomate importEJDimensionMembers Regions RegionMembers.csv ImportMode=Update
delimiter=\t dateFormat="dd/MM/yyyy"
```

importJobConsole

Clone les enregistrements de la console de jobs à l'aide d'un fichier ZIP contenant les enregistrements de la console de jobs exportés à partir d'un environnement.

L'import d'enregistrements de la console de jobs à l'aide de cette commande est une tâche ponctuelle à effectuer après avoir exécuté la commande recreate. Si vous avez déjà utilisé cette commande pour importer des enregistrements de la console de jobs, toute tentative ultérieure d'utilisation de la commande échouera tant que vous n'aurez pas recréé l'environnement.

Utilisez la commande exportJobConsole (`epmAutomate exportJobConsole FILE_NAME.zip nDays=All jobTypes=All jobStatusCode=All`) pour créer le fichier ZIP utilisé en tant qu'entrée pour cette commande.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Strategic Workforce Planning et Sales Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Jobs - Gérer

Syntaxe

`epmautomate importJobConsole` *`FILE_NAME`*`.zip [logFileName=jobConsoleLog]`, où :

- *`FILE_NAME`* est le nom du fichier ZIP qui contient les enregistrements de la console de jobs à importer. Vous utilisez la commande uploadFile pour charger ce fichier vers l'environnement.
- `logFileName` (facultatif) identifie `jobConsoleLog`, où seront enregistrées les erreurs rencontrées au cours de l'import. Si cette valeur n'est pas indiquée, la commande génère un fichier d'erreurs nommé selon la convention suivante : *`username`*`importLog_date_timestamp.zip`. Vous pouvez télécharger ce fichier à l'aide de la commande downloadFile.

Exemple

```
epmautomate importJobConsole jobConsole.zip jobConsoleLog
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

importLibraryArtifact

Importe les artefacts de bibliothèque d'une archive ou d'un fichier dans la bibliothèque Narrative Reporting.

Avant d'exécuter cette commande, un administrateur de service utilise la commande uploadFile pour charger le fichier ou l'archive source vers l'environnement.

Champ d'application

Narrative Reporting

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Un niveau de sécurité supplémentaire doit être accordé aux utilisateurs dotés du rôle Super utilisateur, Utilisateur ou Visualiseur via une liste de contrôle d'accès.

Syntaxe

`epmautomate importLibraryArtifact` *`SOURCE_FILE`* `[errorFile=`*`ERROR_FILE`*`.txt] [importFormat=Native|File] [importFolder=`*`FOLDER_PATH`*`] [ importPermission=true|false] [overwrite=true|false]`, où :

- *`SOURCE_FILE`* est le nom de l'archive contenant les artefacts à importer dans la bibliothèque. Ce fichier doit être disponible dans la boîte de réception.
- *`errorFile`* (facultatif) est le nom unique du fichier texte qui stockera les erreurs liées à l'import.
- `importFormat` (facultatif) est l'un des formats suivants :
  - `Native` importe des artefacts à partir d'un fichier ZIP créé à l'aide de la commande exportLibraryArtifact avec l'option `exportFormat=Native`. Il s'agit de la valeur par défaut.
  - `File` importe un fichier binaire.

> **Remarque**
>
> Vous utilisez la commande importSnapshot pour importer des fichiers ZIP d'artefacts de bibliothèque (créés à l'aide de la commande exportLibraryArtifact avec l'option `exportFormat=LCM`) dans des environnements Financial Consolidation and Close, Planning, de modules Planning ou Tax Reporting.

- `importFolder` (facultatif) est l'emplacement de la bibliothèque où seront stockés les artefacts importés. Indiquez ce chemin si cet emplacement est différent de `Library` (emplacement d'import par défaut).
- *`importPermission`* indique si les autorisations d'accès définies pour l'artefact doivent être importées. La valeur par défaut est `False`.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- *overwrite* indique si les artefacts portant le même nom dans l'emplacement de bibliothèque indiqué doivent être remplacés. La valeur par défaut est `False`, ce qui signifie que le processus n'importera pas d'artefact si un artefact portant le même nom existe dans l'emplacement d'import.

Une fois l'import effectué, utilisez la commande downloadFile pour télécharger les fichiers d'erreurs sur un ordinateur local.

Exemples

- Import d'un fichier au format binaire :
```
epmautomate importLibraryArtifact newReports.doc
errorFile=report_imp_errors.txt importFormat=File importFolder="Library/My
Reports" importPermission=true overwrite=true
```
- Import d'artefacts à partir d'un fichier ZIP exporté :
```
epmautomate importLibraryArtifact newReports.zip
errorFile=report_imp_errors.txt importFormat=Native importFolder="Library/My
Reports" importPermission=true overwrite=true
```
- Import de rapports dans un environnement Financial Consolidation and Close, Planning, de modules Planning ou Tax Reporting à partir d'un fichier ZIP exporté :
```
epmautomate importSnapshot newReports.zip
```

importLibraryDocument

Importe un document ou de nombreux documents contenus dans un fichier ZIP vers un emplacement spécifique de la bibliothèque Planning, Financial Consolidation and Close, Enterprise Profitability and Cost Management ou Tax Reporting.

Les documents que vous pouvez importer à l'aide de cette commande incluent les documents Microsoft Word (.DOC, .DOCX), les documents Excel (.XLS, .XLSX), les présentations PowerPoint (.PPT, .PPTX), les fichiers de valeurs séparées par des virgules (.CSV), les fichiers PDF (jusqu'à 100 Mo), les fichiers texte (.TXT) et les archives de fichiers (.ZIP, jusqu'à 100 Mo). Avant d'exécuter cette commande, utilisez l'écran de migration d'import ou une commande EPM Automate telle que uploadFile, copyFileFromInstance ou copyFromObjectStorage pour charger le fichier ou l'archive source vers l'environnement.

Champ d'application

Planning, FreeForm, Financial Consolidation and Close, Enterprise Profitability and Cost Management, Tax Reporting

Rôles requis

Administrateur de service

Syntaxe

`epmautomate importLibraryDocument` *`SOURCE_FILE DESTINATION`* `[jobName=JOB_NAME] [description=DESCRIPTION][overwrite=true|false][deleteAfterImport=true|false] [errorFile=`*`FILE_NAME`*`.log]`, où :

- *SOURCE_FILE* est le nom d'un document ou d'une archive qui contient un ou plusieurs documents à importer dans la bibliothèque.
- *DESTINATION* est le chemin qualifié complet ou l'identifiant unique universel (UUID) du dossier de destination dans lequel le fichier ou le contenu de l'archive doit être importé.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `jobName` (facultatif) est le nom du job d'import de document de bibliothèque à utiliser pour importer le document. Si aucune valeur n'est indiquée, le job `Importer le document vers la bibliothèque` est utilisé par défaut.
- `description` (facultatif) est une description du document en cours d'import.
- `overwrite` (facultatif) indique si les documents existants du dossier de destination avec des noms identiques aux documents source doivent être remplacés. La valeur par défaut est `false`.
- `deleteAfterImport` (facultatif) indique si le fichier source doit être supprimé une fois le processus d'import terminé. La valeur par défaut est `false`.
- `errorFile` (facultatif) est le nom d'un fichier qui capture les informations rapportées lors du processus d'exécution de la commande. Ce fichier sera créé dans la boîte d'envoi avec l'extension `.log`.
  Utilisez la commande downloadFile pour télécharger ce fichier sur l'ordinateur local.

Exemple

```
epmautomate importLibraryDocument "FinancialReport2024.xlsx" "Library/Insights"
jobName=ImportSingleDocument description="Annual Financial Report 2024"
overwrite=true deleteAfterImport=true errorFile=FinancialReport2024ImportLog.log
```

importLogoImage

Importe le logo d'entreprise utilisé dans un environnement Account Reconciliation dans un autre environnement à partir d'un fichier d'export.

Champ d'application

Account Reconciliation

Rôles requis

Administrateur de service

Syntaxe

`epmautomate importLogoImage` *`IMAGE_NAME`*`.jpg`, où *`IMAGE_NAME`* est le nom du fichier de l'image de logo.
Vous pouvez télécharger l'image exportée à l'aide de la commande downloadFile. Chargez-la vers l'environnement cible à l'aide de la commande uploadFile, puis exécutez la commande importLogoImage pour l'importer.

Exemple

```
epmautomate importLogoImage corpLogo.jpg
```

importMapping

Importe des mappings à partir d'un fichier d'import de mapping, précédemment chargé vers l'environnement.

Un administrateur de service utilise la commande uploadFile pour charger des fichiers dans la boîte de réception de la gestion des données ou un dossier qui s'y trouve.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate importMapping` *`DIMENSION_NAME|ALL FILE_NAME IMPORT_MODE VALIDATION_MODE LOCATION`*, où :

- *`DIMENSION_NAME|ALL`* indique le destinataire du mapping. Indiquez le nom de la dimension dans laquelle les mappings doivent être importés ou `ALL` pour importer tous les mappings inclus dans le fichier vers les dimensions appropriées.
- *`FILE_NAME`* est le nom et l'emplacement du fichier d'import de mapping disponible dans la boîte de réception de la gestion des données ou un répertoire qui s'y trouve. Indiquez le nom de fichier (fichiers TXT au format de gestion des données standard) et son chemin (par exemple, `inbox/AccountMap.txt` ou `inbox/pbcs_maps/AccountMap.txt`).
- *`IMPORT_MODE`* a la valeur `REPLACE` pour effacer les règles de mapping existantes avant l'import de mappings ou `MERGE` pour ajouter de nouvelles règles de mapping à des règles existantes.
- *`VALIDATION_MODE`* a la valeur `TRUE` pour valider des membres cible par rapport à l'application ou `FALSE` pour charger le fichier de mapping sans exécuter de validations.
- *`LOCATION`* est l'emplacement de la gestion des données pour lequel des règles de mapping doivent être chargées.

Exemples

- `epmautomate importMapping Account inbox/AccountMap.txt MERGE FALSE "France Sales"`
- `epmautomate importMapping ALL "inbox/France Sales/AllMaps.txt" MERGE FALSE "France Sales"` (charge les mappings à partir du fichier d'import de mapping dans toutes les dimensions mappées à l'emplacement France Sales)

importMetadata

Importe des métadonnées dans l'application à l'aide des paramètres d'import indiqués dans un job de type `import metadata`. Vous pouvez éventuellement indiquer le nom du fichier ZIP à partir duquel les métadonnées doivent être importées.

Utilisez la commande uploadFile pour charger le fichier contenant les métadonnées vers l'emplacement de chargement par défaut.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate importMetadata` *`JOB_NAME [FILE_NAME]`* `errorFile=`*`ERROR_FILE`*`.zip` où :

- *`JOB_NAME`* représente le nom d'un job défini dans l'application.
- *`FILE_NAME`* (facultatif) identifie le nom du fichier ZIP à partir duquel les métadonnées doivent être importées. Si cette option est définie, le contenu du fichier ZIP indiqué est prioritaire sur les noms de fichier définis dans le job. Le fichier ZIP peut contenir des fichiers CSV. Les noms de fichier contenant des métadonnées pour les dimensions doivent correspondre aux noms de fichier d'import définis dans le job ou se terminant par *`_DIMENSIONNAME`*`.csv`. Exemples : `metadata_Entity.csv`, `metadata_HSP_Smart Lists.csv` et `metadata_Exchange Rates.csv`.
- `errorFile` (facultatif) identifie le nom du fichier ZIP dans lequel les enregistrements rejetés sont enregistrés, le cas échéant, lors des opérations d'import. S'il existe un fichier ZIP du même nom dans la boîte d'envoi, il sera remplacé. Vous pouvez télécharger ce fichier en local à l'aide de la commande downloadFile.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Remarque

- Vous ne pouvez pas renommer des membres en exécutant un job d'import de métadonnées avec un fichier de chargement dans lequel la propriété `old_name` ou `unique_name` a été modifiée. Le changement de nom des membres n'est alors pas pris en compte.
- Vous ne pouvez pas supprimer des dimensions d'attribut lors de l'import de métadonnées à l'aide de cette commande.
- Seules les métadonnées des dimensions pour lesquelles l'import des métadonnées est configuré dans le job sont importées. Les métadonnées des autres dimensions, si elles sont contenues dans le fichier ZIP, ne sont pas prises en compte.

  Une situation d'import ambiguë est créée si les deux conditions ci-dessous sont remplies pour le fichier ZIP :

  – Le fichier ZIP contient un fichier de métadonnées dont le nom correspond à celui du fichier défini dans le job.

  – Le fichier ZIP contient des fichiers de métadonnées ou des fichiers dont les noms se terminent par *_DIMENSIONNAME*`.CSV` ou *_DIMENSIONNAME*`.TXT`, où *DIMENSIONNAME* est le nom de la dimension dans laquelle les métadonnées sont importées.

Oracle recommande que le fichier ZIP contienne un fichier de métadonnées dont le nom est identique à celui qui est référencé dans le job ou un fichier dont le nom se termine par *_DIMENSIONNAME*`.CSV` (ou *_DIMENSIONNAME*`.TXT`), mais pas les deux. Par exemple, si vous êtes en train de charger un job qui référence le fichier de métadonnées `Employés_A-Z.CSV` dans la dimension `Employés`, le fichier ZIP peut contenir le fichier `Employés_A-Z.CSV` ou `Nouveaux_Employés.CSV`, mais pas les deux. Si le ZIP contient `Employés_A-Z.CSV` et `Nouveaux_Employés.CSV`, EPM Automate peut sélectionner l'un ou l'autre des fichiers pour l'import en fonction de l'ordre des fichiers dans le fichier ZIP. Le fichier `Employés_A-Z.CSV` est une correspondance possible pour l'import car son nom correspond à celui du fichier référencé dans le job. Le fichier `Nouveaux_Employés.CSV` est également une correspondance possible car son nom correspond au modèle *_DIMENSIONNAME*`.CSV`.

Exemple

```
epmautomate importMetadata importAccount importAccount.zip
errorFile=metadataImport_error.zip
```

importOwnershipData

Importe les données de participation d'un fichier CSV disponible dans l'environnement vers une période.

Avant d'exécuter cette commande, un administrateur de service utilise la commande uploadFile pour charger le fichier CSV source d'import dans l'environnement. L'en-tête du fichier CSV se présente comme suit :

```
Scenario, Year, Period, Entity, Parent, POwn, Control, Method
```

Les valeurs `POwn`, `Control` et `Method` sont facultatives.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Les données de participation importées sont fusionnées avec les données existantes, ce qui peut créer des entrées de participation non valides. Si une entité est présente dans plusieurs branches d'une hiérarchie, le pourcentage de participation combinée de l'entité risque de dépasser 100 % du fait des données de participation importées. Vous devez corriger le pourcentage de participation manuellement pour garantir qu'il ne dépasse pas 100 %.

Champ d'application

Financial Consolidation and Close et Tax Reporting.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur disposant d'un accès en écriture à l'entité.

Syntaxe

`epmautomate importOwnershipData` *`Scenario Year Period FILE_NAME`*, où :

- `Scenario` est le scénario vers lequel les données de participation doivent être importées.
- `Year` est l'année vers laquelle les données doivent être importées.
- `Period` est la période de l'année vers laquelle les données de participation doivent être importées.
- `FILE_NAME` est le nom du fichier CSV à partir duquel les données doivent être importées.

Exemple

```
epmautomate importOwnershipData FCCS_TotalActual FY19 Jan importfile.csv
```

importPreMappedBalances

Importe des données de solde pré-mappées à partir d'un fichier dans le référentiel Account Reconciliation.

Champ d'application

Account Reconciliation

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmautomate importPreMappedBalances` *`PERIOD FILE_NAME BALANCE_TYPE CURRENCY_BUCKET`*, où :

- *`PERIOD`* est le nom d'une période.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- *FILE_NAME* est le nom du fichier CSV contenant les données à importer.
- *BALANCE_TYPE* est SRC ou SUB.
- *CURRENCY_BUCKET* est Entered, Functional ou Reporting.

Exemple

```
epmautomate importPreMappedBalances "January 2015" dailydata.csv SRC Reporting
```

importPreMappedTransactions

Importe des transactions pré-mappées à partir d'un fichier CSV dans le référentiel Account Reconciliation.

Champ d'application

Account Reconciliation

Rôles requis

Administrateur de service, Super utilisateur, Utilisateur, Visualiseur
Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

epmautomate importPreMappedTransactions *PERIOD TRANSACTION_TYPE FILE_NAME DATE_FORMAT*, où :

- *PERIOD* est le nom d'une période.
- *TRANSACTION_TYPE* est l'une des valeurs suivantes :
  - BEX pour charger les explications de solde
  - SRC pour charger les ajustements de système source
  - SUB pour charger les ajustements de sous-système
  - VEX pour charger les explications d'analyse de la variance
- *FILE_NAME* est le nom du fichier CSV à partir duquel les données doivent être importées.
- *DATE_FORMAT* est la chaîne de texte indiquant le format de date (par exemple, MMM d, yyyy).

Exemple

```
epmautomate importPreMappedTransactions "January 2015" "BEX" transactions.csv
"MMM d, yyyy"
```

importProfiles

Importe de nouvelles définitions de profil à partir d'un fichier CSV dans le référentiel Account Reconciliation.

Champ d'application

Account Reconciliation

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmautomate importProfiles` *`FILE_NAME PROFILE_TYPE METHOD DATE_FORMAT`*, où :

- *`FILE_NAME`* est le nom du fichier CSV à partir duquel les données doivent être importées.
- *`PROFILE_TYPE`* est `profiles` ou `children`.
- *`METHOD`* est `Replace` ou `Update`.
- *`DATE_FORMAT`* est une chaîne de texte indiquant le format de date (par exemple, MMM d, yyyy).

Exemple

```
epmautomate importProfiles NewRecProfiles.csv Profiles Replace "MMM d, yyyy"
```

importRates

Importe des taux de change à partir d'un fichier CSV dans le référentiel Account Reconciliation.

Champ d'application

Account Reconciliation

Rôles requis

Administrateur de service, Super utilisateur, Utilisateur, Visualiseur
Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmautomate importRates` *`PERIOD RATE_TYPE REPLACE_MODE FILE_NAME`*, où :

- *`PERIOD`* est le nom d'une période.
- *`RATE_TYPE`* est un type de taux prédéfini.
- *`REPLACEMENT_MODE`* est `Replace` ou `ReplaceAll`.
- *`FILE_NAME`* est le nom du fichier CSV à partir duquel les taux doivent être importés.

Exemple

```
epmautomate importRates "January 2015" Actual ReplaceAll avgrates.csv
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

importRCAttributeValues

Importe des valeurs d'attribut dans les attributs de liste ou de groupe du module Conformité de rapprochement d'Account Reconciliation.

Champ d'application

Account Reconciliation

Rôles requis

- Administrateur de service
- Super utilisateur (nécessite un niveau de sécurité supplémentaire fourni via des listes de contrôle d'accès)

Syntaxe

`epmautomate importRCAttributeValues` *`ATTRIBUTE_NAME FILE_NAME`* `[METHOD=REPLACE|REPLACE ALL|UPDATE][DATEFORMAT=DD/MM/YYYY|DD-MMM-YYYY|MMM d,yyyy|All]`, où :

- *`ATTRIBUTE_NAME`* est le nom de l'attribut de liste ou de groupe dans lequel importer les valeurs.
- *`FILE_NAME`* est le fichier d'import CSV à partir duquel les valeurs doivent être importées. Utilisez la commande uploadFile pour charger ce fichier dans l'environnement avant d'exécuter cette commande.
- `METHOD` (facultatif) indique la façon dont les valeurs doivent être importées. Valeurs valides :
  - `Replace` pour ajouter toutes les valeurs du fichier d'import en tant que valeurs d'attribut dans le module Conformité de rapprochement. Les valeurs d'attribut existantes sont remplacées par celles du fichier d'import ; les valeurs n'existant pas dans l'attribut mais présentes dans le fichier d'import sont ajoutées. Les valeurs existant dans l'attribut qui ne figurent pas dans le fichier d'import ne sont pas modifiées. Toutes les données d'attribut d'une valeur de clé particulière sont remplacées par le contenu du fichier ou effacées. Les nouvelles valeurs sont ajoutées en bas, dans leur ordre d'apparition dans le fichier.
    Ce type d'import s'avère particulièrement utile si vous déplacez seulement les dernières modifications à partir d'un système source, par exemple, lors de l'ajout de nouvelles données de magasin résultant d'une acquisition afin de ne remplacer que les valeurs d'attribut indiquées, si elles sont présentes, par les valeurs du fichier d'import. Il s'agit du comportement par défaut.
  - `Replace All` pour remplacer les valeurs d'attribut existantes par les valeurs de l'import. Les valeurs existant dans l'attribut mais ne figurant pas dans le fichier d'import sont supprimées.
    Ce type d'import s'avère particulièrement utile si vous mettez en miroir des valeurs à partir d'un système source avec une mise à jour complète, par exemple, pour les mises à jour hebdomadaires de synchronisation avec les données de magasin du système source.
  - `Update` pour remplacer ou ajouter toutes les valeurs du fichier d'import dans l'attribut. Les valeurs d'attribut existantes sont remplacées par celles du fichier d'import. Les valeurs se trouvant dans le fichier d'import mais ne figurant pas dans l'attribut sont ajoutées. Les valeurs existant dans l'attribut mais ne figurant pas dans le fichier d'import ne sont pas modifiées. Seules les données d'attribut d'une valeur de clé particulière sont remplacées par le contenu du fichier ; les données des attributs non

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

disponibles dans le fichier demeurent intactes. Toute clé présente dans le fichier d'import mais pas dans l'attribut génère une erreur.
Ce type d'import s'avère particulièrement utile si vous mettez à jour quelques attributs parmi la totalité des valeurs d'attribut. Par exemple, lorsque vous mettez à jour des directeurs de magasin après une réorganisation, sans que cela ait d'effet sur le reste des données relatives aux magasins.

- `Dateformat` (facultatif) indique les formats de date valides (par exemple, `DD/MM/YYYY`, `DD-MMM-YYYY` (valeur par défaut), `MMM d,yyyy` et `All`) à analyser. Vous pouvez indiquer plusieurs valeurs de format de date en les séparant par un point-virgule.

Exemple

```
epmautomate importRCAttributeValues Stores StoreData.csv METHOD=Replace
DATEFORMAT="All"
```

importReconciliationAttributes

Importe des attributs de rapprochement dans des rapprochements existants à partir d'un fichier qu'un administrateur de service a chargé dans l'environnement Account Reconciliation à l'aide de la commande uploadFile.

Champ d'application

Account Reconciliation

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmautomate importReconciliationAttributes` *`FILE`*`.CSV Period [Rules=`*`RULE_NAME`*`] [Reopen=true|false] [Dateformat=`*`DATE_FORMAT`*`]` où :

- `FILE` est le nom du fichier CSV contenant les attributs de rapprochement que vous voulez importer dans des rapprochements.
- `Period` identifie la période à laquelle appartient le rapprochement.
- `Rules` (facultatif) identifie les règles qui doivent être exécutées sur les rapprochements concernés après l'import des attributs. Séparez les noms de règle par une virgule. Les valeurs valides sont les suivantes :
  - `None` : aucune règle n'est exécutée sur les rapprochements concernés. Il s'agit de la valeur par défaut, à ne pas combiner avec d'autres valeurs.
  - `ALL` : exécute toutes les règles définies pour les rapprochements pour la période indiquée. Cette valeur doit être utilisée seule et ne peut pas être combiné avec d'autres noms de règle.
  - `SET_ATTR_VAL` : exécute la règle prédéfinie pour définir la valeur d'attribut.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

  - `CRT_ALT` : exécute la règle prédéfinie pour créer une alerte.
  - `AUTO_APP` : exécute la règle prédéfinie pour approuver automatiquement le rapprochement.
  - `AUTO_SUB` : exécute la règle prédéfinie pour soumettre automatiquement le rapprochement.
  - `EMAIL_ON_SAVE` : exécute la règle prédéfinie pour envoyer automatiquement un courriel après la mise à jour du rapprochement.
- `Reopen` (facultatif) indique si les rapprochements modifiés doivent être rouverts lorsque l'opération d'import a été exécutée. La valeur par défaut est `false`.
- `Dateformat` (facultatif) indique les formats de date valides (par exemple, `MM-dd-yyyy`, `dd-MMMM-yy`, `MMM d` et `yyyy`) à analyser. Vous pouvez indiquer plusieurs valeurs de format de date en les séparant par un point-virgule.

Exemples

- **Import de valeurs d'attribut pour une période et exécution de plusieurs règles avec plusieurs formats de date :**
  ```
  epmAutomate importReconciliationAttributes Reconciliations.csv "July 2020" Rules=SET_ATTR_VAL,CRT_ALT,AUTO_APP,AUTO_SUB" Reopen=true "Dateformat=MM-dd-yyyy;dd-MMM-yy;MMM d, yyyy"
  ```
- **Import de valeurs d'attribut pour une période sans exécution de règles :**
  ```
  epmAutomate importReconciliationAttributes Reconciliations.csv "July 2020"
  ```
- **Import de valeurs d'attribut pour une période, exécution de toutes le règles applicables et réouverture des rapprochements concernés :**
  ```
  epmAutomate importReconciliationAttributes Reconciliations.csv "July 2020" Rules=ALL Reopen=true
  ```

importSDMDimensionMembers

Importe les membres de dimension Supplemental Data Manager dans une application Financial Consolidation and Close.

Cette commande crée un job d'import à l'aide des paramètres de ligne de commande que vous spécifiez. Les informations relatives à ce job sont disponibles dans la console de jobs.

Champ d'application

Financial Consolidation and Close, Tax Reporting

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate importSDMDimensionMembers` *`DIMENSION_NAME FILE_NAME`* `[ImportMode=Replace|Update] [delimiter=`*`DELIMITER`*`] [dateFormat=`*`DATE_FORMAT`*`]`, où :

- *`DIMENSION_NAME`* est le nom de la dimension Supplemental Data Manager pour laquelle les membres sont en cours d'import.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- *FILE_NAME* est le nom du fichier CSV source contenant les données de membre de la dimension indiquée par le paramètre *DIMENSION_NAME*. Utilisez la commande uploadFile pour charger ce fichier vers l'emplacement de chargement par défaut.
- `ImportMode` (facultatif) est le mode à utiliser pour l'opération. Indiquez les éléments suivants :
  - `Update` pour modifier uniquement les membres inclus dans le fichier CSV source. Les autres membres ne sont pas modifiés.
  - `Replace` pour effacer les membres de dimension du formulaire et les remplacer par les membres du fichier CSV source. Dans ce mode, les valeurs non disponibles dans le fichier source seront laissées vides après l'import. Il s'agit du mode par défaut.
- `delimiter` (facultatif) identifie le séparateur utilisé pour séparer les membres de dimension dans le fichier CSV source. La valeur par défaut est la virgule (,). Pour utiliser la tabulation comme séparateur, spécifiez \t.
- `dateFormat`, (facultatif) indique le format de date à utiliser pour convertir les valeurs de date ou de date/heure, le cas échéant, dans le fichier CSV source. Formats de date pris en charge : `MM/dd/yyyy`, `dd/MM/yyyy`, `d-M-yyyy`, `dd-MMM-yy`, `MM-dd-yyyy` et `MMM d, yyyy`, qui est le format de date par défaut.

Exemple

```
epmautomate importSDMDimensionMembers Regions RegionMembers.csv delimiter=\t
dateFormat=dd/MM/yyyy
```

importSnapshot

Importe le contenu d'un instantané dans l'environnement de service. L'instantané que vous importez doit être disponible à l'emplacement de chargement par défaut.

Un administrateur de service utilise la commande uploadFile pour charger un instantané ou la commande copySnapshotFromInstance pour le copier à partir d'une autre instance.

- Les données suivantes ne font pas partie des instantanés d'application Planning, modules Planning et FreeForm :
  - Données d'audit
  - Données de console de job

  Utilisez la commande cloneEnvironment ou la fonctionnalité de clonage d'environnement si vous souhaitez copier les données d'audit et de console de job dans l'environnement cible. L'import d'instantané peut échouer si le processus métier Planning contient un membre de période prédéfini renommé qui a été remplacé par un membre de période libre. Par exemple, vous avez renommé le membre de période *YearTotal* prédéfini en *unused_YearTotal*, puis vous avez ajouté un membre de période d'un autre type avec le nom du membre prédéfini d'origine (*YearTotal* dans cet exemple). Dans ce scénario, l'import de l'instantané peut échouer.
- Les instantanés ne contiennent pas les données de table intermédiaire de gestion des données. Pour importer ces données, utilisez les commandes exportDataManagement et importDataManagement ou l'interface de scripts de maintenance système de gestion des données. Vous pouvez utiliser la commande cloneEnvironment ou la fonctionnalité de clonage d'environnement pour créer une copie identique de l'environnement, y compris des données de table intermédiaire de gestion des données.

Les activités que vous pouvez exécuter à l'aide de cette commande dépendent de votre rôle.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Les administrateurs de service peuvent importer uniquement des artefacts d'application dans un environnement.
- Vous devez disposer des rôles Administrateur de service et Administrateur de domaine d'identité pour importer du contenu d'application dans l'environnement de service, ainsi que des artefacts de domaine d'identité (utilisateurs et leurs affectations de rôle prédéfini) dans le domaine d'identité de l'environnement.
  Si un utilisateur qui ne se trouve pas dans le domaine d'identité est référencé dans l'instantané en cours d'import, la commande crée un utilisateur dans le domaine d'identité et lui affecte le mot de passe par défaut que vous indiquez dans la commande ou affecte un mot de passe unique temporaire à chaque utilisateur si vous n'indiquez pas de mot de passe dans la commande. Par défaut, l'utilisateur devra réinitialiser le mot de passe lors de la première connexion.

> **Remarque**
>
> - Pour les processus métier autres qu'Account Reconciliation, que Profitability and Cost Management et qu'Oracle Fusion Cloud Enterprise Data Management : lors du chargement des métadonnées, Oracle Fusion Cloud Enterprise Performance Management peut effectuer plusieurs transmissions de chargement lorsque la tentative précédente entraîne le rejet des enregistrements car des membres partagés précèdent des membres de base dans l'outline. De telles tentatives peuvent augmenter le temps de traitement de la commande.
> - Un rôle prédéfini doit être affecté aux utilisateurs qui sont membres de groupes dans Contrôle d'accès. Les tentatives d'affectation d'un utilisateur ne disposant pas d'un rôle prédéfini à un groupe ne sont pas autorisées.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Cloud EDM, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Le rôle Administrateur de domaine d'identité est requis pour importer des utilisateurs et des affectations de rôle prédéfini à l'aide de l'option *`importUsers=true`*.

Syntaxe

`epmautomate importSnapshot` *`SNAPSHOT_NAME`* `[importUsers=true|false] [userPassword=`*`DEFAULT_PASSWORD`*`] [resetPassword=true|false]` où :

- *`SNAPSHOT_NAME`* est le nom d'un instantané à l'emplacement de chargement par défaut.
- `importUsers` (facultatif) indique si des utilisateurs et leurs affectations de rôle prédéfini doivent être importés à partir de l'instantané. La valeur par défaut est `false`. Utilisez `importUsers=true` pour importer des utilisateurs et des affectations de rôle prédéfini dans le domaine d'identité si l'instantané source contient des données à propos des nouveaux utilisateurs ou si de nouveaux rôles ont été affectés aux utilisateurs actuels.
  Les valeurs de connexion utilisateur ne respectent pas la casse. Par exemple, la valeur de connexion utilisateur `jane.doe@example.com` est considérée comme identique à `Jane.Doe@Example.com`, ou à toute autre variation de la casse. Si une quelconque variation

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

correspond à une valeur de connexion utilisateur existant dans le domaine d'identité, cette commande n'importe pas l'utilisateur à partir de l'instantané.

> **Remarque**
>
> – L'import des utilisateurs et de leur rôle prédéfini échoue si un utilisateur qui n'est pas administrateur de domaine d'identité exécute l'opération d'import. L'erreur suivante est enregistrée dans le rapport sur le statut de la migration : `Echec de l'import de l'artefact d'annuaire externe ARTIFACT_NAME. L'utilisateur USER_NAME n'est pas autorisé à effectuer cette opération. L'utilisateur doit être doté du rôle Administrateur de domaine d'identité pour effectuer cette opération.`
>
> – Si vous n'importez pas les utilisateurs et qu'un utilisateur dans l'instantané source n'est pas affecté à un rôle prédéfini dans l'environnement cible, une erreur (`EPMIE-00070 : Utilisateur introuvable lors de l'import des rôles affectés.`) s'affiche.

– Les modifications apportées aux rôles prédéfinis de l'utilisateur seront mises à jour en fonction des rôles affectés dans l'instantané source. Toutefois, les affectations de rôle dans la cible ne seront pas enlevées pour correspondre à celles de l'instantané source. Par exemple, supposons que `jdoe` se voit affecter le rôle prédéfini de super utilisateur dans l'environnement cible, mais ne dispose que du rôle d'utilisateur dans l'instantané source. Dans ce cas, cette commande affecte à `jdoe` le rôle d'utilisateur et n'enlève pas l'affectation de rôle de super utilisateur dans l'environnement cible.

– Cette commande ne supprime pas les utilisateurs existants de l'environnement cible s'ils n'existent pas dans l'instantané source. Par exemple, `jdoe` dispose d'un compte dans l'environnement cible, mais ce compte n'est pas présent dans l'instantané source. Dans ce cas, le compte de `jdoe` dans l'environnement cible n'est pas supprimé.

– Cette commande ajoute des utilisateurs qui n'existent pas dans l'environnement cible. Elle ne met pas à jour les propriétés de l'utilisateur actuel dans l'environnement cible, même si celles-ci sont différentes dans l'instantané source. Par exemple, si l'orthographe du nom de famille de `jdoe` est différente dans l'instantané source et dans l'environnement cible, aucune modification ne sera apportée dans l'environnement cible.

– Cette commande ne modifie pas les mots de passe des utilisateurs existants dans l'environnement cible, même s'ils sont différents dans l'instantané source.

- `userPassword` (facultatif) indique le mot de passe par défaut à affecter aux nouveaux utilisateurs qui sont créés dans le domaine d'identité. Le mot de passe que vous indiquez doit respecter les exigences minimales en matière de mot de passe. Si vous n'indiquez aucune valeur pour ce paramètre, un mot de passe temporaire unique est affecté à chaque utilisateur.
- *`resetPassword`* (facultatif) indique si le nouvel utilisateur doit modifier le mot de passe lors de la première connexion. La valeur par défaut est `true`, ce qui oblige les nouveaux utilisateurs à modifier leur mot de passe lors de leur première connexion. Si cette valeur est définie sur `true`, les nouveaux utilisateurs recevront des courriels d'activation de compte les invitant à modifier leur mot de passe.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exemples

- Import d'artefacts d'application uniquement : `epmautomate importSnapshot April16FullApp`
- Import d'artefacts de domaine d'identité et d'application (nécessite les rôles Administrateur de service et Administrateur de domaine d'identité) :
  - Affectation d'un mot de passe temporaire unique à chaque nouvel utilisateur avec obligation de le réinitialiser lors de la première connexion : `epmautomate importSnapshot April16FullApp importUsers=true`
  - Affectation d'un mot de passe spécifique, les utilisateurs ayant le choix de le modifier ou non (non recommandé pour les imports dans des environnements de production) : `epmautomate importSnapshot April16FullApp importUsers=true userPassword=P@ssw0rd1 resetPassword=false`

importSupplementalCollectionData

Importe des données de collecte supplémentaires d'un fichier dans l'application.

Utilisez la commande uploadFile pour charger le fichier contenant les données vers l'emplacement de chargement par défaut. Le format du fichier d'import est le suivant :

```
#Workflow
Workflow_Dimension_1_Name,Workflow_Dimension_2_Name,Workflow_Dimension_n_Name
Workflow_Dimension_1_Member,Workflow_Dimension_2_Member,Workflow_Dimension_n_M
ember
#Collection
Collection_Attribute_1,Collection_Attribute_2,Collection_Attribute_n
Record1_Attr_Value_1,Record1_Attr_Value_2, Record1_Attr_Value_n
```

Par exemple :

```
#Workflow
Entity
9100
#Collection
Custody Account Code,Trade Currency Code,Account Description,Base Currency
Code,CIC Code,IFRS 13 Tier,SII Portfolio Type,WPM Detailed NAV ID,WPM Asset
Description
1,,,,111,,,,6
```

Champ d'application

Financial Consolidation and Close, et Tax Reporting.

Rôles requis

Administrateur de service

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Syntaxe

> **Remarque**
>
> Tous les paramètres de commande doivent être placés entre guillemets doubles.

`epmautomate importSupplementalCollectionData "FILE_NAME""COLLECTION_NAME" "YEAR" "PERIOD" "[FREQUENCY_DIMENSION=MEMBER]"` où :

- `FILE_NAME` est le nom d'un fichier CSV, disponible à l'emplacement de chargement par défaut, qui contient les données supplémentaires au format correct.
- `COLLECTION_NAME` est le nom de la collecte dans laquelle les données supplémentaires figurant dans le fichier doivent être importées.
- `YEAR` est le membre de dimension d'année à utiliser pour la collecte.
- `PERIOD` est le nom de la dimension de période à utiliser pour la collecte.
- *`FREQUENCY_DIMENSION`* (facultatif) est le nom de la dimension de fréquence à utiliser pour la collecte. Vous pouvez indiquer autant de dimensions de fréquence que nécessaire au format *`"FREQUENCY_DIMENSION1=MEMBER" "FREQUENCY_DIMENSION2=MEMBER"`*.

Exemple

```
epmautomate importSupplementalCollectionData "datafile.csv" "Journal Data
Collection" "FY20" "Jan" "Account=PAYROLL" "JournalID=LNR 113"
```

importSupplementalData

Importe des données supplémentaires d'un fichier dans l'application.

Utilisez la commande uploadFile pour charger le fichier contenant les données vers l'emplacement de chargement par défaut.

Champ d'application

Financial Consolidation and Close et Tax Reporting.

Rôles requis

Administrateur de service

Syntaxe

> **Remarque**
>
> Tous les paramètres de commande doivent être placés entre guillemets doubles.

`epmautomate importSupplementalData "FILE_NAME" "DATA_SET_NAME" "YEAR" "PERIOD_NAME" "SCENARIO_NAME"` où :

- `FILE_NAME` est le nom d'un fichier CSV, disponible à l'emplacement de téléchargement par défaut, qui contient les données supplémentaires au format correct.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `DATA_SET_NAME` est le nom de l'ensemble de données dans lequel les données supplémentaires figurant dans le fichier doivent être importées.
- `YEAR` est l'année pour laquelle l'ensemble de données est déployé.
- `PERIOD_NAME` est le nom de la période pour laquelle l'ensemble de données est déployé.
- `SCENARIO_NAME` est le nom du scénario pour lequel l'ensemble de données est déployé.

Exemple

```
epmautomate importSupplementalData "DatasetImport.csv" "EmployeeDataSet" "FY17"
"Jan" "Actual"
```

importTemplate

Crée une structure d'application en procédant à un import à partir d'un fichier de modèle qui existe dans profitinbox.

Un administrateur de service charge un fichier de modèle dans profitinbox à l'aide de la commande uploadFile.

Champ d'application

Profitability and Cost Management

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate importTemplate` *`APPLICATION_NAME File_Name`* `isApplicationOverwrite=true|false` où :

- *`APPLICATION_NAME`* est le nom de l'application Profitability and Cost Management que vous voulez créer en important le modèle.
- `File_Name` est le nom du fichier .ZIP contenant le modèle d'application. Ce fichier doit exister dans profitinbox.
- `isApplicationOverwrite` indique si l'application existante doit être remplacée, le cas échéant. Indiquez cette valeur de paramètre entièrement en minuscules.

Exemple

```
epmautomate importTemplate BksML12 template1.zip isApplicationOverwrite=true
```

importTMAttributeValues

Importe des valeurs dans les attributs de groupe du module Correspondance de transactions d'Account Reconciliation.

Un administrateur de service utilise la commande uploadFile pour charger le fichier d'import contenant les valeurs d'attribut correspondant aux transactions dans l'environnement avant d'exécuter cette commande.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Champ d'application

Account Reconciliation

Rôles requis

- Administrateur de service
- Super utilisateur (nécessite un niveau de sécurité supplémentaire fourni via des listes de contrôle d'accès)

Syntaxe

`epmautomate importTMAttributeValues` *`ATTRIBUTE_NAME FILE_NAME`* `[METHOD=REPLACE|REPLACE ALL|UPDATE][DATEFORMAT=DD/MM/YYYY|DD-MMM-YYYY|MMM d,yyyy|All]`, où :

- *`ATTRIBUTE_NAME`* est le nom de l'attribut de groupe dans lequel importer les valeurs.
- *`FILE_NAME`* est le fichier d'import CSV à partir duquel les valeurs doivent être importées dans le module Correspondance de transactions.
- `METHOD` (facultatif) indique la façon dont les valeurs doivent être importées. Valeurs valides :
  - `Replace` pour ajouter toutes les valeurs du fichier d'import dans les attributs de groupe du module Correspondance de transactions. Les valeurs d'attribut existantes sont remplacées par celles du fichier d'import ; les valeurs n'existant pas dans l'attribut mais présentes dans le fichier d'import sont ajoutées. Les valeurs existant dans l'attribut qui ne figurent pas dans le fichier d'import ne sont pas modifiées. Toutes les données d'attribut d'une valeur de clé particulière sont remplacées par le contenu du fichier ou effacées. Les nouvelles valeurs sont ajoutées en bas, dans leur ordre d'apparition dans le fichier.
    Ce type d'import s'avère particulièrement utile si vous déplacez seulement les dernières modifications à partir d'un système source, par exemple, lors de l'ajout de nouvelles données de magasin résultant d'une acquisition afin de ne remplacer que les valeurs d'attribut indiquées, si elles sont présentes, par les valeurs du fichier d'import. Il s'agit du comportement par défaut.
  - `Replace All` pour remplacer les valeurs d'attribut existantes par les valeurs de l'import. Les valeurs existant dans l'attribut mais ne figurant pas dans le fichier d'import sont supprimées.
    Ce type d'import s'avère particulièrement utile si vous mettez en miroir des valeurs à partir d'un système source avec une mise à jour complète, par exemple, pour les mises à jour hebdomadaires de synchronisation avec les données de magasin du système source.
  - `Update` pour remplacer ou ajouter toutes les valeurs du fichier d'import dans l'attribut. Les valeurs d'attribut existantes sont remplacées par celles du fichier d'import. Les valeurs se trouvant dans le fichier d'import mais ne figurant pas dans l'attribut sont ajoutées. Les valeurs existant dans l'attribut mais ne figurant pas dans le fichier d'import ne sont pas modifiées. Seules les données d'attribut d'une valeur de clé particulière sont remplacées par le contenu du fichier ; les données des attributs non disponibles dans le fichier demeurent intactes. Toute clé présente dans le fichier d'import mais pas dans l'attribut génère une erreur.
    Ce type d'import s'avère particulièrement utile si vous mettez à jour quelques attributs parmi la totalité des valeurs d'attribut, par exemple, lors de la mise à jour des directeurs de magasin après une réorganisation, sans que cela ait d'effet sur le reste des données relatives aux magasins.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `Dateformat` (facultatif) indique les formats de date valides (par exemple, `DD/MM/YYYY`, `DD-MMM-YYYY` (valeur par défaut), `MMM d,yyyy` et `All`) à analyser. Vous pouvez indiquer plusieurs valeurs de format de date en les séparant par un point-virgule.

Exemple

```
epmautomate importTMAttributeValues TMGA TMGA.csv METHOD=Replace DATEFORMAT="All"
```

importTmPremappedTransactions

Pour une source de données spécifique, importe les données de transactions pré-mappées à partir d'un fichier du référentiel Account Reconciliation dans le module Correspondance de transactions.

Un administrateur de service utilise la commande uploadFile pour charger le fichier de transactions vers le service.

Cette commande affiche le statut d'import et un nom de fichier journal d'import dans la console. Un administrateur de service utilise la commande downloadFile pour télécharger le fichier journal sur un ordinateur local.

Reportez-vous à la section Import de données du guide *Rapprochement de comptes avec Account Reconciliation* pour en savoir plus sur les exigences en matière de format de fichier d'import et pour obtenir des informations sur l'import des données.

> **Remarque**
>
> - Vous ne pouvez importer des transactions que pour un type de correspondance à la fois. Toutefois, les imports parallèles peuvent être exécutés vers différents types de correspondance.
> - Contrairement à ce qu'il se passe sur l'écran Jobs, vous pouvez importer les données de transactions pré-mappées à partir d'un seul fichier à la fois.
> - Après avoir importé les transactions pré-mappées pour toutes les sources de données, exécutez la commande `runautomatch`.

Champ d'application

Account Reconciliation

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmautomate importTmPremappedTransactions` *`MATCH_TYPE DATA_SOURCE FILE_NAME`* [*`DATE_FORMAT`*] où :

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `MATCH_TYPE` est un type de correspondance défini dans Account Reconciliation.
- `DATA_SOURCE` est l'identificateur de la source de données associée au type de rapprochement que vous avez indiqué.
- `FILE_NAME` est le nom du fichier CSV contenant les transactions à importer. Ce fichier doit être disponible dans le service.
- `DATE_FORMAT` est un paramètre facultatif indiquant le format des champs de date inclus dans le fichier d'import des transactions. La valeur par défaut est `dd-MMM-YYYY`. Autres formats de date pris en charge : `MM/dd/yyyy`, `dd/MM/yyyy`, `MM-dd-yyyy`, `d-M-yyyy` et `MMM d,yyyy`.

Exemple

```
epmautomate importTmPremappedTransactions "INTERCOMPANY" "AP" dailydata.csv d-M-yyyy
```

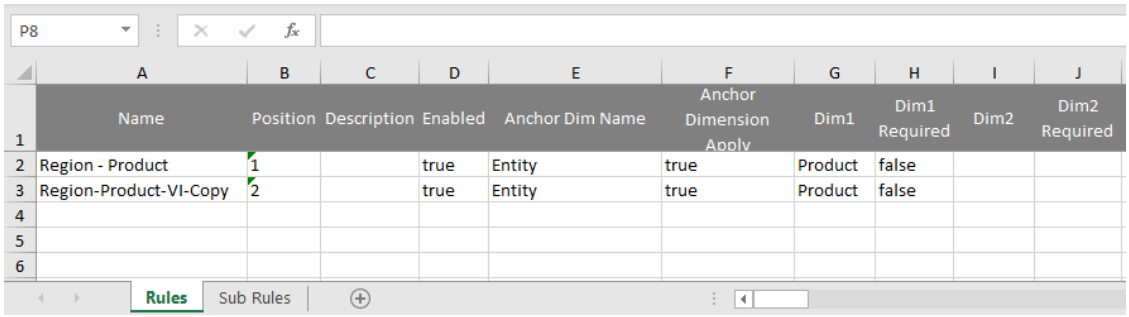
importValidIntersections

Importe les groupes de croisements valides à partir d'un fichier ZIP vers le processus métier. Avant d'exécuter cette commande, utilisez la commande uploadFile pour charger le fichier ZIP vers l'environnement.

Le fichier ZIP doit contenir un fichier Excel (contenant les définitions de croisement valide) avec deux feuilles de calcul : Rules et Sub Rules. La feuille de calcul Rules doit définir le groupe de croisements, dimensions comprises, ainsi que des propriétés telles que Les membres non spécifiés sont valides et Dimensions supplémentaires requises. La feuille de calcul Sub Rules doit fournir des sélections de membres et des exclusions. Pour plus d'informations, reportez-vous aux rubriques suivantes du guide *Administration de Planning*.

- Dimensions d'ancrage et de non-ancrage
- Exemples de croisements valides

La meilleure méthode pour obtenir le modèle de format de fichier d'import est d'exporter des croisements valides à partir de l'application. Les illustrations suivantes présentent un exemple de format.

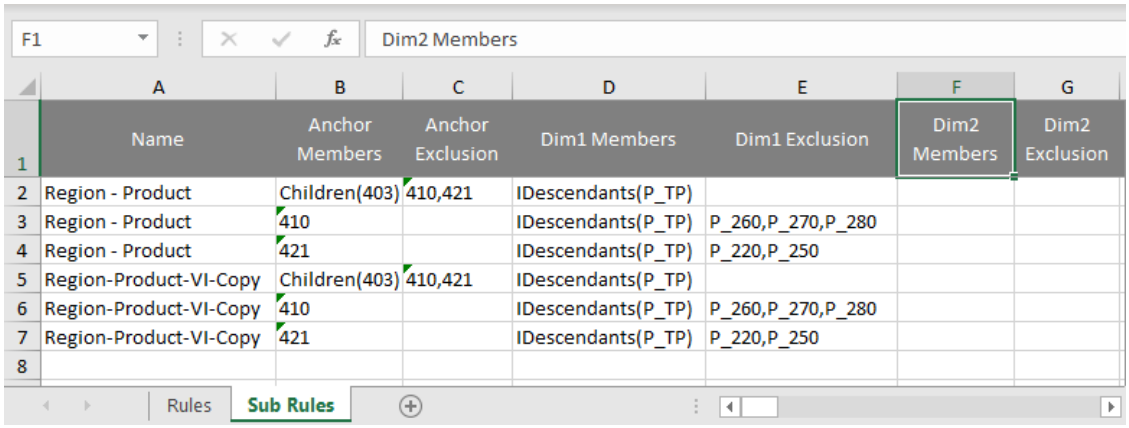


P8

	A	B	C	D	E	F	G	H	I	J
1	Name	Position	Description	Enabled	Anchor Dim Name	Anchor Dimension Apply	Dim1	Dim1 Required	Dim2	Dim2 Required
2	Region - Product	1		true	Entity	true	Product	false		
3	Region-Product-VI-Copy	2		true	Entity	true	Product	false		
4										
5										
6										

Rules | Sub Rules

Copyright © 2016, 2025, Oracle et/ou ses affiliés.



	A	B	C	D	E	F	G
1	Name	Anchor Members	Anchor Exclusion	Dim1 Members	Dim1 Exclusion	Dim2 Members	Dim2 Exclusion
2	Region - Product	Children(403)	410,421	IDescendants(P_TP)			
3	Region - Product	410		IDescendants(P_TP)	P_260,P_270,P_280		
4	Region - Product	421		IDescendants(P_TP)	P_220,P_250		
5	Region-Product-VI-Copy	Children(403)	410,421	IDescendants(P_TP)			
6	Region-Product-VI-Copy	410		IDescendants(P_TP)	P_260,P_270,P_280		
7	Region-Product-VI-Copy	421		IDescendants(P_TP)	P_220,P_250		
8							

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate importValidIntersections` *`FILE_NAME`*`.zip [ErrorFile=`*`ERROR_FILE_NAME`*`.txt]` où :

- `FILE_NAME` est le nom du fichier ZIP contenant le fichier Excel qui comporte la définition de croisement valide.
- `ErrorFile` (facultatif) identifie le nom du fichier texte dans lequel les enregistrements d'erreurs sont écrits. Si aucune valeur n'est indiquée pour ce paramètre, la commande génère automatiquement un fichier d'erreur dont vous pouvez voir le nom dans la console de jobs.

Exemple

```
epmautomate importValidIntersections VI_Import_File.zip
ErrorFile=VI_Import_Log.txt
```

invalidLoginReport

Crée un rapport sur les connexions non valides qui répertorie les tentatives de connexion à l'environnement ayant échoué au cours d'une période donnée correspondant à la période de conservation d'audit indiquée pour l'environnement. La période de conservation par défaut est de 30 jours. Vous pouvez la prolonger à 90 jours au maximum en modifiant le paramètre **Période de conservation d'audit (jours)** dans la console de gestion des identités Oracle Cloud. Pour conserver les données d'audit pendant plus de 90 jours, téléchargez et archivez régulièrement ce rapport ainsi que le rapport d'audit sur l'affectation de rôle.

Le rapport sur les connexions non valides contient des informations telles que les suivantes :

- Nom de l'utilisateur qui a tenté de se connecter
- Adresse IP distante à partir de laquelle l'utilisateur a tenté de se connecter
- Horodatage de la tentative de connexion

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Ce rapport présente toutes les tentatives de connexion à l'instance Identity Cloud Service correspondante ayant échoué. Elles ne sont pas nécessairement toutes liées à une seule instance Oracle Fusion Cloud Enterprise Performance Management.

	A	B	C
1	User Name	IP Address	Access Date and Time
2	john.doe@example.com	xxx.xx.xx.xx5	July 15, 2021 11:14:58 UTC
3	jane.doe@example.com	xxx.xx.xx.xx9	July 15, 2021 11:14:58 UTC
4	john.smith@example.com	xxx.xx.xx.xx3	July 15, 2021 11:14:57 UTC

Utilisez la commande downloadFile pour télécharger le rapport sur un ordinateur local.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de domaine d'identité et tout rôle prédéfini

Syntaxe

`epmAutomate invalidLoginReport FROM_DATE TO_DATE FILE_NAME.CSV`, où :

- *FROM_DATE* indique la date de début (au format `YYYY-MM-DD`) de la période pour laquelle le rapport doit être généré.
- *TO_DATE* indique la date de fin (au format `YYYY-MM-DD`) de la période pour laquelle le rapport doit être généré.
- *FILE_NAME* est le nom du fichier CSV pour le rapport.

> **Note**
>
> Cloud EPM s'assure que seule une plage de dates valide est utilisée lors de la génération du rapport. Les validations suivantes sont effectuées pour les dates *de début* et *de fin* :
>
> - La date *FROM_DATE* ne peut pas être antérieure à la période de conservation maximale autorisée à compter de la date actuelle. Cette date doit être comprise dans la plage de valeurs du paramètre **Période de conservation des données d'audit** de la console de gestion des identités Oracle Cloud.
> - La date *TO_DATE* ne peut pas être postérieure à la période de conservation maximale à compter de la date *FROM_DATE*.
> - La date *TO_DATE* ne peut pas être antérieure à la date *FROM_DATE*.

Exemple

```
epmAutomate invalidLoginReport 2021-06-01 2021-06-30 invalidLoginReport.CSV
```

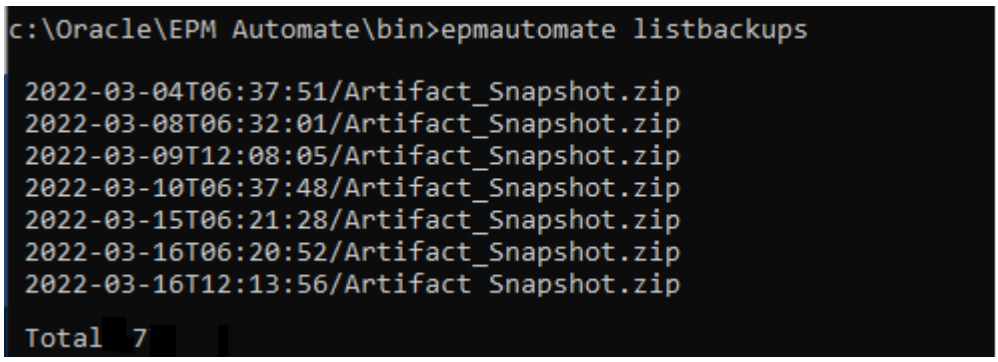

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

listBackups

Répertorie les instantanés de sauvegarde disponibles des environnements afin de déterminer si une sauvegarde spécifique est disponible pour que vous puissiez l'archiver ou l'utiliser afin de restaurer vous-même l'environnement actuel.

Avant d'essayer de restaurer une sauvegarde spécifique, utilisez cette commande pour vérifier si la sauvegarde requise est disponible dans Oracle Object Storage. Si c'est le cas, vous pouvez la restaurer (la copier dans votre environnement) en exécutant la commande restoreBackup. Une fois la sauvegarde copiée, vous pouvez l'importer à l'aide de la commande importSnapshot. La restauration en libre-service de l'environnement vous fait gagner du temps de traitement.

Pour les services autres que Narrative Reporting, cette commande répertorie les instantanés de sauvegarde disponibles (créés par le processus de maintenance quotidienne) en utilisant la convention de dénomination `YYYY-MM-DDTHH:MM:SS/Artifact_Snapshot.zip`, par exemple : `2022-02-16T21:00:02/Artifact_Snapshot.zip`. Pour Narrative Reporting, les instantanés disponibles utilisent la convention de dénomination `YYYY-MM-DDTHH:MM:SS/EPRCS_Backup.tar.gz`, par exemple : `2022-02-16T21:00:02/EPRCS_Backup.tar.gz`. Dans les deux cas, l'horodatage reflète l'heure UTC de création de l'instantané. L'illustration suivante présente un exemple de sortie de commande.



Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Syntaxe

```
epmAutomate listBackups
```

Exemple

```
epmAutomate listBackups
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

listFiles

Répertorie les noms des fichiers qui se trouvent à l'emplacement par défaut, dans les dossiers de gestion des données et dans profitinbox/profitoutbox (Profitability and Cost Management).

Cette commande répertorie également les fichiers d'export incrémentiel et de sauvegarde, les instantanés de migration, les journaux d'accès et les rapports d'activité. Cette illustration montre une version tronquée de la sortie de commande.

```
apr/2022-01-27 05_23_36/activityreport.json
apr/2022-01-28 05_24_07/2022-01-28 05_24_07.html
apr/2022-01-28 05_24_07/access_log.zip
apr/2022-01-28 05_24_07/activityreport.json
apr/2022-01-29 05_24_06/2022-01-29 05_24_06.html
apr/2022-01-29 05_24_06/access_log.zip
outbox/Vision_99.dat
roleassign.csv
RoleAssignment.csv
sanity_no_data_22-01-18.zip
U-1.csv
U2.csv
user1.csv
user12.csv
users12.csv
Uservariables-MemberFormula.zip
UsrGrpReport.CSV
Vision_DTsetup.zip
VisionADCForms2010.zip
```

Cette commande ne répertorie pas l'instantané actuel si elle est exécutée lors de la génération de l'instantané de l'environnement, par exemple lors de la maintenance quotidienne.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Syntaxe

```
epmautomate listFiles
```

Exemple

```
epmautomate listFiles
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

loadData

Charge des données dans un cube de calcul à l'aide d'un fichier disponible dans profitinbox.

Un administrateur de service utilise la commande uploadFile pour charger des fichiers dans profitinbox.

Champ d'application

Profitability and Cost Management

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate loadData` *`APPLICATION_NAME`* `dataFileName=`*`File_Name`* `PARAMETER=`*`VALUE`*, où :

- *`APPLICATION_NAME`* est le nom de l'application Profitability and Cost Management dans laquelle vous voulez charger les données.
- `dataFileName=`*`File_Name`* indique un fichier de chargement de données disponible dans profitinbox.
  Les noms de fichier de données doivent être entourés de guillemets doubles.
- `PARAMETER=`*`VALUE`* indique les paramètres d'exécution et leurs valeurs pour charger des données. Indiquez autant de paires paramètre-valeur que le processus l'exige. Paramètres valides et leurs valeurs :
  - `clearAllDataFlag=true|false` indique si les données existantes doivent être effacées dans le cube d'application
  - `dataLoadValue=OVERWRITE_EXISTING_VALUES|ADD_TO_EXISTING` indique comment gérer les données existantes

Exemple

```
epmautomate loadData BksML12 dataFileName="data1.txt"clearAllDataFlag=true
dataLoadValue="OVERWRITE_EXISTING_VALUES"
```

loadDimensionViewpoint

Charge des données dans un point de vue (un sous-ensemble de noeuds) non lié, lié ou partiellement lié à partir d'un fichier de chargement à l'aide d'un chargement défini.

Le fichier de chargement (fichier CSV, fichier Excel au format XLSX ou fichier ZIP contenant un fichier CSV ou XLSX) doit être disponible dans l'environnement dans lequel vous chargez le point de vue. Vous pouvez charger le fichier de chargement vers l'environnement à l'aide de la commande uploadFile ou copyFileFromInstance.

Pour charger un point de vue à l'aide de la définition de chargement par défaut, utilisez la commande loadViewpoint.

Champ d'application

Oracle Fusion Cloud Enterprise Data Management.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate loadDimensionViewpoint` *`APPLICATION_NAME DIMENSION_NAME FILE_NAME LOAD_NAME`* `[loadOption=ReplaceNodes|Merge] [purpose="PURPOSE"]`, où :

- `APPLICATION_NAME` est le nom d'une application Cloud EDM.
- `DIMENSION_NAME` est le nom de la dimension à charger.
- `FILE_NAME` est le nom du fichier (avec extension CSV, XLSX ou ZIP), à partir duquel le point de vue doit être chargé.
- `LOAD_NAME` est le nom du chargement défini à utiliser pour charger le point de vue.
- `loadOption` (facultatif) identifie comment charger le point de vue. Les options de chargement valides sont les suivantes :
  - `ReplaceNodes` permet d'effacer de la hiérarchie toutes les relations (y compris les relations orphelines et celles employées par d'autres points de vue utilisant le même ensemble de hiérarchies) autres que celles du fichier de chargement. Il s'agit du type de chargement par défaut.
  - `Merge` permet de conserver les relations existantes en traitant uniquement les modifications incrémentielles.
- `purpose` (facultatif) est une chaîne de texte placée entre guillemets qui identifie le motif du chargement du point de vue.

Exemples

- Chargement d'un point de vue remplaçant les hiérarchies existantes : `epmautomate loadDimensionViewPoint "Data Warehouse" Citizen Data_Warehouse_Citizen_20241108.csv DW-Citizen`
- Chargement d'un point de vue en fusionnant les modifications incrémentielles avec le chargement indiqué : `epmautomate loadDimensionViewPoint"Data Warehouse" Citizen Data_Warehouse_Citizen_20241108.csv DW-Citizen loadOption=Merge purpose="Load Test"`

loadDimData

Charge des métadonnées de dimension à partir de fichiers dans profitinbox dans une application.

Un administrateur de service utilise la commande uploadFile pour charger des fichiers de métadonnées dans profitinbox.

Champ d'application

Profitability and Cost Management

Rôles requis

- Administrateur de service
- Super utilisateur

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Syntaxe

`epmautomate loadDimData APPLICATION_NAME dataFileName=File_Name [stringDelimiter="DELIMITER"] [acceptableDecreasePercentage=PERCENTAGE]` où :

- *APPLICATION_NAME* est le nom de l'application Profitability and Cost Management dans laquelle vous voulez charger les métadonnées de dimension.
- `dataFileName` indique un fichier de chargement de métadonnées de dimension disponible dans profitinbox. Pour charger des métadonnées à partir de plusieurs fichiers, répertoriez les noms de fichier différenciés par un séparateur.
- `stringDelimiter` (facultatif) indique le séparateur utilisé pour séparer les noms de fichier de métadonnées. Le séparateur doit être placé entre guillemets doubles.
- `acceptableDecreasePercentage` (facultatif) spécifie la différence en pourcentage (sans le symbole %) au niveau du nombre de membres qui sera acceptable pour l'opération. Si le nombre de nouveaux membres provenant du fichier entrant est inférieur au nombre de membres existants, cette valeur représente le pourcentage de diminution acceptable. Le chargement des données de dimension échoue si l'écart du nombre de membres dépasse ce pourcentage.

Exemple

```
epmautomate loadDimData BksML12 dataFileName="dimdata1.txt#dimdata1.txt"
stringDelimiter="#" acceptableDecreasePercentage=5
```

loadViewpoint

Charge un point de vue (un sous-ensemble de noeuds) d'un fichier de chargement vers une application Oracle Fusion Cloud Enterprise Data Management.

Les chargements de point de vue permettent de charger des données dans des points de vue non liés, liés ou partiellement liés. Le fichier de chargement de point de vue (fichier CSV, fichier Excel au format XLSX ou fichier ZIP contenant un fichier CSV ou XLSX), doit être disponible dans l'environnement dans lequel vous chargez le point de vue. Vous pouvez charger le fichier de chargement vers l'environnement à l'aide de la commande uploadFile ou copyFileFromInstance.

Champ d'application

Cloud EDM

Rôles requis

Administrateur de service

Syntaxe

`epmautomate loadViewpoint VIEW VIEWPOINT PURPOSE FILE_NAME [loadType=ReplaceNodes|Merge]`
, où :

- `VIEW` est le nom d'une vue Cloud EDM.
- `VIEWPOINT` est le nom du point de vue à charger.
- `PURPOSE` est une chaîne de texte, placée entre guillemets, indiquant pourquoi le point de vue est chargé.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `FILE_NAME` est le nom du fichier, avec extension, à partir duquel le point de vue doit être chargé.
- `loadType` (facultatif) identifie comment charger le point de vue. Les valeurs valides sont `Merge` et `ReplaceNodes`.
  - Utilisez `Merge` pour conserver les relations existantes en traitant les modifications incrémentielles.
  - Utilisez `ReplaceNodes` pour effacer de la hiérarchie toutes les relations (y compris les relations orphelines et celles employées par d'autres points de vue utilisant le même ensemble de hiérarchies) autres que celles du fichier de chargement. Il s'agit du type de chargement par défaut.

Exemples

- **Fusionner les modifications incrémentielles :** `epmautomate loadViewpoint USOperations Entity "Daily Upstream Load" data_Entity.CSV loadType=Merge`
- **Remplacer les hiérarchies existantes :** `epmautomate loadViewpoint USOperations Entity "Replace US Operations data" data_Entity.CSV`

login

Etablit une connexion sécurisée à un environnement. Cette commande permet de se connecter à un environnement à l'aide d'un mot de passe en texte brut ou d'un fichier de mots de passe cryptés contenant le mot de passe ou le jeton d'actualisation OAuth 2.0.

Connectez-vous pour lancer une session, qui reste active jusqu'à ce que vous vous déconnectiez.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Remarque

- Cette commande n'est pas prise en charge pour les utilisateurs qui sont configurés pour une authentification de base avec l'authentification à plusieurs facteurs.
- EPM Automate ne prend pas en charge la connexion à l'aide des informations d'identification d'authentification unique de votre organisation.
- EPM Automate ne fonctionne pas avec un proxy SOCKS, mais uniquement avec un proxy HTTP/HTTPS.
- Lors de l'utilisation de cette commande dans des fichiers batch pour automatiser les activités, Oracle vous recommande d'utiliser un jeton d'actualisation OAuth 2.0 ou un mot de passe crypté pour éviter l'enregistrement de mots de passe en clair dans les fichiers batch.
- Sur les ordinateurs Windows, cette commande identifie automatiquement le certificat de sécurité intermédiaire de serveur proxy manquant susceptible de vous empêcher d'établir une connexion et l'ajoute à l'environnement JRE installé sous `C:\Oracle\EPM Automate`. Cela permet d'éviter les erreurs de connexion en lien avec les certificats de sécurité lorsque des serveurs proxy sont utilisés pour accéder à Internet.
  Sur les ordinateurs Linux, la commande login identifie le certificat de sécurité manquant du serveur proxy, le télécharge et affiche une erreur. Un utilisateur disposant d'un accès `root` peut alors installer le certificat téléchargé dans l'environnement JRE disponible dans le répertoire de base `JAVA_HOME` identifié dans les variables d'environnement. Reportez-vous aux sources d'information suivantes :
  - Environnement JRE (Java Runtime Environment) et EPM Automate
  - Documentation sur l'utilitaire keytool Java

Si vous utilisez une version plus ancienne, lors de la connexion, un message vous indique de mettre à niveau EPM Automate. Vous pouvez utiliser la commande upgrade pour mettre à niveau l'installation sans invite.

Si vous prévoyez d'exécuter la commande addUsers, removeUsers, assignRole ou unassignRole, ne vous connectez pas à l'aide du jeton d'actualisation OAuth. Ces commandes exigent que vous utilisiez l'authentification de base. Toutes les autres commandes fonctionnent avec OAuth 2.0.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Syntaxe

- **Utilisation d'un mot de passe non crypté :** `epmautomate login USERNAME PASSWORD URL [IDENTITYDOMAIN] [ProxyServerUserName=PROXY_USERNAME ProxyServerPassword=PROXY_PASSWORD ProxyServerDomain=PROXY_DOMAIN] ] [KeystorePassword=PASSWORD]`
- **Utilisation d'un fichier crypté :** `epmautomate login USERNAME PASSWORD_FILE URL [IDENTITYDOMAIN] [ProxyServerUserName=PROXY_USERNAME] [ProxyServerPassword=PROXY_PASSWORD] [ProxyServerDomain=PROXY_DOMAIN] [KeystorePassword=KEYSTORE_PASSWORD]`

Dans ces commandes :

- *USERNAME* représente le nom de l'utilisateur.

  > **Remarque**
  >
  > Il s'agit d'un paramètre requis, mais sa valeur est utilisée pour l'authentification de base uniquement. Pour l'authentification basée sur OAuth2, cette valeur est dérivée du jeton OAuth2. La valeur fournie lors de l'exécution de cette commande est ignorée.

- *PASSWORD* représente le mot de passe de l'utilisateur.
- *PASSWORD_FILE* représente le nom et l'emplacement du fichier qui stocke le jeton d'actualisation OAuth 2.0 ou le mot de passe crypté de l'utilisateur. Reportez-vous à la commande encrypt.
- *URL* est l'URL de base de l'environnement auquel vous devez vous connecter. Vous pouvez utiliser une URL personnalisée à la place de l'URL Oracle Fusion Cloud Enterprise Performance Management. Reportez-vous à la section Utilisation d'URL personnalisées du guide *Guide de mise en route pour les administrateurs*.

  > **Remarque**
  >
  > Si vous utilisez une passerelle d'API ou un proxy inverse, servez-vous de son URL et du contexte défini pour votre environnement au lieu de l'URL Cloud EPM.

- *IDENTITYDOMAIN* (facultatif) est le domaine d'identité de l'environnement.
  Cette valeur est automatiquement dérivée de l'URL Cloud EPM. Toute valeur indiquée est ignorée.
- `ProxyServerUserName` est le nom d'utilisateur qui permet de s'authentifier pour une session sécurisée auprès du serveur proxy HTTP qui contrôle l'accès à Internet. Indiquez le nom d'utilisateur sans ajouter de préfixe de nom de domaine. Nécessaire uniquement si l'authentification auprès du serveur proxy est activée pour votre réseau.
- `ProxyServerPassword` est le mot de passe qui permet d'authentifier l'utilisateur auprès du serveur proxy. Nécessaire uniquement si l'authentification auprès du serveur proxy est activée pour votre réseau. Ce mot de passe peut être crypté. Reportez-vous à la commande encrypt. Si le mot de passe est crypté, il est lu à partir du fichier *PASSWORD_FILE*.
- `ProxyServerDomain` est le nom du domaine défini pour le serveur proxy HTTP (et non le nom du serveur ou le nom d'hôte du serveur proxy). Nécessaire uniquement si l'authentification auprès du serveur proxy est activée pour votre réseau et qu'un domaine de serveur proxy est configuré.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `KeystorePassword` (facultatif) est le mot de passe de fichier de clés d'accès requis pour l'import du certificat de sécurité du serveur proxy. N'utilisez ce paramètre que sous Windows, et uniquement si vous obtenez les erreurs suivantes dans des environnements où un serveur proxy est utilisé pour permettre l'accès à Internet :

  ```
  EPMAT-7 : Impossible de se connecter car certains certificats SSL sont manquants
  dans le fichier de clés d'accès

  EPMAT-7 : Impossible de se connecter car les certificats SSL mentionnés ci-avant
  sont manquants dans le fichier de clés d'accès
  ```

> **Remarque**
>
> EPM Automate détecte et utilise les paramètres de proxy HTTP/HTTPS sur votre ordinateur.
> EPM Automate prend en charge les mécanismes d'authentification suivants pour se connecter au serveur proxy :
>
> - Authentification de base
> - Authentification Digest
> - Authentification Kerberos
> - Authentification par proxy Negociate
> - Authentification NTLM
>
> La méthode d'authentification disponible et sa configuration dépendent du serveur proxy que vous utilisez.
>
> Sur les ordinateurs Linux, si les paramètres de proxy requièrent que vous vous authentifiiez auprès du serveur proxy, vous devez saisir le domaine de serveur proxy, le nom d'utilisateur et le mot de passe en tant que paramètres dans cette commande. Contactez l'administrateur réseau si vous avez besoin d'aide avec les informations d'identification et le nom de domaine de serveur proxy.

Exemples

- Utilisation d'un mot de passe Cloud EPM non crypté, aucune authentification de proxy :
  ```
  epmautomate login serviceAdmin P@ssword1 https://test-cloud-
  pln.pbcs.us1.oraclecloud.com
  ```
- Utilisation d'un fichier crypté, aucune authentification de proxy :
  ```
  epmautomate login serviceAdmin C:\mySecuredir\password.epw https://test-cloud-
  pln.pbcs.us1.oraclecloud.com
  ```
- Utilisation d'un fichier crypté, si l'authentification auprès du serveur proxy est activée avec un domaine de serveur :
  ```
  epmautomate login serviceAdmin C:\mySecuredir\password.epw https://test-cloud-
  pln.pbcs.us1.oraclecloud.com ProxyServerUserName=john.doe@example.com
  ProxyServerPassword=example ProxyServerDomain=example
  ```
- Utilisation d'un fichier crypté, si l'authentification auprès du serveur proxy est activée sans domaine de serveur :
  ```
  epmautomate login serviceAdmin C:\mySecuredir\password.epw https://test-cloud-
  pln.pbcs.us1.oraclecloud.com ProxyServerUserName=john.doe@example.com
  ProxyServerPassword=example
  ```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Utilisation d'un mot de passe Cloud EPM crypté et d'un mot de passe de serveur proxy, si l'authentification auprès du serveur proxy est activée avec un domaine de serveur :
  ```
  epmautomate login serviceAdmin C:\mySecuredir\password.epw https://test-cloud-pln.pbcs.us1.oraclecloud.com ProxyServerUserName=john.doe@example.com ProxyServerDomain=example
  ```
- Utilisation d'un mot de passe Cloud EPM crypté et d'un mot de passe de serveur proxy, si l'authentification auprès du serveur proxy est activée sans domaine de serveur :
  ```
  epmautomate login serviceAdmin C:\mySecuredir\password.epw https://test-cloud-pln.pbcs.us1.oraclecloud.com ProxyServerUserName=john.doe@example.com
  ```
- Utilisation d'un fichier crypté avec un jeton d'actualisation OAuth2 :
  ```
  epmautomate login ignoredUsername C:\mySecuredir\OAuth2.epw https://test-cloud-pln.epm.us-ashburn-1.ocs.oraclecloud.com
  ```

> **Remarque**
>
> Dans cette commande, EPM Automate ignore le nom d'utilisateur *ignoredUsername* indiqué. Le nom d'utilisateur employé pour se connecter à l'environnement est dérivé du jeton OAuth2.

- Utilisation d'un fichier crypté avec une passerelle d'API APIGEE :
  ```
  epmautomate login serviceAdmin C:\mySecuredir\password.epw https://exampleapigee.apigee.com/epm example_ID_DOM
  ```
- Utilisation d'une URL personnalisée :
  ```
  epmautomate login serviceAdmin C:\mySecuredir\password.epw https://rebrand.ly/Automate
  ```

logout

Interrompt la connexion en cours à un environnement.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Syntaxe

```
epmautomate logout
```

Exemple

```
epmautomate logout
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

maskData

Masque les données d'application afin d'assurer la confidentialité des données. Utilisez cette commande uniquement sur des environnements de test pour cacher les données sensibles aux développeurs d'application.

AVERTISSEMENT : n'utilisez pas cette commande sur des environnements de production car elle rend aléatoires les données d'application en cours, les rendant donc dénuées de sens. Vous ne pouvez pas annuler les effets de cette commande. Si vous avez masqué par erreur les données dans un environnement de service, vous devez les restaurer à partir d'une sauvegarde ou de l'instantané de maintenance.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate maskData [-f]` où `-f` est une option permettant de forcer le démarrage du processus de masquage sans confirmation de l'utilisateur. Si vous n'utilisez pas l'option `-f`, la commande vous invite à confirmer votre action.

Exemple

```
epmautomate maskData [-f]
```

mergeDataSlices

Fusionne toutes les tranches de données incrémentielles d'un cube en mode "aggregate storage" dans la tranche de base de données principale et enlève éventuellement les cellules dont la valeur est égale à zéro.

Champ d'application

Planning, modules Planning, FreeForm, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate mergeDataSlices` *`CUBE_NAME`* `[keepZeroCells=true|false]` où :

- `CUBE_NAME` identifie le cube en mode "aggregate storage" dont toutes les tranches de données doivent être fusionnées.
- `keepZeroCells` (facultatif) indique si les cellules dont la valeur est égale à zéro doivent être enlevées (l'effacement logique des données d'une région a pour résultat une cellule dont la valeur est égale à zéro). La valeur par défaut est `true`.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exemple

```
epmautomate mergeDataSlices rep1 keepZeroCells=false
```

mergeSlices

Fusionne les secteurs de données incrémentielles dans le cube de base de données principal et, éventuellement, enlève les cellules Oracle Essbase contenant la valeur 0 (zéro) pour rendre le cube compact.

La suppression des cellules contenant la valeur 0 optimise les performances du cube.

Champ d'application

Profitability and Cost Management

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate mergeSlices applicationName [removeZeroCells=true|false]` où :

- `applicationName` est le nom d'une application Profitability and Cost Management.
- `removeZeroCells` indique si les cellules contenant la valeur 0 doivent être enlevées (facultatif). La valeur par défaut de ce paramètre est `false`.

Exemples

- Fusionner les secteurs sans enlever les cellules contenant des 0:
  - `epmautomate mergeSlices BksML30`
  - `epmautomate mergeSlices BksML30 removeZeroCells=false`
- Fusion de secteurs et suppression des cellules avec une valeur nulle : `epmautomate mergeSlices BksML30 removeZeroCells=true`

optimizeASOCube

Optimise les performances des requêtes de sélection de vues agrégées pour l'extraction de données à partir de cubes ASO.

Cette commande vous permet d'effectuer des opérations d'optimisation des requêtes sur des cubes ASO lorsque l'agrégation par défaut est jugée insuffisante pour répondre à vos besoins en matière de génération de rapports ou d'extraction de données en raison de la taille importante des données. Le processus d'optimisation classique est le suivant :

- Supprimez les agrégations par défaut et reposant sur des requêtes.
- Démarrez le suivi des requêtes.
- Exécutez des exemples de requête à partir du gestionnaire de requêtes Profitability and Cost Management, d'Oracle Smart View for Office ou de la gestion des données, et toute autre requête MDX représentative du type de requête à optimiser pour entraîner Oracle Essbase.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Créez une agrégation en fonction de requêtes par défaut ou optimisées.

Champ d'application

Profitability and Cost Management

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate optimizeASOCube` *`APPLICATION_NAME OPTIMIZATION_TYPE`*, où :

- *`APPLICATION_NAME`* est le nom de l'application Profitability and Cost Management à laquelle le cube ASO appartient.
- *`OPTIMIZATION_TYPE`* est une opération d'optimisation de cube. Les valeurs acceptables sont les suivantes :
  - `clearAggregations`, qui enlève les vues par défaut et reposant sur des requêtes.
  - `createAggregations`, qui crée des vues agrégées Essbase par défaut. Utilisez cette option pour effectuer une agrégation par défaut plutôt qu'une agrégation reposant sur des requêtes.
  - `startQueryTracking`, qui démarre le suivi des requêtes.
  - `stopQueryTracking`, qui arrête le suivi des requêtes. Utilisez cette option pour empêcher Essbase de collecter des informations d'optimisation. Essbase continue à collecter des informations d'optimisation jusqu'à ce que vous arrêtiez le suivi des requêtes ou Essbase. Essbase peut agréger des vues en fonction des données collectées jusqu'à l'arrêt du suivi des requêtes.
  - `createQBOAggregations`, qui crée des vues agrégées Essbase en fonction des requêtes optimisées exécutées après l'activation du suivi des requêtes.

Exemples

- Suppression des vues agrégées par défaut et reposant sur des requêtes :
  `epmautomate optimizeASOCube BksML12 clearAggregations`
- Démarrage du suivi des requêtes :
  `epmautomate optimizeASOCube BksML12 startQueryTracking`
- Création de vues agrégées Essbase en fonction des requêtes optimisées exécutées après le démarrage du suivi des requêtes :
  `epmautomate optimizeASOCube BksML12 createQBOAggregations`

programDocumentationReport

Crée le rapport de documentation de programme contenant la logique d'application Profitability and Cost Management.

Vous pouvez télécharger le rapport sur un ordinateur local à l'aide de la commande downloadFile.

Champ d'application

Profitability and Cost Management

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Syntaxe

`epmautomate programDocumentationReport` *`APPLICATION_NAME POV_NAME`* `[fileName=`*`FILE_NAME]`* `[fileType=PDF|WORD|EXCEL|HTML] [useAlias=true|false] [skipFilters=true|false] stringDelimiter=`*`"DELIMITER"`*, où :

- *`APPLICATION_NAME`* est le nom de l'application Profitability and Cost Management pour laquelle le rapport de documentation de programme doit être créé.
- *`POV_NAME`* est le nom du PDV de modèle dans l'application pour laquelle le rapport doit être généré.
- `fileName` (facultatif) est le nom unique (extension comprise) du fichier de rapport. Le nom du fichier de rapport par défaut est `HPCMMLProgramDocumentationReport_`*`APPLICATION_NAME_POV_NAME`*`.pdf`.
- `fileType` (facultatif) est le format du fichier de sortie. La valeur par défaut est `PDF`.
- `useAlias` (facultatif) indique si les alias doivent être imprimés au lieu des noms de membre. La valeur par défaut est `false`.
- `skipFilters` (facultatif) indique si les filtres doivent être ignorés afin d'accélérer la génération du rapport pour les modèles volumineux qui contiennent de nombreuses règles utilisant des filtres. La valeur par défaut est false.
  Si vous définissez ce paramètre sur `true`, la commande ignore le processus de résolution de chaque filtre de règle permettant de déterminer les valeurs suivantes dans le rapport : nombre de sources estimé, nombre de destinations estimé, nombre de cibles estimé. A la place, le processus utilise les nombres du membre de niveau 0 correspondant. Si ce paramètre est défini sur `false` ou s'il n'est pas spécifié, la commande résout tous les filtres afin de générer des nombres plus précis.
- `stringDelimiter` est le séparateur utilisé dans les valeurs de PDV. Le séparateur doit être placé entre guillemets doubles.

Exemples

- Création du rapport après résolution des filtres de règle :
  ```
  epmautomate programDocumentationReport BksML30 2024_Feb_Actual fileName=Feb-Actual.xls fileType=Excel useAlias=true stringDelimiter="_"
  ```
- Création du rapport sans prendre en compte les filtres :
  ```
  epmautomate programDocumentationReport BksML30 2024_Feb_Actual fileName=Feb-Actual.xls fileType=Excel useAlias=true skipFilters=true stringDelimiter="_"
  ```

provisionReport

Génère un rapport sur l'affectation de rôle (fichier CSV) et le stocke à l'emplacement de téléchargement par défaut.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Le rapport répertorie les rôles prédéfinis (par exemple, Super utilisateur nom-service) et les rôles d'application (par exemple, Allocation en masse, qui est un rôle d'application Planning) affectés aux utilisateurs. Utilisez la commande downloadFile pour télécharger le rapport.

Deux versions du rapport peuvent être générées : une version simplifiée ou une version classique. Le rapport simplifié, qui est identique au rapport sur l'affectation de rôle disponible sur l'écran Contrôle d'accès, ne répertorie pas les rôles d'application inclus dans les rôles prédéfinis ou les rôles de composant des rôles d'application affectés à l'utilisateur. La version classique du rapport répertorie les rôles de composant inclus dans les rôles prédéfinis auxquels sont affectés les utilisateurs. Elle répertorie également les rôles d'application affectés à l'utilisateur (directement ou via des groupes).

La génération de ce rapport actualise les informations sur l'utilisateur et sur les rôles disponibles dans Contrôle d'accès.

Oracle Fusion Cloud Enterprise Performance Management considère les utilisateurs désactivés comme des utilisateurs auxquels aucun rôle prédéfini n'a été affecté, même si ces utilisateurs avaient un rôle prédéfini au moment de leur désactivation. Les informations sur les utilisateurs désactivés ne sont pas incluses dans ce rapport.

> **Remarque**
>
> Cette commande sera désapprouvée dans une prochaine version. Utilisez plutôt la commande roleAssignmentReport qui génère un rapport équivalent.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Afficher

Syntaxe

`epmautomate provisionReport` *`REPORT_NAME`* `[format=classic|simplified] [userType=serviceUsers|IDAdmins]` où :

- *`REPORT_NAME`* est le nom du rapport.
- `format` (facultatif) indique la façon dont le rapport doit être formaté. Valeurs acceptables :
  - `simplified` (option par défaut) crée un rapport identique au rapport sur l'affectation de rôle généré à partir de l'écran Contrôle d'accès.
  - `classic` crée un rapport qui répertorie les rôles de composant inclus dans les rôles prédéfinis auxquels sont affectés les utilisateurs. Elle répertorie également les rôles d'application affectés à l'utilisateur (directement ou via des groupes).
- `userType` (facultatif) identifie les utilisateurs à inclure dans le rapport. Si vous ne spécifiez pas de valeur pour ce paramètre, la valeur par défaut `serviceUsers` est utilisée. Valeurs acceptables :

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

  - La valeur `serviceUsers` crée un rapport qui contient des informations sur tous les utilisateurs fonctionnels (n'inclut pas les administrateurs de domaine d'identité s'ils ne disposent pas d'un rôle prédéfini qui leur donne accès à l'application).
  - `IDAdmins` crée un rapport qui répertorie uniquement les utilisateurs affectés au rôle Administrateur de domaine d'identité. Le rapport est identique au format classique et simplifié.

Exemples

- Création d'un rapport classique : `epmautomate provisionReport myProvReport.CSV format=classic`
- Création d'un rapport simplifié :
  - `epmautomate provisionReport myProvReport.CSV format=simplified`
  - `epmautomate provisionReport myProvReport.CSV userType=serviceUsers`
- Création d'un rapport répertoriant uniquement les administrateurs de domaine d'identité :
  - `epmautomate provisionReport myProvReport.CSV userType=IDAdmins`
  - `epmautomate provisionReport myProvReport.CSV userType=IDAdmins format=classic`

purgeArchivedTmTransactions

Purge les transactions mises en correspondance archivées de l'application Account Reconciliation.

Utilisez régulièrement la commande archiveTmTransactions afin d'archiver les anciennes transactions mises en correspondance, puis exécutez cette commande pour les enlever d'Account Reconciliation afin de garantir une taille d'application optimale.

Champ d'application

Account Reconciliation

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmautomate purgeArchivedTMTransactions JobID=`*`JOB_ID`* où `JobID` est l'identificateur du job d'archivage des transactions Correspondance de transactions exécuté pour archiver les transactions mises en correspondance. Cet ID de job est affiché dans la console EPM Automate lorsque vous exécutez la commande archiveTmTransactions. Vous pouvez également le trouver dans la console de jobs.

Exemple

```
epmautomate purgeArchivedTMTransactions JobID=100000002655003
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

purgeTmTransactions

Enlève les transactions mises en correspondance d'Account Reconciliation.

Champ d'application

Account Reconciliation

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmautomate purgeTmTransactions matchType age [filterOperator=`*`VALUE`*`] [filterValue=`*`VALUE`*`] [logFilename=`*`FILE_NAME`*`]`, où :

- `matchType` est l'identificateur (TextID) du type de correspondance dont les transactions mises en correspondance doivent être supprimées.
- `age` indique le nombre de jours écoulés depuis la mise en correspondance de la transaction. Toute transaction mise en correspondance antérieure ou égale à cette valeur sera supprimée.
- `filterOperator` (facultatif) est l'une des conditions de filtre suivantes permettant d'identifier les comptes qui contiennent des transactions mises en correspondance pour suppression. Cette valeur est associée à `filterValue` pour identifier les comptes desquels les transactions mises en correspondance doivent être purgées :
  - `equals`
  - `not_equals`
  - `starts_with`
  - `ends_with`
  - `contains`
  - `not_contains`
- `filterValue` (facultatif) est une valeur de filtre permettant d'identifier les transactions à purger. Si `filterOperator` est `equals` ou `not_equals`, vous pouvez utiliser une liste de valeurs séparées par des espaces pour indiquer plusieurs valeurs. Par exemple : `filterValue=101-120 filterValue=102-202`. Si plusieurs valeurs sont indiquées, les transactions provenant des comptes correspondant à toute combinaison d'opérateur de filtre et de valeur de filtre sont sélectionnées afin d'être purgées.
- `logFilename` (facultatif) est le nom d'un fichier journal permettant d'enregistrer des informations à propos de l'activité de commande. Si vous n'indiquez aucun nom de fichier, un fichier journal nommé `PurgeTransactions_`*`JOB_ID`* est généré automatiquement.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Note**
>
> Si `filterOperator` et `filterValue` ne sont pas indiqués, toutes les transactions mises en correspondance antérieures ou égales à `age` provenant de tous les comptes pour la valeur `matchType` indiquée sont purgées.

Exemples

- Purge des transactions mises en correspondance il y a `180` jours ou plus de type `cashrecon` :
  ```
  epmautomate purgeTMTransactions cashrecon 180 logFileName=tmlogs.log
  ```
- Purge des transactions mises en correspondance il y a `180` jours ou plus de type `cashrecon` pour le compte `101-120` ou `102-202` :
  ```
  epmautomate purgeTMTransactions cashrecon 180 filterOperator=equals
  filterValue=101-120 FilterValue=102-202
  ```
- Purge des transactions mises en correspondance il y a `180` jours ou plus de type `cashrecon` pour tout compte contenant la chaîne `11` :
  ```
  epmautomate purgeTMTransactions cashrecon 180 filterOperator=contains
  filterValue=11
  ```

recomputeOwnershipData

Recalcule les données de participation.

Le recalcul des données de participation dans Financial Consolidation and Close est requis dans les situations suivantes :

- après l'ajout ou la suppression de règles de remplacement pour des comptes de gestion de la participation ;
- après la modification de paramètres de plage des méthodes de consolidation ;
- après une actualisation de base de données, que la structure d'entité ait été ou non modifiée.

Le recalcul des données de participation dans Tax Reporting est requis après chaque actualisation de base données même si la structure d'entité n'a pas été modifiée.

Champ d'application

Financial Consolidation and Close et Tax Reporting.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur

Syntaxe

`epmautomate recomputeOwnershipData` *`Scenario Year Period`*, où :

- `Scenario` est le nom du scénario à recalculer.
- `Year` est l'année à recalculer.
- `Period` est la première période de l'année à recalculer.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

La période sélectionnée et toutes les périodes suivantes sont recalculées.

> **Remarque**
>
> Un PDV nécessitant un recalcul ne peut être consolidé qu'une fois les données de participation recalculées.

Exemple

```
epmautomate recomputeOwnershipData FCCS_total_Actual FY19 Jan
```

recreate

Restaure un environnement vers un état propre en le recréant.

Pour recréer l'environnement, procédez comme suit :

- Nettoyez un environnement avant d'importer un instantané complet.
- Modifiez le processus métier pouvant être déployé dans un environnement.

> **Attention**
>
> - Cette commande supprime de l'environnement l'application existante et, éventuellement, tous les artefacts définis par l'utilisateur. Elle permet en outre de recréer la base de données et d'enlever toutes les données existantes. Une fois le service recréé, vous pouvez créer un processus métier ou en importer un à l'aide de Migration ou d'EPM Automate.
> - Cette commande supprime l'historique de migration. Par conséquent, le rapport sur le statut de la migration disponible dans Migration ne contiendra plus d'informations d'historique.
> - Avant d'exécuter cette commande, effectuez une sauvegarde complète de l'environnement. Vous pouvez créer un instantané de sauvegarde en exécutant la commande runDailyMaintenance.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate recreate [-f] [removeAll=true|false] [TempServiceType=Service_type]`, où :

- `-f` force le démarrage du processus de recréation sans attendre la confirmation de l'utilisateur. Si vous n'utilisez pas l'option `-f`, la commande vous invite à confirmer votre action.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `removeAll` (facultatif) enlève tous les instantanés ainsi que le contenu des boîtes de réception (fichiers chargés) et d'envoi (fichiers exportés à partir de l'environnement). La valeur par défaut est `false`, qui conserve les instantanés et le contenu des boîtes de réception et d'envoi.
- `TempServiceType` (facultatif) convertit un environnement en un environnement d'un autre service.
  Le processus métier que vous pouvez déployer dans un environnement est régi par votre type d'abonnement. Par exemple, si vous disposez d'un abonnement EPM Standard Cloud Service, vous ne pouvez pas créer d'application de forme libre après avoir converti un environnement Account Reconciliation en un environnement Planning. Si vous disposez d'un abonnement EPM Enterprise Cloud Service, vous pouvez créer n'importe quel processus métier dans votre environnement après avoir modifié le type de service de façon appropriée. Reportez-vous à la section A propos des nouveaux services Cloud EPM du *Guide de mise en route pour les administrateurs*

  Le comportement de ce paramètre dépend de votre abonnement.

  – **Abonnements autres qu'EPM Standard Cloud Service et EPM Enterprise Cloud Service :**
    Vous pouvez utiliser l'option `TempServiceType` pour convertir temporairement un environnement Planning, Enterprise Planning, Tax Reporting ou Financial Consolidation and Close en environnement Account Reconciliation, Cloud EDM ou Profitability and Cost Management. Par exemple, si vous avez acheté un environnement Planning, vous pouvez le convertir en un environnement Account Reconciliation en exécutant la commande suivante :

    ```
    epmautomate recreate -f removeAll=true TempServiceType=ARCS
    ```

    Après avoir converti l'environnement en environnement Account Reconciliation, vous pouvez le convertir en environnement Cloud EDM ou Profitability and Cost Management en utilisant la valeur appropriée pour l'option `TempServiceType`. Par exemple, pour le convertir en un environnement Profitability and Cost Management, exécutez la commande suivante :

    ```
    epmautomate recreate -f removeAll=true TempServiceType=PCMCS
    ```

    Pour revenir au type de service d'origine, exécutez la commande suivante :

    ```
    epmautomate recreate -f
    ```

    Profitability and Cost Management : vous pouvez convertir votre environnement Profitability and Cost Management en environnement Planning, Enterprise Planning ou Enterprise Profitability and Cost Management en exécutant la commande suivante :

    ```
    epmautomate recreate -f removeAll=true TempServiceType=PBCS
    ```

    Pour revenir à l'environnement Profitability and Cost Management d'origine, utilisez la commande suivante :

    ```
    epmautomate recreate -f TempServiceType=PCMCS
    ```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Remarque**
>
> Les environnements Profitability and Cost Management ne peuvent pas être convertis en environnements Account Reconciliation, Cloud EDM ou Narrative Reporting.

- **Abonnements EPM Standard Cloud Service et EPM Enterprise Cloud Service :** Vous pouvez utiliser l'option `TempServiceType` pour convertir un environnement Oracle Fusion Cloud Enterprise Performance Management en n'importe quel environnement pris en charge.

  Les abonnements EPM Enterprise Cloud Service utilisent une plate-forme Cloud EPM commune. Dans un premier temps, vous pouvez déployer tout processus métier Cloud EPM pris en charge.

  Pour passer d'un processus métier déployé à un autre, recréez l'environnement en indiquant le nouveau type de service pour l'environnement. Par exemple, si vous avez créé un processus métier Account Reconciliation, mais que vous souhaitez à présent créer un environnement Cloud EDM, exécutez la commande de recréation comme suit :

  ```
  epmautomate recreate -f removeAll=true TempServiceType=EDMCS
  ```

  Pour convertir un processus métier (par exemple, Account Reconciliation) vers Planning, Tax Reporting ou Financial Consolidation and Close, n'indiquez aucune valeur pour l'option `TempServiceType`. Par exemple, si vous avez créé un processus métier Account Reconciliation, mais que vous souhaitez à présent créer un environnement de modules Planning, exécutez la commande recreate comme suit :

  ```
  epmautomate recreate -f removeAll=true
  ```

Valeurs `TempServiceType` acceptables :

- `ARCS` convertit un environnement en un environnement Account Reconciliation.
- `EDMCS` convertit un environnement en environnement Cloud EDM.
- `EPRCS` convertit un environnement en un environnement Narrative Reporting.
- `PCMCS` convertit un environnement en un environnement Profitability and Cost Management.

Exemples

- Recréation de l'environnement actuel, restauration de celui-ci vers le type de service d'origine (si une commande recreate a été émise auparavant avec le paramètre `TempServiceType`) sans enlever les instantanés créés par l'utilisateur ni le contenu de la boîte de réception et de la boîte d'envoi :

  ```
  epmautomate recreate -f
  ```

- Recréation de l'environnement actuel, restauration de celui-ci vers le type de service d'origine (si une commande recreate a été émise auparavant avec le paramètre `TempServiceType`), et retrait du contenu de la boîte de réception et de la boîte d'envoi :
  ```
  epmautomate recreate -f removeAll=true
  ```
- Recréation de l'environnement actuel en tant qu'environnement Enterprise Data Management et retrait du contenu de la boîte de réception et de la boîte d'envoi, ainsi que des instantanés existants :
  ```
  epmautomate recreate -f removeAll=true TempServiceType=EDMCS
  ```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Recréation de l'environnement actuel EPM Enterprise Cloud Service Account Reconciliation vers un environnement Financial Consolidation and Close et retrait du contenu de la boîte de réception et de la boîte d'envoi, ainsi que des instantanés existants :
  `epmautomate recreate -f removeAll=true`

refreshCube

Actualise le cube de l'application. En général, vous actualisez le cube après avoir importé des métadonnées dans l'application.

Le temps requis pour effectuer une opération d'actualisation de cube dépend des modifications que vous avez apportées à la structure de l'application et de son incidence sur le cube. Par exemple, une actualisation après la mise à jour d'un membre de cube en mode stockage en bloc dispersé peut ne pas être très longue, tandis qu'une actualisation de cube après la mise à jour d'un membre de cube en mode stockage en bloc dense ou d'un membre de cube en mode stockage agrégé peut être très longue. Vous devez vous assurer que l'opération d'actualisation de cube prend fin avant que l'application soit sauvegardée lors de la prochaine fenêtre de maintenance.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate refreshCube [JOB_NAME]`, où `JOB_NAME` (facultatif) est le nom d'un job d'actualisation de base de données défini dans l'application.
Le statut de l'opération est reflété dans la console à partir de laquelle la commande est exécutée. Vous pouvez également visualiser le statut à partir de la page **Activité récente** de l'écran **Jobs** dans l'application.

Exemple

```
epmautomate refreshCube DaliyCubeRefresh
```

refreshReportPackageDatasource

Actualise les sources de données d'un package de rapports afin que Narrative Reporting mette à jour systématiquement tous les doclets associés aux sources de données.

Le temps nécessaire pour exécuter cette commande est directement lié au nombre de doclets concernés.

Les sources de données Narrative Reporting incluent Planning, FreeForm, Financial Consolidation and Close, Enterprise Profitability and Cost Management, Tax Reporting, Profitability and Cost Management, Oracle ERP Cloud et Oracle Essbase.

Champ d'application

Narrative Reporting

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate refreshReportPackageDatasource` *`REPORT_PACKAGE_NAME`* `[refreshableSources=`*`DATA_SOURCE_NAME`*`]`, où :

- *`REPORT_PACKAGE_NAME`* est l'emplacement (chemin) et le nom du package de rapports qui utilise la source de données à actualiser.
- `refreshableSources` (facultatif) représente la liste des sources de données, séparées par des virgules, utilisées pour le package de rapports. Si cette valeur n'est pas spécifiée, la commande actualise toutes les sources de données utilisées par le package de rapports.

Exemple

```
epmautomate refreshReportPackageDatasource "Samples/Report Packages/Sample Report Package - MS Word" refreshableSources="Sample Model, Sample Word Model"
```

removeUserFromGroups

Enlève l'appartenance d'un utilisateur aux groupes Contrôle d'accès identifiés dans un fichier CSV codé en ANSI ou UTF-8.

Le format de fichier est le suivant :

```
Group Name
Group1
Group2
```

> **Remarque**
>
> Ces groupes doivent exister dans Contrôle d'accès. Les valeurs de nom de groupe ne respectent pas la casse.

Un administrateur de service utilise la commande uploadFile pour charger le fichier vers un environnement.

Une fois l'exécution terminée, la commande imprime les informations relatives à chaque entrée en échec dans la console. Reportez-vous à ces informations afin de comprendre pourquoi l'exécution de la commande a échoué pour certaines entrées dans le fichier CSV.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Syntaxe

`epmautomate removeUserFromGroups FILE_NAME User_Login`, où :

- `FILE_NAME` est le nom d'un fichier CSV contenant les noms des groupes Contrôle d'accès desquels l'appartenance de l'utilisateur doit être enlevée.
- `User_Login` est l'ID de connexion d'un utilisateur Oracle Fusion Cloud Enterprise Performance Management dont l'appartenance à des groupes Contrôle d'accès doit être enlevée. Cet ID de connexion utilisateur doit exister dans le domaine d'identité qui gère l'environnement et doit être affecté à un rôle prédéfini. Cette valeur ne respecte pas la casse.

Exemple

```
epmautomate removeUserFromGroups groups.CSV jdoe@examle.com
```

removeUsers

Supprime les comptes identifiés dans un fichier CSV ANSI ou UTF-8 chargé vers l'environnement à partir d'un domaine d'identité.

Le format de fichier est le suivant :

```
User Login
jane.doe@example.com
jdoe@example.com
```

Un administrateur de service utilise la commande uploadFile pour charger le fichier vers l'environnement. Les valeurs de connexion utilisateur ne respectent pas la casse. Par exemple, la valeur `jane.doe@example.com` est considérée comme identique à `Jane.Doe@Example.com`, ou à toute autre variation de la casse.

> **Remarque**
>
> - Le fichier CSV ne doit pas inclure le compte de l'utilisateur qui exécute cette commande.
> - Les comptes utilisateur étant communs à tous les environnements de service pris en charge par un administrateur de domaine d'identité, la suppression d'un compte pour un environnement le supprime de tous les environnements qui partagent le même administrateur de domaine d'identité.

Une fois l'exécution terminée, la commande imprime les informations relatives à chaque entrée en échec dans la console. Reportez-vous à ces informations afin de comprendre pourquoi l'exécution de la commande a échoué pour certaines entrées dans le fichier CSV.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

Administrateur de domaine d'identité et tout rôle prédéfini

Syntaxe

`epmautomate removeUsers` *`FILE_NAME`*, où *`FILE_NAME`* est le nom du fichier CSV contenant les ID de connexion des utilisateurs à enlever du domaine d'identité.

Exemple

```
epmautomate removeUsers remove_users.CSV
```

removeUsersFromGroup

Enlève des utilisateurs répertoriés dans un fichier CSV ANSI ou UTF-8 d'un groupe géré dans Contrôle d'accès.

Le format de fichier est le suivant :

```
User Login
jdoe
john.doe@example.com
```

Les valeurs de connexion utilisateur ne respectent pas la casse. Par exemple, la valeur `jane.doe@example.com` est considérée comme identique à `Jane.Doe@Example.com`, ou à toute autre variation de la casse. Un administrateur de service utilise la commande uploadFile pour charger le fichier contenant les connexions utilisateur vers l'environnement.

Une fois l'exécution terminée, la commande imprime les informations relatives à chaque entrée en échec dans la console. Reportez-vous à ces informations afin de comprendre pourquoi l'exécution de la commande a échoué pour certaines entrées dans le fichier CSV.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer

Syntaxe

`epmautomate removeUsersFromGroup` *`FILE_NAME GROUP_NAME`*, où :

- *`FILE_NAME`* correspond au nom d'un fichier CSV contenant les noms de connexion des utilisateurs à enlever d'un groupe géré dans Contrôle d'accès.
- *`GROUP_NAME`* correspond au nom du groupe de contrôle d'accès duquel des utilisateurs doivent être enlevés. Cette valeur ne respecte pas la casse.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Remarque**
>
> L'utilisateur est enlevé d'un groupe uniquement si les deux conditions suivantes sont remplies :
>
> - Les ID de connexion utilisateur inclus dans le fichier existent dans le domaine d'identité qui gère l'environnement.
> - Un rôle prédéfini a été affecté à l'utilisateur dans le domaine d'identité.

Exemple

```
epmautomate removeUsersFromGroup user_file.CSV example_group
```

removeUsersFromTeam

Enlève d'une équipe les utilisateurs Oracle Fusion Cloud Enterprise Performance Management répertoriés dans un fichier CSV.

Si un utilisateur répertorié dans le fichier CSV n'est pas un membre de l'équipe, cette commande l'ignore. Les valeurs figurant dans ce fichier ne respectent pas la casse. Le format du fichier CSV est le suivant :

```
User Login
jdoe
jane.doe@example.com
```

Un administrateur de service utilise la commande uploadFile pour charger le fichier CSV vers l'environnement.

Champ d'application

Financial Consolidation and Close, Tax Reporting et Account Reconciliation.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Equipes - Gérer
- Tout rôle prédéfini et rôle d'application Utilisateurs - Gérer

Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmautomate removeUsersFromTeam` *`FILE`*`.CSV` *`TEAM_NAME`*, où :

- `FILE` identifie un fichier CSV au format UTF-8 contenant la liste des ID de connexion des utilisateurs à enlever de l'équipe.
- `TEAM_NAME` identifie un nom d'équipe tel qu'il est défini dans Contrôle d'accès. Cette valeur ne respecte pas la casse.

Exemple

```
epmautomate removeUsersFromTeam example_users.csv example_team
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

renameSnapshot

Renomme un instantané chargé ou créé dans un environnement.

Si cette commande est exécutée pour renommer un instantané en cours de génération ou d'archivage, vous obtiendrez l'un des messages d'erreur suivants :

- Erreur `Fichier introuvable` ou `Fichier non valide` si l'instantané est en cours de génération
- `Le processus d'archivage est en cours. Impossible de renommer ou de supprimer X` si l'instantané est en cours d'archivage

Ne renommez pas l'instantané de maintenance d'un environnement. Pour conserver une sauvegarde de l'instantané de maintenance, téléchargez l'instantané `Artifact Snapshot` à partir de l'environnement vers un ordinateur local, puis renommez-le selon vos besoins. Reportez-vous à la section Présentation de l'instantané de maintenance du guide *Guide de mise en route pour les administrateurs*.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Syntaxe

`epmautomate renameSnapshot` *`SNAPSHOT_NAME NEW_SNAPSHOT_NAME`*, où :

- *`SNAPSHOT_NAME`* est le nom d'un instantané existant. Cette valeur ne doit pas contenir les caractères spéciaux suivants : espace, \ (barre oblique inverse), / (barre oblique), * (astérisque), ? (point d'interrogation), " (guillemet), < (inférieur à) et > (supérieur à).
- *`NEW_SNAPSHOT_NAME`* est le nom unique à affecter à l'instantané.

Exemple

```
epmautomate renameSnapshot "Example Snapshot" Example_Snapshot_18_09_25
```

replay

Réexécute le chargement d'Oracle Smart View for Office, de l'API REST ou d'EPM Automate dans un environnement afin de permettre le test des performances dans des conditions de charge élevée, pour vérifier que l'expérience utilisateur est acceptable lorsque le service est soumis à la charge indiquée.

Vous devez créer un fichier de réexécution qui identifie les activités à exécuter sur le service. Reportez-vous à la section Préparation de l'exécution de la commande replay pour obtenir des informations détaillées sur la méthode de création du fichier de réexécution.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Syntaxe

`epmautomate replay` *`REPLAY_FILE_NAME`*`.csv [duration=`*`N`*`] [trace=true] [lagTime=`*`t`*`] [encrypt=true|false]`, où :

- *`REPLAY_FILE_NAME`* est un fichier CSV qui stocke les activités à exécuter sur l'environnement.
- `Duration` (facultatif) indique le nombre de minutes pendant lesquelles les activités sont exécutées dans l'environnement.
  Les activités dans le fichier HAR sont exécutées une seule fois si cette valeur n'est pas définie. Si les activités dans le fichier HAR sont terminées pendant la période indiquée par ce paramètre, la commande réexécute le fichier HAR jusqu'à ce que les activités soient terminées. Par exemple, vous avez défini `duration=10` pour réexécuter un fichier HAR dont l'exécution dure trois minutes. Dans ce scénario, la commande de réexécution exécute les activités du fichier HAR quatre fois (soit une durée de 12 minutes) jusqu'à ce que la quatrième itération soit terminée.
- `trace=true` est un paramètre facultatif qui indique à la commande de créer des fichiers de trace au format XML.
  Si ce paramètre facultatif est indiqué, la commande crée un dossier pour chaque fichier HAR inclus dans le fichier CSV de réexécution et y stocke tous les fichiers de trace associés. Pour chaque activité dans le fichier HAR, la commande génère un fichier de trace qui contient une réponse Smart View. Les fichiers de trace sont nommés `trace-`*`N`*`.xml` ; par exemple, `trace-1.xml` où `N` est un compteur qui débute à 1.

  Les dossiers qui stockent les fichiers de trace sont créés dans le répertoire à partir duquel EPM Automate est exécuté. La commande emploie une combinaison du temps système en cours de l'environnement et du nom du fichier HAR au format `YYYY_MM_DD_HH_MM_SS_`*`HAR_FILE_NAME`* pour nommer les dossiers. Par exemple, si le nom du fichier HAR est `forecast1.har`, le nom du dossier peut être `2016_06_08_10_21_42_forecast1`.
- `[lagTime=`*`t`*`]` (facultatif) indique le nombre de secondes d'attente de la commande entre l'exécution de chaque fichier HAR inclus dans le fichier de réexécution. La valeur par défaut est 5 secondes.
  La commande affiche un message d'erreur si vous indiquez une valeur inférieure à cinq secondes. Les nombres négatifs (par exemple, -1) et les fractions (par exemple, 1/2) ne sont pas acceptés comme valeurs du paramètre. Les valeurs décimales sont prises en charge.

  Après le lancement de l'exécution du premier fichier HAR, la commande attend le nombre de secondes indiqué par ce paramètre pour lancer le traitement du fichier HAR suivant.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Les activités utilisateur n'étant généralement pas lancées simultanément, la définition de ce paramètre contribue à créer une simulation plus réaliste du chargement dans un environnement.

Par exemple, supposons que vous voulez simuler le chargement de 1 000 utilisateurs qui se connectent à un environnement à l'heure de pointe pour effectuer des activités. Vous pouvez créer des fichiers HAR pour simuler ces sessions, puis exécuter cette commande avec un décalage de 6 secondes pour répliquer le chargement imposé à l'environnement.

- `encrypt=true|false` (facultatif) indique si tous les mots de passe inclus dans le fichier de réexécution doivent être cryptés. La valeur par défaut est `true`. Une clé de cryptage aléatoire est utilisée pour crypter le mot de passe.

Reportez-vous à la section Exemple de session de réexécution pour connaître les étapes détaillées d'exécution de cette commande.

Exemple

```
epmautomate replay forecast1.CSV duration=60 lagTime=5.6
```

resetService

Redémarre l'environnement. Avant de redémarrer l'environnement, vous pouvez le régler automatiquement afin que les caches d'index Oracle Essbase pour les cubes avec option "block storage" (BSO) soient optimisés pour l'application.

Par défaut, les environnements sont redémarrés juste après la fin de la maintenance quotidienne. Le réglage automatique de l'environnement est important, par exemple après l'import d'un instantané dans un environnement. Utilisez cette commande uniquement lorsque vous remarquez une dégradation notable des performances ou si vous recevez des messages d'erreur indiquant que l'environnement est inutilisable. Le redémarrage d'un environnement n'a aucune incidence sur les personnalisations d'application (par exemple, les différences locales, les paramètres liés au thème et à la devise, etc.). Le redémarrage peut prendre jusqu'à 15 minutes.

Avant d'utiliser cette commande, assurez-vous qu'aucune règle métier n'est en cours d'exécution dans l'environnement.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate resetService "comment" [AutoTune=true|false] [-f]` où :

- `Comment` est une description du problème qui vous a contraint à réinitialiser l'environnement. Les commentaires doivent être placés entre guillemets.
- `AutoTune` (facultatif) indique si l'environnement doit être réglé automatiquement afin d'optimiser les caches Essbase des cubes BSO de l'application. La valeur par défaut est `false`.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Servez-vous de ce paramètre uniquement dans les environnements utilisant des cubes BSO Essbase : Planning (y compris les modules Planning), Financial Consolidation and Close et Tax Reporting.

- `-f` (facultatif) indique que vous souhaitez forcer le redémarrage de l'environnement sans interaction utilisateur supplémentaire. Si vous n'utilisez pas cette option, la commande vous invite à confirmer votre action. Cette option est utile si vous planifiez un script qui utilise cette commande.

Exemples

- `epmautomate resetService "Les utilisateurs font face à des connexions lentes ; redémarrage forcé de l'environnement" -f`
- `epmautomate resetService "Les utilisateurs font face à des connexions trop lentes"`
- `epmautomate resetService "Optimizing the Essbase cache" AutoTune=true`

restoreBackup

Copie un instantané de sauvegarde disponible pour qu'il puisse être importé dans l'environnement.

Utilisez la commande listBackups pour déterminer si la sauvegarde à restaurer est disponible. La restauration en libre-service d'un instantané vers l'environnement vous fait gagner du temps de traitement. Après avoir restauré l'instantané, utilisez la commande importSnapshot pour l'importer dans l'environnement.

> **Note**
>
> **Narrative Reporting uniquement :** pour importer l'instantané restauré, procédez comme suit.
>
> 1. Employez la commande downloadFile pour télécharger l'instantané restauré sur un ordinateur local.
> 2. Renommez l'instantané téléchargé `EPRCS_Backup`.
> 3. Employez la commande uploadFile pour charger l'instantané renommé (`EPRCS_Backup`) vers l'environnement. Veillez à indiquer l'emplacement de chargement `to_be_imported` dans la commande.
> 4. Dans Narrative Reporting, ouvrez la carte **Maintenance quotidienne**.
>   a. Sélectionnez **Utiliser un instantané de sauvegarde chargé**.
>   b. Cliquez sur **Planifier la restauration** pour importer l'instantané lors de la prochaine maintenance quotidienne.
>   c. Cliquez sur **Oui**.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Syntaxe

`epmAutomate restoreBackup` *`SNAPSHOT_NAME`* `[targetName=`*`TARGET_SNAPSHOT_NAME`*`]`, où :

- *`SNAPSHOT_NAME`* est le nom d'un instantané de sauvegarde disponible dans l'environnement tel que répertorié par la commande listBackups.
- `targetName` (facultatif) est le nom de l'instantané de sauvegarde, sans extension, dans l'environnement cible. Si vous n'indiquez pas cette valeur, l'instantané de sauvegarde est restauré vers l'environnement cible à l'aide de la valeur *`SNAPSHOT_NAME`*, mais avec un trait de soulignement (_) remplaçant la barre oblique (/). Par exemple, si *`SNAPSHOT_NAME`* a la valeur `2022-05-14T00:08:56/Artifact_Snapshot.zip`, la valeur `targetName` sera `2022-05-14T00:08:56_Artifact_Snapshot.zip`.

Exemples

- Services autres que Narrative Reporting :
  ```
  epmAutomate restoreBackup 2022-05-14T00:08:56/Artifact_Snapshot.zip
  targetName=example_Artifact_Snapshot
  ```
- Narrative Reporting uniquement :
  ```
  epmAutomate restoreBackup 2022-02-16T21:00:02/EPRCS_Backup
  targetName=Example_EPRCS_Backup
  ```

restructureCube

Effectue une restructuration complète d'un cube en mode "block storage" pour éliminer ou réduire la fragmentation. La restructuration permet également d'enlever les blocs vides. Elle ne transmet aucune des modifications de l'application au cube.

> **Remarque**
>
> Avant d'exécuter cette commande, assurez-vous que personne n'utilise l'application.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate restructureCube` *`CUBE_NAME`*, où *`CUBE_NAME`* est le nom du cube tel qu'il apparaît dans l'application.

Exemple

```
epmautomate restructureCube Plan1
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

roleAssignmentAuditReport

Crée un rapport d'audit qui répertorie les modifications apportées aux affectations de rôle prédéfini et d'application au cours d'une période donnée correspondant à la période de conservation des données d'audit indiquée pour l'environnement. La période de conservation par défaut est de 30 jours. Vous pouvez prolonger la conservation des données d'audit de rôle prédéfini à 90 jours au maximum en modifiant le paramètre **Période de conservation des données d'audit (jours)** dans la console de gestion des identités Oracle Cloud. Les données d'audit de rôle d'application sont disponibles pendant 120 jours au maximum. Pour conserver les données d'audit pendant plus longtemps, téléchargez et archivez régulièrement ce rapport, ainsi que le Rapport sur les connexions non valides.

Le rapport d'audit sur l'affectation de rôle répertorie le nom de connexion utilisateur, le nom de groupe IDCS ou le nom de groupe EPM pour lequel une modification de rôle (dans la colonne Action) a été apportée. Il inclut également le rôle prédéfini ou d'application affecté ou celui dont l'affectation a été annulée, l'utilisateur ayant effectué la modification de rôle (colonne Effectué par) ainsi que l'horodatage (UTC) au format 24 heures de l'action. La colonne Type indique ce que la colonne Nom représente. Elle peut contenir l'une de ces trois valeurs : Utilisateur (si la colonne Nom identifie le nom de connexion d'un utilisateur), Groupe IDCS (si la colonne Nom affiche un nom de groupe IDCS) ou Groupe EPM (si la colonne Nom répertorie un nom de groupe EPM).

	A	B	C	D	E	F
1	**Name**	**Type**	**Role**	**Action**	**Performed By**	**Date and Time**
2	IDCS-Group1	IDCS Group	Power User	Assign	serviceadmin@example.com	December 11, 2024 00:47:10 UTC
3	jane.doe@example.com	User	Power User	Assign	serviceadmin@example.com	December 11, 2024 00:47:27 UTC
4	epmIDCSGroup	User	Service Administrator	Assign	serviceadmin@example.com	December 11, 2024 00:48:47 UTC
5	john.doe@example.com	User	Power User	Assign	serviceadmin@example.com	December 11, 2024 00:59:06 UTC
6	john.doe@example.com	User	Ad Hoc - Create	Assign	serviceadmin@example.com	December 11, 2024 01:08:37 UTC
7	john.doe@example.com	User	Access Control - Manage	Unassign	serviceadmin@example.com	December 11, 2024 01:10:30 UTC
8	john.doe@example.com	User	Ad Hoc - Create	Unassign	serviceadmin@example.com	January 07, 2025 03:28:56 UTC

Les informations relatives aux utilisateurs supprimés, auxquels des rôles prédéfinis avaient été précédemment affectés dans l'environnement, sont répertoriées avec le nom d'affichage (prénom et nom) de l'utilisateur dans la colonne Nom. Dans ce cas, la colonne Rôle indique le rôle prédéfini qui avait été affecté à l'utilisateur avant la suppression du compte de ce dernier. Cette modification ne s'applique pas aux rôles d'application affectés à l'utilisateur supprimé, le cas échéant. Ces affectations apparaissent avec le nom de connexion de l'utilisateur. A titre d'exemple, reportez-vous aux informations qui figurent dans l'encadré rouge de l'illustration suivante.

	A	B	C	D	E	F
1	**Name**	**Type**	**Role**	**Action**	**Performed By**	**Date and Time**
2	Jane Doe	User	User	Assign	admin@example.com	January 08, 2025 05:42:47 UTC
3	jane.doe@example.com	User	Run Integrations	Assign	admin@example.com	January 08, 2025 05:44:18 UTC
4	john.smith@example.com	User	Power User	Assign	admin@example.com	January 08, 2025 05:44:18 UTC
5	john.smith@example.com	User	User	Unassign	admin@example.com	January 08, 2025 05:44:18 UTC
6	john.smith@example.com	User	Access Control - View	Assign	admin@example.com	January 08, 2025 05:44:18 UTC
7	epmIDCSGroup	IDCS Group	Viewer	Assign	admin@example.com	January 08, 2025 05:44:22 UTC
8	epmGroup	EPM Group	Ad Hoc - Create	Assign	admin@example.com	January 08, 2025 05:44:22 UTC

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Strategic Workforce Planning et Sales Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle Administrateur de domaine d'identité
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Afficher

Syntaxe

`epmAutomate roleAssignmentAuditReport FROM_DATE TO_DATE FILE_NAME.CSV`, où :

- *FROM_DATE* indique la date de début (au format `YYYY-MM-DD`) de la période pour laquelle le rapport doit être généré.
- *TO_DATE* indique la date de fin (au format `YYYY-MM-DD`) de la période pour laquelle le rapport doit être généré.
- *FILE_NAME* est le nom du fichier CSV pour le rapport. Vous pouvez télécharger ce rapport généré à l'aide de la commande downloadFile.
- **Note**

  Oracle Fusion Cloud Enterprise Performance Management s'assure que seule une plage de dates valide est utilisée lors de la génération du rapport. Les validations suivantes sont effectuées pour les dates *de début* et *de fin* :

  – La date *FROM_DATE* ne peut pas être antérieure à la période de conservation maximale autorisée à compter de la date actuelle. Cette date doit être comprise dans la plage de valeurs du paramètre **Période de conservation des données d'audit (jours)** de la console de gestion des identités Oracle Cloud.

  – La date *TO_DATE* ne peut pas être postérieure à la période de conservation maximale à compter de la date *FROM_DATE*.

  – La date *TO_DATE* ne peut pas être antérieure à la date *FROM_DATE*.

Exemple

```
epmAutomate roleAssignmentAuditReport 2024-12-11 2025-01-09 RoleAuditReport.CSV
```

roleAssignmentReport

Génère un rapport sur l'affectation de rôle (fichier CSV).

Par défaut, ce rapport répertorie les rôles prédéfinis (par exemple, Administrateur de service) et les rôles d'application (par exemple, Cédant de propriété des approbations, Superviseur des approbations, Administrateur des approbations et Concepteur du processus des approbations, qui sont des rôles d'application Planning) affectés aux utilisateurs. Ce rapport peut éventuellement aussi être généré pour répertorier les administrateurs de domaine d'identité de l'environnement. Ce rapport correspond à la version CSV du rapport d'affectation de rôle généré à partir de Contrôle d'accès.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

	A	B	C	D	E	F
1	User Login	First Name	Last Name	Email	Role	Granted through Group
2	Jdoe	John	Doe	jdoe@example.com	Planner	
3	jdoe	John	Doe	jdoe@example.com	Power User	
4	Jdoe	John	Doe	jdoe@example.com	Service Administrator	
5	jdoe	John	Doe	jdoe@example.com	Viewer	
6	Jdoe	John	Doe	jdoe@example.com	Mass Allocation	example->Power User
7	jdoe	John	Doe	jdoe@example.com	Run Integration	
8	jane.doe@example.com	Jane	Doe	jane.doe@example.com	Planner	
9	jane.doe@example.com	Jane	Doe	jane.doe@example.com	Power User	
10	jane.doe@example.com	Jane	Doe	jane.doe@example.com	Viewer	
11	jane.doe@example.com	Jane	Doe	jane.doe@example.com	Mass Allocation	example

La génération de ce rapport actualise les informations sur l'utilisateur et sur les rôles disponibles dans Contrôle d'accès.

Oracle Fusion Cloud Enterprise Performance Management considère les utilisateurs désactivés comme des utilisateurs auxquels aucun rôle prédéfini n'a été affecté, même si ces utilisateurs avaient un rôle prédéfini au moment de leur désactivation. Les informations sur les utilisateurs désactivés ne sont pas incluses dans ce rapport.

> **Remarque**
>
> Cette commande génère un rapport équivalent à celui créé à l'aide de la commande provisionReport.

Vous pouvez télécharger ce rapport à l'aide de la commande downloadFile.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Afficher

Syntaxe

`epmautomate roleAssignmentReport` *`REPORT_NAME`*`.CSV [userType=IDAdmins|serviceUsers]`, où :

- *`REPORT_NAME`* est le nom du rapport.
- `userType` (facultatif) identifie le type des utilisateurs dont inclure les informations dans le rapport. La valeur par défaut est `serviceUsers`. Les valeurs valides sont les suivantes :
  - La valeur `serviceUsers` crée un rapport qui contient des informations sur tous les utilisateurs fonctionnels (n'inclut pas les administrateurs de domaine d'identité s'ils ne disposent pas d'un rôle prédéfini qui leur donne accès à l'application).
  - `IDAdmins` crée un rapport qui répertorie uniquement les utilisateurs affectés au rôle Administrateur de domaine d'identité.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exemples

- Générez un rapport qui répertorie uniquement les utilisateurs fonctionnels :
  - `epmautomate roleAssignmentReport myReport.CSV`
  - `epmautomate roleAssignmentReport myReport.CSV userType=serviceUsers`
- Générez un rapport répertoriant uniquement les administrateurs de domaine d'identité : `epmautomate roleAssignmentReport myReport.CSV userType=IDAdmins`

runAutoAlert

Exécute le job d'alerte automatique pour un type de correspondance. Le job traite les règles d'alerte, puis crée automatiquement des alertes pour les transactions sans correspondance, le cas échéant. Vous pouvez surveiller le statut du job d'alerte automatique dans l'onglet **Historique des jobs** dans Account Reconciliation.

Champ d'application

Account Reconciliation

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmAutomate runAutoAlert` *`MATCH_TYPE`*, où *`MATCH_TYPE`* est l'identificateur d'un type de correspondance défini dans Account Reconciliation.

Exemple

```
epmautomate runAutoAlert INTERCOMPANY
```

runAutomatch

Exécute le processus de correspondance automatique pour mettre en correspondance les transactions à l'aide des règles définies par un administrateur de service.

> **Remarque**
>
> Exécutez cette commande après avoir importé les données de transactions dans le module Correspondance de transactions à l'aide de la commande importTmPremappedTransactions ou runDataRule.

Vous pouvez surveiller le statut du processus de correspondance automatique dans l'onglet **Historique des jobs** dans Account Reconciliation.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Champ d'application

Account Reconciliation

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmautomate runAutomatch RECONCILIATION_TYPE` où `RECONCILIATION_TYPE` est un type de correspondance défini dans Account Reconciliation.

Exemple

```
epmautomate runAutomatch INTERCOMPANY
```

runBatch

Exécute un lot de gestion des données.

Si le mode d'exécution par lots dans la gestion des données est défini sur Série, le contrôle est renvoyé lorsque tous les jobs du lot sont terminés. S'il est défini sur Parallèle, le contrôle est renvoyé lorsque tous les jobs du lot sont soumis en vue de leur exécution.

> **Remarque**
>
> Cette commande ne peut pas être utilisée pour exécuter une intégration de chargement de données direct à partir de sources de données dans Oracle Fusion Cloud Enterprise Performance Management. Utilisez l'agent d'intégration EPM pour intégrer des chargements de données directs. Pour obtenir des informations détaillées, reportez-vous à la section Exécution d'un chargement des données direct à l'aide de l'agent d'intégration EPM du guide *Administration de l'intégration des données*.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Super utilisateur

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Syntaxe

`epmautomate runBatch` *`BATCH_NAME`* où *`BATCH_NAME`* est le nom du lot défini dans Data Management.

Exemples

```
epmautomate runBatch Accounting_batch
```

runBusinessRule

Lance une règle métier.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Super utilisateur (si l'accès en lancement à la règle est accordé)

Syntaxe

`epmautomate runBusinessRule` *`RULE_NAME`* `[`*`PARAMETER=VALUE`*`]` où :

- *`RULE_NAME`* est le nom d'une règle métier exactement tel qu'il est défini dans l'environnement.
- *`PARAMETER=VALUE`* indique des paramètres d'exécution facultatifs et leurs valeurs requises pour exécuter la règle métier.

> **Remarque**
>
> – Cette commande peut exécuter une seule règle métier. Pour exécuter un ensemble de règles, utilisez la commande runRuleSet.
>
> – La règle est exécutée sur le type de plan vers lequel elle a été déployée.
>
> – Les valeurs par défaut sont utilisées si vous n'indiquez aucune valeur pour les paramètres d'exécution. La commande ignore les invites d'exécution qui ne sont pas des correspondances exactes avec celles définies dans la règle.
>
> – Utilisez la paire *`PARAMETER=VALUE`* pour indiquer autant d'invites d'exécution que la règle métier l'exige. L'exemple suivant utilise deux invites d'exécution (`Period` et `Entity`) et leurs valeurs (`Q1` et `USA`).
> Pour plus d'informations sur la saisie de plusieurs valeurs pour un paramètre, reportez-vous à la section Spécification de plusieurs valeurs pour un paramètre.

Exemple

```
epmautomate runBusinessRule RollupUSSales Period=Q1 Entity=USA
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

runCalc

Exécute des calculs dans une application.

Cette commande vous permet d'exécuter des calculs en utilisant des règles dans un PDV de modèle par rapport à des données qui se trouvent dans un autre PDV de données, sans copier les règles entre les PDV.

Champ d'application

Profitability and Cost Management

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate runCalc` *`APPLICATION_NAME POV_NAME [DATA_POV_NAME]`* `PARAMETER=`*`VALUE`* `[comment="comment"] stringDelimiter=`*`"DELIMITER"`*, où :

- *`APPLICATION_NAME`* est le nom de l'application Profitability and Cost Management contenant le PDV à calculer.
- *`POV_NAME`* est le nom du PDV de modèle à calculer.
- *`DATA_POV_NAME`* (facultatif) est le nom du PDV de données à calculer à l'aide des règles du PDV de modèle.

  Si *`DATA_POV_NAME`* n'est pas spécifié, par défaut, la valeur *`POV_NAME`* sera utilisée.

  Vous pouvez utiliser uniquement `exeType=ALL_RULES` si vous spécifiez *`DATA_POV_NAME`*.
- `PARAMETER=`*`VALUE`* indique les paramètres d'exécution et leurs valeurs pour exécuter le calcul. Indiquez autant de paires paramètre-valeur que le processus l'exige. Paramètres valides et leurs valeurs :
  - `exeType=ALL_RULES|RULESET_SUBSET|SINGLE_RULE` identifie le type d'exécution de règle. Ce paramètre est obligatoire.
    Selon la valeur définie pour `exeType`, les paramètres suivants peuvent être spécifiés :
    - Si vous indiquez `exeType=ALL_RULES`, n'incluez pas de paramètres de règle unique ou de sous-ensemble de règles tels que `subsetStart`, `subsetEnd`, `ruleSetName` et `ruleName`. Vous devez utiliser ce type `exeType` si vous définissez le paramètre *`DATA_POV_NAME`*.
    - Si vous indiquez `exeType=SINGLE_RULE`, spécifiez les valeurs de `ruleSetName` et `ruleName` uniquement.
    - Si vous indiquez `exeType=RULESET_SUBSET`, spécifiez les valeurs pour `subsetStart` et `subsetEnd`.
  - `subsetStart` indique le numéro de séquence de la première règle dans l'ensemble de règles à exécuter
  - `subsetEnd` indique le numéro de séquence de la dernière règle dans l'ensemble de règles à exécuter
  - `ruleSetName` identifie l'ensemble de règles qui contient les calculs à exécuter
  - `ruleName` nom de la règle à exécuter (exécution d'une règle unique)

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

  - `isClearCalculated=true|false` indique si les calculs existants doivent être effacés
  - `isExecuteCalculations=true|false` indique si des calculs doivent être exécutés
  - Pour `isRunNow=true|false`, définissez cette valeur sur `true` pour exécuter le processus immédiatement.
  - Pour `optimizeReporting=true|false`, définissez cette valeur facultative sur `false` si les calculs doivent être exécutés sans optimisation à des fins de reporting. La valeur par défaut est `true`.
    Meilleures pratiques :
    * Définissez `optimizeReporting=false` uniquement lorsque cela est nécessaire pour gagner du temps de traitement, par exemple, lors de l'exécution d'une règle unique ou d'une série séquentielle de PDV
    * En cas d'exécution de plusieurs jobs de calcul simultanés, définissez `optimizeReporting=true` pour tous les jobs. Le dernier job effectué réalise l'agrégation, ce qui évite un traitement redondant et le ralentissement de l'exécution des jobs.

> **Remarque**
>
> Les valeurs de paramètre (`true` ou `false`) doivent être indiquées entièrement en minuscules.

- `comment` est un commentaire facultatif placé entre guillemets doubles.
- `stringDelimiter` est le séparateur utilisé dans les valeurs de PDV. Le séparateur doit être placé entre guillemets doubles.

Exemple

```
epmautomate runCalc BksML12 2012_Jan_Actual Jan-2016 isClearCalculated=true
isExecuteCalculations=true isRunNow=true subsetStart=10 subsetEnd=20
ruleSetName="Utilities Expense Adjustment" ruleName="Occupancy Expense
Allocations" exeType="ALL_RULES" comment="Test calculation" stringDelimiter="_"
```

runComplianceReport

Génère un rapport qui est défini dans le module Conformité de rapprochement.

Reportez-vous aux sources d'information suivantes dans le guide *Administration d'Account Reconciliation* :

- Pour obtenir des instructions concernant la définition des rapports : Utilisation des rapports.
- Pour obtenir la liste des rapports prédéfinis du module Conformité de rapprochement et les paramètres permettant de les générer : Génération de rapports prédéfinis dans le module Conformité de rapprochement.

Champ d'application

Account Reconciliation

Rôles requis

- Administrateur de service

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Super utilisateur
- Utilisateur
- Visualiseur

Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmautomate runComplianceReport` *`FILE_NAME GROUP_NAME REPORT_NAME REPORT_FORMAT`* `[Param=value]` où :

- `FILE_NAME` est le nom de fichier unique du rapport à générer. Si un rapport portant le même nom existe déjà sur le serveur, il sera remplacé. Employez la commande downloadFile pour télécharger ce rapport sur un ordinateur local.
- `GROUP_NAME` est le nom du groupe auquel est associé le rapport.
- `REPORT_NAME` est le nom unique du rapport à générer.
- `REPORT_FORMAT` est l'un des formats de rapport suivants :
  - PDF
  - HTML (non pris en charge pour les graphiques)
  - XLSX (non pris en charge pour les graphiques)
  - CSV
  - CSV2

> **Remarque**
>
> *`REPORT_FORMAT`* `CSV` n'autorise pas le formatage des données à partir d'un modèle alors que `CSV2` le permet. La génération d'un rapport formaté `CSV2` prend plus de temps que celle d'une sortie `CSV`.

- `Param=value` (facultatif) identifie les paramètres requis pour générer le rapport. Par exemple, le rapport `Balance By Account Type` accepte deux paramètres : `Period` dont la valeur est `July 2017` et `Currency Bucket` dont la valeur est `Entered`. Vous devez indiquer ces paramètres comme suit : `"Period=July 2017" "Currency Bucket=Entered"`.

Exemple

```
epmautomate runComplianceReport "Example_File Name""Reconciliation Manager"
"Balance By Account Type" PDF "Period=July 2017" "Currency Bucket=Entered"
```

runDailyMaintenance

Démarre le processus de maintenance de service quotidien directement au lieu d'attendre la fenêtre de maintenance quotidienne programmée.

Cette commande vous permet de forcer la création d'un instantané de sauvegarde et de mettre à jour l'environnement. Avant d'exécuter cette commande, assurez-vous que personne n'utilise l'environnement. Le programme de maintenance quotidienne n'est pas affecté par cette commande. Utilisez cette commande pour éviter d'attendre la prochaine fenêtre de maintenance afin que les modifications apportées à l'environnement prennent effet, par exemple, après l'application d'un patch unique.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service, Utilisateur (Cloud EDM uniquement)

Syntaxe

`epmautomate runDailyMaintenance [skipNext=true|false] [-f]` où :

- `skipNext` (facultatif) indique si la prochaine occurrence du processus de maintenance quotidienne doit être ignorée. La valeur par défaut est false.
- `-f` (facultatif) indique si le démarrage du processus de maintenance doit être forcé sans la confirmation de l'utilisateur. Si vous n'utilisez pas l'option `-f`, la commande vous invite à confirmer votre action.

Exemples

- Pour forcer le démarrage d'une maintenance quotidienne hors cycle sans ignorer la prochaine maintenance planifiée : `epmautomate runDailyMaintenance -f`
- Pour forcer le démarrage d'une maintenance quotidienne hors cycle et ignorer la prochaine maintenance planifiée : `epmautomate runDailyMaintenance skipNext=true -f`
- Pour démarrer une maintenance quotidienne hors cycle et ignorer la prochaine maintenance planifiée : `epmautomate runDailyMaintenance skipNext=true`

runDataRule

Exécute une règle de chargement des données de la gestion des données en fonction de la période de début et de fin, ainsi que des options d'import ou d'export indiquées.

> **Remarque**
>
> Cette commande ne peut pas être utilisée pour exécuter une intégration de chargement de données direct à partir de sources de données dans Oracle Fusion Cloud Enterprise Performance Management. Utilisez l'agent d'intégration EPM pour intégrer des chargements de données directs. Pour obtenir des informations détaillées, reportez-vous à la section Exécution d'un chargement des données direct à l'aide de l'agent d'intégration EPM du guide *Administration de l'intégration des données*.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Super utilisateur

Syntaxe

`epmautomate runDataRule` *`RULE_NAME START_PERIOD END_PERIOD IMPORT_MODE EXPORT_MODE [FILE_NAME]`*, où :

- *`RULE_NAME`* est le nom d'une règle de chargement des données dans la gestion des données. Vous devez placer le nom de la règle entre guillemets s'il contient un espace.
- *`START_PERIOD`* est la première période pour laquelle des données doivent être chargées. Ce nom de période doit être défini dans le mapping de période de la gestion des données.
- *`END_PERIOD`* est, pour un chargement des données multipériodes, la dernière période pour laquelle des données doivent être chargées. Pour le chargement de période unique, saisissez la même période que la période de début. Ce nom de période doit être défini dans le mapping de période de la gestion des données.
- *`IMPORT_MODE`* détermine comment les données sont importées dans la gestion des données.

  Les paramètres de mode d'import respectent la casse. Les valeurs acceptables sont les suivantes :

  - `APPEND` pour les ajouter aux données de PDV existantes dans la gestion des données.
  - `REPLACE` pour supprimer les données de PDV et les remplacer par celles du fichier.
  - `RECALCULATE` pour recalculer les données.
  - `NONE` pour ignorer l'import des données dans la table intermédiaire de gestion des données.
- *`EXPORT_MODE`* détermine comment les données sont exportées vers l'application.

  Les paramètres de mode d'export respectent la casse. Les valeurs acceptables sont les suivantes :

  - `STORE_DATA` pour fusionner les données de la table intermédiaire de gestion des données avec les données existantes. Utilisez toujours cette option d'export dans les jobs de gestion des données employés pour charger les métadonnées.
  - `ADD_DATA` pour ajouter les données de la table intermédiaire de gestion des données à l'application.
  - `SUBTRACT_DATA` pour soustraire les données de la table intermédiaire de gestion des données des données existantes.
  - `REPLACE_DATA` pour effacer les données de PDV et les remplacer par des données dans la table intermédiaire de gestion des données. Les données sont effacées pour Scénario, Version, Année, Période et Entité.
  - `NONE` pour ignorer l'export de données de la gestion des données vers l'application.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Remarque**
>
> Pour Financial Consolidation and Close, seuls les modes d'export suivants sont pris en charge :
>
> – `MERGE` pour fusionner les données de la table intermédiaire de gestion des données avec les données existantes.
>
> – `REPLACE` pour enlever les entrées de la table intermédiaire DM et les remplacer par celles du chargement des données.
>
> – `NONE` pour ignorer l'export de données de la gestion des données vers l'application
>
> Pour Oracle Fusion Cloud en tant que cible, seuls les modes d'export suivants sont pris en charge :
>
> – `MERGE` pour fusionner les données de la table intermédiaire de gestion des données avec les données existantes.
>
> – `NONE` pour ignorer l'export de données de la gestion des données vers l'application

- *`FILE_NAME`* est un nom de fichier facultatif. Si vous n'indiquez aucun nom de fichier, la commande importe les données contenues dans le fichier spécifié dans la règle de chargement des données. Ce fichier doit être disponible dans le dossier de boîte de réception ou dans un dossier qui s'y trouve.

  Lors du chargement de fichiers au format BAI (Bank Administration Institute) pour Account Reconciliation, n'indiquez aucune valeur pour ce paramètre. Vous devez toujours indiquer le nom du fichier pour le chargement de fichiers BAI dans la définition de la règle de données.

> **Remarque**
>
> Si un chemin est indiqué dans la règle de données, ne spécifiez pas le chemin du fichier dans la commande. Indiquez uniquement son nom. Si le chemin n'est pas indiqué dans la règle de données, spécifiez le chemin complet du fichier de données.

Exemples

- Import de plusieurs périodes :
  ```
  epmautomate runDataRule VisionActual Mar-15 Jun-15 REPLACE STORE_DATA inbox/
  Vision/GLActual.dat
  ```
- Import de période unique :

  ```
  epmautomate runDataRule "Vision Actual" Mar-15 Mar-15 REPLACE STORE_DATA
  inbox/Vision/GLActual.dat
  ```

runDMReport

Crée un rapport de gestion des données et le stocke dans le dossier `outbox/reports`.

Le rapport généré est nommé en fonction de l'ID du job de gestion des données qui génère le rapport et son format. Par exemple, si l'ID du job de rapport est 2112 et que le format de sortie du rapport que vous indiquez est PDF, le nom du rapport est `2112.pdf`. Le nom du rapport est

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

affiché dans la console une fois le rapport généré. Vous pouvez également identifier le nom du rapport dans l'onglet Détails du processus de Data Management ou à l'aide de la commande listFiles.

Utilisez la commande downloadFile pour télécharger le rapport sur un ordinateur local.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

`epmautomate runDMReport` *`REPORT_NAME PARAMETER=Value`* `"Report Output Format=[PDF|HTML|XLS|XLSX]"` où :

- *`REPORT_NAME`* est le nom du modèle de rapport de gestion des données à utiliser pour la génération du rapport.
- *`PARAMETER=Value`* indique les paramètres du rapport et leurs valeurs. Vous indiquez autant de paramètres que requis au format `PARAMETER=Value`. La liste des paramètres requis dépend du rapport que vous voulez générer.

> **Remarque**
>
> Les paramètres d'exécution de rapport sont définis lorsque vous concevez vos rapports. Pour exécuter cette commande, vous devez générer et utiliser ces paramètres et valeurs à partir de l'onglet Workflow. Pour générer les paramètres d'exécution d'un rapport, dans l'onglet Workflow de la gestion des données, cliquez sur **Exécution de rapport** et sélectionnez un groupe dans **Groupe de rapports**. Sélectionnez le rapport pour lequel vous voulez générer les paramètres, puis cliquez sur **Créer le script de rapport**. Indiquez éventuellement des valeurs de paramètre de rapport, puis sélectionnez un format de sortie et cliquez sur **OK**. Utilisez les paramètres affichés dans **Générer le script de rapport** afin d'indiquer les paramètres et valeurs d'exécution pour générer le rapport.

- `Report Output Format` indique le format de sortie du rapport. Les options valides sont PDF, HTML, XLS et XLSX. Le format de rapport par défaut est PDF.

Exemple

```
epmautomate runDMReport "TB Current Location By Target Acct (Cat,Per)"
"Period=Jul 14" "Category=Forecast" "Location=FCSTtoVISCONSOL1" "Rule
Name=FCSTtoVISCONSOL1" "Report Output Format=HTML"
```

runIntegration

Exécute un job d'intégration des données pour importer des données dans un processus métier Oracle Fusion Cloud Enterprise Performance Management ou exporter des données d'un processus métier vers un système externe.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Cette commande désapprouve la commande runDataRule. Oracle vous recommande de commencer à utiliser cette commande à la place de la commande runDataRule.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Super utilisateur

Syntaxe

```
epmautomate runIntegration JOB_NAME importMode=Append|Replace|"Map and
Validate"|"No Import"|Direct exportMode=Merge|Replace|Accumulate|Subtract|"No
Export"|Check periodName={PERIOD_NAME} [inputFileName=FILE_NAME] [PARAMETERS]
```

- Pour les intégrations en mode standard, vous devez indiquer les valeurs correspondant à `importMode`, `exportMode` et `periodName`.
- Pour les intégrations en mode rapide, vous devez indiquer la valeur correspondant à `exportMode`.
- Les noms de paramètre et leurs valeurs respectent la casse.

Dans cette commande :

- *JOB_NAME* est le nom d'un job d'intégration défini dans l'intégration des données.
- `importMode` détermine le mode d'import des données dans l'intégration des données. Les modes d'import acceptables sont les suivants :
  - `Append` : permet d'ajouter des éléments aux données existantes du PDV dans l'intégration des données.
  - `Replace` : permet de supprimer les données de PDV et de les remplacer par celles du fichier.
  - `Map and Validate` : permet d'ignorer l'import de données, mais de traiter à nouveau les données à l'aide de comptes logiques et de mappings mis à jour.
  - `No Import` : permet d'ignorer l'import de données dans la table intermédiaire d'intégration des données.
- `exportMode` détermine le mode de chargement des données dans l'application cible. Pour les intégrations en mode rapide, vous ne pouvez pas utiliser `Check` ou `No Export` en tant que valeur du paramètre `exportMode`. Les valeurs de mode d'export acceptables sont les suivantes :
  - `Merge` : permet de mettre à jour les données existantes et d'en ajouter de nouvelles.
  - `Replace` : permet d'effacer les données existantes dans le PDV et d'y charger de nouvelles données. En mode standard, les données sont effacées pour les dimensions Scénario, Version, Année, Période et Entité. En mode rapide, les données sont effacées pour les dimensions Année, Période et Entité. Vous pouvez définir des régions d'effacement personnalisées pour les deux modes.
  - `Accumulate` : permet d'ajouter les données aux données existantes. Champ d'application : Planning, modules Planning, Financial Consolidation and Close, Tax

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Reporting, Profitability and Cost Management et Enterprise Profitability and Cost Management.

- `Subtract` : permet de soustraire les données du solde existant. Champ d'application : Profitability and Cost Management et Enterprise Profitability and Cost Management. Pour les intégrations en mode rapide :
  - Vous ne pouvez pas utiliser `Check` ou `No Export` en tant que valeur du paramètre.
  - Pour Planning, les modules Planning et Financial Consolidation and Close, les seules valeurs valides sont `Replace`, `Merge` et `Accumulate`.
- `No Export` : permet d'ignorer l'export de données. Utilisez ce mode afin de charger des données dans la table intermédiaire pour examen avant chargement dans l'application cible.
- `Check` : permet d'effectuer uniquement une vérification de validation des données.

  Pour Oracle Fusion Cloud en tant que cible, seuls les modes d'export suivants sont pris en charge :
  - `MERGE`, pour fusionner les données de la table intermédiaire d'intégration des données avec les données existantes ;
  - `NONE`, pour ignorer l'export de données d'intégration des données vers l'application.

- `periodName` est le nom des périodes ou plages de périodes, chacune placée entre accolades, pour lesquelles importer ou exporter les données. Les conventions de dénomination de période acceptables sont les suivantes :
  - Pour les chargements de période unique, indiquez le nom de la période entre accolades, par exemple `{Jan-21}`.
  - Pour les chargements multipériode, indiquez les noms de période de début et de fin entre accolades, par exemple `{Jan-21}{Mar-21}` (afin de charger les données relatives à l'ensemble des périodes comprises entre janvier et mars 2021).
  - **Pour Planning, les modules Planning, Financial Consolidation and CloseFreeForm et Tax Reporting :** vous pouvez indiquer le nom de période et l'année du processus métier au format `{Jan#FY21}{Mar#FY21}` afin de charger les données relatives à l'ensemble des périodes comprises entre janvier et mars 2021. Le nom de période doit être placé entre accolades.
    - Période unique : désigne le nom de période de gestion des données pour une période unique définie dans le mapping de période.
    - Multipériode : désigne un chargement multipériode. Le paramètre est indiqué au format `{Month-Year}{Month-Year}`. Par exemple, {Jan-20}{Mar-20} pour un chargement multipériode entre janvier et mars 2020.
    - Nom de période Planning : désigne un nom de période Planning au format `{Month#Year}`, par exemple {Jan#FY20}{Mar#FY20}. Lorsque vous utilisez cette convention, vous n'avez pas besoin d'indiquer les noms de période d'intégration des données. Il vous suffit d'indiquer à la place les noms de membre pour les dimensions Année et Scénario.
      Ce paramètre est pris en charge dans les processus métier Planning, Tax Reporting et Financial Consolidation and Close. Il est fonctionnel à la fois pour vos applications de service et pour les déploiements cloud dérivés des sources de données sur site.

      L'utilisation de cette convention est utile lorsque la commande est déclenchée à partir d'un script Groovy Cloud EPM en capturant les noms de membre Année et Période. Le mapping de période d'application ou le mapping global de période doit exister en tant que valeur cible du mapping de période avec l'année et le mois.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

* Variable de substitution : il s'agit d'une extension du format de nom de période Planning précédent selon lequel une variable de substitution peut être indiquée à la place des noms de membre Année et Mois actuels au format `{Month#&CurYr}` `{&FcstMonth#&CurYr}`, par exemple `{Jan#&CurYr}{&FcstMonth#&CurYr}`.
  Une combinaison des noms de membre actuels et des variables de substitution est prise en charge.

  Ce format est pris en charge dans les processus métier Planning, Tax Reporting et Financial Consolidation and Close.

  Le mapping de période d'application ou le mapping global de période doit exister dans l'intégration des données de l'environnement dans lequel la commande est exécutée, avec l'année et le mois disponibles en tant que valeurs cibles du mapping de période. Dans ce cas, l'année et le mois désignent la valeur actuelle de la variable de substitution lors de l'exécution.
* PDV global : exécute le chargement des données pour la période de PDV global. Utilisez le format `{GLOBAL_POV}`.

> **Note**
>
> Si vous utilisez un paramètre de dénomination de période autre que les paramètres décrits dans cette discussion, vous obtiendrez le message d'erreur `Entrée non valide – HTTP 400`.

– `{GLOBAL_POV}` afin d'exécuter le chargement des données pour la période définie dans le PDV global du système ou dans les paramètres d'application de l'intégration des données.

> **Note**
>
> Le format relatif à la convention de dénomination de période {Month#Year} est pris en charge pour Planning, les modules Planning, Financial Consolidation and Close et Tax Reporting. En vertu de cette convention, vous pouvez indiquer des noms de membre pour les dimensions Année et Scénario au lieu des noms de période d'intégration des données. Cette approche est utile si la commande est déclenchée à partir d'un script Groovy en capturant les noms de membre Année et Période.
> La convention de dénomination de variable de substitution `{Jan#&CurYr}` `{&FcstMonth#&CurYr}` est une extension de la convention de dénomination de la période précédente. Vous pouvez indiquer une variable de substitution à la place des noms de membre Année et Mois si vous utilisez cette commande avec Planning, les modules Planning, Financial Consolidation and Close et Tax Reporting. Une combinaison des noms de membre et des variables de substitution est également prise en charge.
>
> Les conventions de dénomination de variable de substitution et de période précédente fonctionnent uniquement lorsque les mappings de période d'application ou les mappings globaux de période avec l'année et le mois en tant que valeurs cible existent déjà dans l'intégration des données.

- `inputFileName`, pour les chargements de données basés sur des fichiers, indique le nom du fichier (disponible dans la boîte de la réception) à partir duquel les données doivent être importées. Si vous indiquez le nom du répertoire dans la définition d'intégration, ne précisez que le nom du fichier. Si vous n'incluez pas le nom du répertoire dans la définition d'intégration, utilisez le format `inbox/`*`DIR_NAME/FILE_NAME`*, par exemple `inbox/`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

`GLBALANCES.txt` ou `inbox/EBSGL/GLBALANCES.txt`. Si le fichier a été chargé vers l'emplacement par défaut de l'environnement, utilisez la convention `#epminbox/FILE_NAME`, par exemple `#epminbox/GLBALANCES.txt`, pour identifier le fichier de données d'entrée. Ce paramètre est uniquement applicable aux chargements de données basés sur des fichiers natifs. Si vous n'indiquez pas la valeur de ce paramètre pour les chargements de données basés sur des fichiers, cette commande importe les données à partir du fichier indiqué dans la définition d'intégration. Si vous indiquez la valeur de ce paramètre pour les chargements de données qui ne sont pas basés sur des fichiers, la commande les ignore.

- *PARAMETERS* (facultatif) identifie les paramètres d'exécution au format *PARAMETER_NAME="PARAMETER"*. Les paramètres incluent à la fois des filtres source et des options cibles.

> **Note**
>
> Pour le moment, le seul paramètre que vous pouvez utiliser pour un type de dimension (métadonnées) d'application cible est `"Refresh Database"=Yes|No`.

Exemples

- Import sur une seule période :
```
epmAutomate runIntegration VisionDataLoad importMode=Replace exportMode=Merge
periodName="{Mar-15}"
```
- Import de plusieurs périodes :
```
epmAutomate runIntegration VisionDataLoad importMode=Replace exportMode=Merge
periodName="{Mar-15}{Jun-15}"
```
- Intégration incrémentielle des données basée sur un fichier :
```
epmAutomate runIntegration IncrementalFileLoad importMode=Replace
exportMode=Merge periodName="{Jan-20}{Mar-20}" inputFileName=File1.txt
```

runIntercompanyMatchingReport

Génère un rapport de rapprochement des intragroupes qui vous aide à rapprocher correctement les transactions entre les partenaires et les entités connexes dans le cadre du processus de consolidation, ainsi qu'à des fins d'analyse et d'audit.

Champ d'application

Financial Consolidation and Close

Rôles requis

- Administrateur de service
- Super utilisateur (si un niveau de sécurité supplémentaire est accordé via une liste de contrôle d'accès)
- Utilisateur (si un niveau de sécurité supplémentaire est accordé via une liste de contrôle d'accès)

Syntaxe

`epmautomate runIntercompanyMatchingReport JOB_NAME FILE_NAME [scenario=scenario] [years=years] [period=period_name] [ReportFormat=HTML]` où :

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `JOB_NAME` est le nom d'une définition de job de rapport intragroupe existant dans l'application. Pour plus d'informations, reportez-vous à la section Gestion des rapports de rapprochement intragroupe dans le guide *Utilisation de Financial Consolidation and Close*. Cette commande génère le rapport de rapprochement des intragroupes à l'aide des paramètres disponibles dans le job. Veillez à sélectionner Boîte d'envoi comme emplacement du rapport généré.

  > **Note**
  >
  > Vous pouvez remplacer les paramètres de scénario, d'année, de période et de format de rapport spécifiés dans le job en saisissant ces valeurs facultatives lors de l'exécution de cette commande.

- `FILE_NAME` est le nom du fichier de rapport. Ce rapport est généré dans la boîte d'envoi. Utilisez la commande downloadFile pour le télécharger sur un ordinateur local ou la commande sendMail afin de l'envoyer par courriel.
- `SCENARIO` (facultatif) est le nom d'un scénario, défini dans l'application, pour lequel le rapport doit être généré.
- `YEARS` (facultatif) est l'année pour laquelle le rapport doit être généré.
- `PERIOD` (facultatif) est la période pour laquelle le rapport doit être généré.
- `ReportFormat` (facultatif) est le format du rapport. Valeurs acceptables : HTML, PDF et XLSX.

> **Note**
>
> - Vous pouvez générer ce rapport pour une combinaison d'un scénario, d'une période et d'une année. Ne spécifiez pas plusieurs scénarios, années ou périodes.
> - Si vous n'indiquez aucune valeur pour le scénario, l'année, la période et le format de rapport, les valeurs correspondantes spécifiées dans la définition de job de rapport (identifiées par `JOB_NAME`) sont utilisées. Si elles sont indiquées, ces valeurs facultatives remplaceront les valeurs définies dans la définition de job de rapport.

Exemples

- Création du rapport par remplacement des valeurs définies dans la définition de job :
  ```
  epmautomate runIntercompanyMatchingReport IC_Job_01 SampleICReport.html
  scenario=Actual years=FY22 period=Jan reportFormat=HTML
  ```
- Création du rapport à l'aide des valeurs définies dans la définition de job :
  ```
  epmautomate runIntercompanyMatchingReport IC_Job_01 SampleICReport
  ```

runMatchingReport

Génère un rapport qui est défini dans le module Correspondance de transactions.

Pour obtenir la liste des rapports prédéfinis du module Correspondance de transactions et les paramètres permettant de les générer, reportez-vous à la section Génération de rapports prédéfinis dans le module Correspondance de transactions du guide *Administration d'Account Reconciliation* .

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Champ d'application

Account Reconciliation

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmautomate runMatchingReport` *`FILE_NAME GROUP_NAME REPORT_NAME REPORT_FORMAT`* `[Param=value]` où :

- `FILE_NAME` est le nom de fichier unique du rapport à générer. Si un rapport portant le même nom existe déjà sur le serveur, il sera remplacé. Employez la commande downloadFile pour télécharger ce rapport sur un ordinateur local.
- `GROUP_NAME` est le nom du groupe auquel est associé le rapport.
- `REPORT_NAME` est le nom unique du rapport à générer.
- `REPORT_FORMAT` est l'un des formats de rapport suivants :
  - PDF
  - HTML (non pris en charge pour les graphiques)
  - XLSX (non pris en charge pour les graphiques)
  - CSV
  - CSV2

> **Remarque**
>
> *`REPORT_FORMAT`* `CSV` n'autorise pas le formatage des données à partir d'un modèle alors que `CSV2` le permet. La génération d'un rapport formaté `CSV2` prend plus de temps que celle d'une sortie `CSV`.

- `Param=Value` (facultatif) identifie les paramètres requis pour générer le rapport. Par exemple, pour le rapport `Match Type Configuration` qui accepte le paramètre `status` avec la valeur `approved`, indiquez le paramètre et la valeur comme suit : `status=Approved`.

Exemple

```
epmautomate runMatchingReport Example_FileName "Transaction Matching" "Match Type
Configuration" HTML "status=Approved"
```

runPipeline

Exécute un pipeline d'intégration des données (orchestration de plusieurs jobs en un seul processus) en fonction des variables définies pour le pipeline. Pour obtenir des informations

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

détaillées sur les pipelines, reportez-vous à la section Utilisation du pipeline du guide *Administration de l'intégration des données pour Oracle Enterprise Performance Management Cloud*.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Enterprise Profitability and Cost Management, Tax Reporting, Account Reconciliation, Sales Planning et Strategic Workforce Planning

Rôles requis

Administrateur de service ou tout utilisateur défini comme utilisateur proxy pour l'exécution d'une définition de pipeline (à l'aide de la sécurité par emplacement ou en définissant le paramètre **Activer l'exécution de pipeline pour les utilisateurs non administrateurs** sur *Oui*).

Syntaxe

`epmautomate runPipeline` *`PIPELINE_CODE`* `[PARAMETER=VALUE]`, où :

- *`PIPELINE_CODE`* est un code défini lors de la création du pipeline dans l'intégration des données. Pour obtenir des informations détaillées, reportez-vous à la section Description du processus de pipeline du guide *Administration de l'intégration des données*.
- `PARAMETER=`*`VALUE`* (facultatif) identifie les paramètres et leurs valeurs pour exécuter le pipeline. Indiquez autant de paires paramètre-valeur que le processus l'exige.
  La liste des paramètres dépend du nombre de variables définies dans l'écran des variables de pipeline de l'intégration des données. Paramètres de pipeline par défaut et valeurs acceptables :
  - `STARTPERIOD` : première période pour laquelle charger les données. Ce nom de période doit être défini dans le mapping de période de l'intégration des données.
  - `ENDPERIOD` : dernière période pour laquelle charger les données. Ce nom de période doit être défini dans le mapping de période de l'intégration des données.
  - `IMPORTMODE` détermine le mode d'import des données dans l'intégration des données. Les valeurs acceptables sont les suivantes :
    - `Append` ajoute des éléments aux données existantes du PDV dans l'intégration des données.
    - `Replace` supprime les données de PDV et les remplace par les données du fichier indiqué dans la définition de pipeline ou sous forme de paramètre.
    - `Map and Validate` retraite les données avec les mappings et les comptes logiques mis à jour, sans importer les données du fichier indiqué dans la définition de pipeline ou sous forme de paramètre.
    - `No Import` ignore l'import de données dans la table intermédiaire d'intégration des données.
  - `EXPORTMODE` détermine le mode d'export des données vers l'intégration des données. Les modes d'export acceptables sont les suivants
    - `Merge` fusionne les données de la table intermédiaire d'intégration des données avec les données existantes dans le processus métier.
    - `Accumulate` ajoute les données de la table intermédiaire d'intégration des données au processus métier.
      Ce mode d'export est uniquement applicable à Planning, aux modules Planning, à FreeForm, à Sales Planning et à Strategic Workforce Planning.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

    - `Replace` supprime les données de PDV dans le processus métier et les remplace par les données de la table intermédiaire d'intégration des données. Les données sont effacées pour les dimensions Scénario, Version, Année, Période et Entité.
    - `No Export` ignore l'export de données de l'intégration des données vers le processus métier.
  - `ATTACH_LOGS` indique si les fichiers journaux doivent être compressés et joints aux notifications relatives à l'exécution du pipeline. Les valeurs acceptables sont `Y` pour Oui et `N` pour Non.
  - `SEND_MAIL` indique si un courriel doit être envoyé lorsque l'exécution du pipeline est terminée. Valeurs acceptables : `Always`, `No` (valeur par défaut), `On Failure` et `On Success`.
  - `SEND_TO` est la liste des adresses électroniques, séparées par des virgules, auxquelles le courriel doit être envoyé.

Exemple

```
epmautomate runPipeline DAILYLOAD "STARTPERIOD=Jan-24" "ENDPERIOD=Jan-24"
"IMPORTMODE=Replace" "EXPORTMODE=Merge" "SEND_MAIL=Always"
"SEND_TO=John.Doe@example.com, Jane.Doe@example.com" "ATTACH_LOGS=Y"
```

runPlanTypeMap

Copie des données d'une base de données en mode "block storage" vers une base de données en mode "aggregate storage", ou d'une base de données en mode "block storage" vers une autre base de données en mode "block storage", en fonction des paramètres indiqués dans un job de type `plan type map`.

Champ d'application

Planning, modules Planning, FreeForm, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate runPlanTypeMap` *`JOB_NAME`* `[clearData=true|false]` où :

- `JOB_NAME` est le nom d'un job de type `plan type map` défini dans l'application.
- `clearData` est un paramètre facultatif qui indique si les données dans la base de données cible doivent être enlevées avant la copie des données. Si cette valeur de paramètre n'est pas définie, la valeur par défaut `true` est utilisée.

  Les valeurs de paramètre (`true` ou `false`) doivent être indiquées entièrement en minuscules.

Exemple

```
epmautomate runPlanTypeMap CampaignToReporting clearData=false
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

runRuleSet

Lance un ensemble de règles métier.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service, Super utilisateur (si l'accès au lancement de règle est accordé)

Syntaxe

`epmautomate runRuleSet` *`RULESET_NAME`* `[`*`PARAMETER=VALUE`*`]` où :

- *`RULESET_NAME`* est le nom d'un ensemble de règles métier exactement tel qu'il est défini dans l'environnement.
- *`PARAMETER=VALUE`* indique des paramètres d'exécution facultatifs et leurs valeurs requises pour exécuter l'ensemble de règles.

> **Remarque**
>
> L'ensemble de règles est exécuté sur le type de plan vers lequel il est déployé.
>
> Utilisez la paire *`PARAMETER=VALUE`* pour indiquer autant d'invites d'exécution que l'ensemble de règles l'exige. L'exemple suivant utilise deux invites d'exécution (`Period` et `Entity`) et leurs valeurs (`Q1` et `USA`).
>
> Les valeurs par défaut sont utilisées si vous n'indiquez aucune valeur pour les paramètres d'exécution. La commande ignore les invites d'exécution qui ne sont pas des correspondances exactes avec celles définies pour l'ensemble de règles.
>
> Pour plus d'informations sur la saisie de plusieurs valeurs pour un paramètre, reportez-vous à la section Spécification de plusieurs valeurs pour un paramètre.

Exemple

```
epmautomate runRuleSet RollupUSSales Period=Q1 Entity=USA
```

runSupplementalDataReport

Génère des rapports relationnels affichant des données de Supplemental Data Manager.

Les rapports Supplemental Data sont regroupés en tant que rapports de non-consolidation dans Financial Consolidation and Close et Tax Reporting. Reportez-vous à la liste des rapports prédéfinis et des paramètres dans la section Génération d'un rapport pour Financial Consolidation and Close et Tax Reporting du guide *API REST pour Oracle Fusion Cloud EPM* afin d'obtenir la liste des rapports que vous pouvez générer et les paramètres permettant de les générer.

Champ d'application

Financial Consolidation and Close, et Tax Reporting.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Syntaxe

`epmautomate runSupplementalDataReport` *`FILE_NAME GROUP_NAME REPORT_NAME REPORT_FORMAT`* `[Param=value]` où :

- `FILE_NAME` est le nom de fichier unique du rapport.
- `GROUP_NAME` est le nom du groupe auquel est associé le rapport.
- `REPORT_NAME` est le nom unique du rapport à générer.
- `REPORT_FORMAT` est l'un des formats de rapport suivants :
  - PDF
  - HTML (non pris en charge pour les graphiques)
  - XLSX (non pris en charge pour les graphiques)
  - CSV
  - CSV2

  `REPORT_FORMAT` `CSV` n'autorise pas le formatage des données à partir d'un modèle alors que `CSV2` le permet. La génération d'un rapport formaté `CSV2` prend plus de temps que celle d'une sortie `CSV`.
- `Param=value` (facultatif) identifie les paramètres requis pour générer le rapport. Par exemple, pour générer le rapport `At Risk Tasks`, qui accepte un paramètre `schedule name` avec la valeur `monthly` et un paramètre `period` avec la valeur `Jan`, indiquez `"schedule name"=monthly period=Jan`.

Exemple

```
epmautomate runSupplementalDataReport Example_File_name Group1 "At Risk Tasks"
html "schedule name"=monthly period=Jan
```

runTaskManagerReport

Génère des rapports relationnels affichant des données de Task Manager.

Champ d'application

Enterprise Profitability and Cost Management, Financial Consolidation and Close et Tax Reporting.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Visualiseur

Syntaxe

`epmautomate runTaskManagerReport FILE_NAME GROUP_NAME REPORT_NAME REPORT_FORMAT [Param=value]` où :

- `FILE_NAME` est le nom de fichier unique du rapport.
- `GROUP_NAME` est le nom du groupe auquel est associé le rapport.
- `REPORT_NAME` est le nom unique du rapport à générer.
- `REPORT_FORMAT` est l'un des formats de rapport suivants :
  - PDF
  - HTML (non pris en charge pour les graphiques)
  - XLSX (non pris en charge pour les graphiques)
  - CSV
  - CSV2

> **Remarque**
>
> *`REPORT_FORMAT`* `CSV` n'autorise pas le formatage des données à partir d'un modèle alors que `CSV2` le permet. La génération d'un rapport formaté `CSV2` prend plus de temps que celle d'une sortie `CSV`.

- `Param=value` (facultatif) identifie les paramètres requis pour générer le rapport. Par exemple, pour générer le rapport `Early Tasks`, qui accepte un paramètre `schedule name` avec la valeur `monthly` et un paramètre `period` avec la valeur `Jan`, indiquez `"schedule name"=monthly period=Jan`.

Exemple

```
epmautomate runTaskManagerReport Example_File_name Group1 "Early Tasks" PDF
"schedule name"=monthly period=Jan
```

sendMail

Envoie un courriel, avec la possibilité de joindre des fichiers à partir d'Oracle Fusion Cloud Enterprise Performance Management.

Vous pouvez incorporer cette commande dans des scripts afin d'informer les utilisateurs de différentes conditions ou d'envoyer des rapports.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Narrative Reporting, Oracle Fusion Cloud Enterprise Data Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Syntaxe

`epmautomate sendMail ToAddress Subject [Body="MessageBody"] [Attachments=FILE1,FILE2]`, où :

- `ToAddress` identifie les adresses électroniques des destinataires, séparées par des points-virgules et entourées de guillemets. Example, `"jdoe@example.com;jane.doe@example.com"`.
- `Subject` identifie l'objet du courriel.
- `Body="MessageBody"` (facultatif) identifie le contenu du courriel. Si aucune valeur n'est indiquée, le courriel n'a pas de corps de message.

> **Note**
>
> Utilisez des balises HTML valides pour formater le corps du message et ainsi créer le format de courriel souhaité. L'intégralité du corps du message (y compris toutes les balises HTML) doit être indiquée sur une seule ligne et ne doit pas contenir de caractères de retour à la ligne. Reportez-vous à la section Exemples.

- `Attachments` (facultatif) identifie la liste des fichiers, séparés par des virgules, disponibles dans Cloud EPM pour être joints au courriel. Par exemple, `outbox/errorFile.txt,inbox/users.csv`.

> **Note**
>
> – Utilisez * (astérisque) comme caractère générique pour un caractère unique dans le nom de fichier. Par exemple, indiquez `outbox/user*.csv` afin de joindre tous les fichiers de la boîte d'envoi dont le nom comprend cinq lettres et correspond au modèle.
>
> – Vous pouvez joindre n'importe quel fichier, autre qu'un instantané, répertorié par la commande listFiles en tant que pièce jointe de courriel. La taille de la pièce jointe ne doit pas dépasser 10 Mo.

Exemples

- **Courriel non formaté :** `epmautomate sendMail "jdoe@example.com;jane.doe@example.com" "Data Load Process Failed" Body="Data Load 1 Failed" Attachments=outbox/Errorfile.txt,outbox/Errofile2.txt`
- **Courriel formaté :** `epmautomate sendMail jdoe@example.com "Send Formatted Email" "Body=<!DOCTYPE html> <html> <head> <title>EpmAutomate Email Formatting</title> </head> <body> <h1>EpmAutomate Email Formatting</h1><p>Hi,</p><p>Test Allocation Rules, Volume, and SPT data were loaded into FY22_Feb_Actual_Version POV.</p><p>Check the attachment for details.</p></body></html> "Attachments=outbox/loadResults.txt`

setApplicationAdminMode

Passe l'application en mode d'administration afin que l'accès à l'application soit limité aux administrateurs de service uniquement.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Cette commande est utile pour empêcher les utilisateurs de travailler sur l'application lorsque les administrateurs de service effectuent des opérations d'administration. L'application reste en mode d'administration jusqu'à ce que vous rétablissiez le paramétrage initial afin de permettre à tous les utilisateurs d'y accéder.

> **Note**
>
> Cette commande remplace la commande applicationAdminMode, qui a été désapprouvée mais pas encore été enlevée d'EPM Automate.

Utilisez la commande getApplicationAdminMode pour consulter le statut actuel de l'environnement.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Account Reconciliation, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de service

Syntaxe

```
epmautomate setApplicationAdminMode true|false
```

Dans cette commande, indiquez `true` pour passer l'application en mode d'administration et `false` pour la repasser en mode normal afin que tous les utilisateurs puissent y accéder.

Exemples

- Passer l'application en mode d'administration :
  `epmautomate setApplicationAdminMode true`
- Repasser l'application en mode de fonctionnement normal :
  `epmautomate setApplicationAdminMode false`

setDailyMaintenanceStartTime

Définit l'heure (UTC ou un autre fuseau horaire) à laquelle démarre la maintenance quotidienne de l'environnement. La maintenance de l'environnement ne doit pas nécessairement commencer à l'heure pile. Vous pouvez définir l'heure et les minutes auxquelles elle doit commencer.

Afin de garantir que cette commande n'interférera pas avec les exigences Oracle concernant la création des sauvegardes, elle ne modifiera pas l'heure de début de la maintenance si le processus de maintenance quotidien n'a pas été exécuté au cours des dernières 36 heures.

> **Remarque**
>
> Les administrateurs de service qui sont actuellement connectés à l'environnement avec un navigateur verront la nouvelle heure de début de la maintenance quotidienne uniquement après s'être déconnectés puis reconnectés.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Syntaxe

`epmautomate setDailyMaintenanceStartTime` *`StartTime`*, où *`StartTime`* correspond à l'heure (au format HH:MM sur 24 heures) à laquelle le processus de maintenance doit démarrer et, éventuellement, à un fuseau horaire. L'heure de début doit être comprise entre 00:00 et 23:59. Si l'heure de début n'est pas définie au format UTC, indiquez un fuseau horaire standard valide. Par exemple, `"14:35 America/Los_Angeles"` pour 14 h 35 (heure standard du Pacifique).

Exemples

- Définition de l'heure de début de la maintenance quotidienne sur 14 h 20 (UTC) :
  ```
  epmautomate setDailyMaintenanceStartTime 14:20
  ```
- Définition de l'heure de début de la maintenance quotidienne sur 14 h 35 (heure standard du Pacifique) :
  ```
  epmautomate setDailyMaintenanceStartTime "14:35 America/Los Angeles"
  ```

setDemoDates

Met à jour les données de démonstration internes d'Oracle selon les besoins.

Utilisez cette commande uniquement sur des installations configurées avec des données de démonstration internes Oracle.

Account Reconciliation uniquement : cette commande réinitialise les dates de tous les rapprochements auxquels sont associées des valeurs d'attribut `Demo Code` de `setdemodates` ou `setdemodatesnostatuschange`. Cette commande gère les rapprochements de 12 périodes au maximum : une période en cours et 11 périodes précédentes (historiques). Si les rapprochements de plus de deux périodes sont marqués avec l'attribut `Demo Code`, la commande traite ces périodes comme faisant partie de la période précédente. Les rapprochements ne disposant pas de cette valeur d'attribut ne sont pas concernés.

- Si la valeur est `setdemodates`, la commande réinitialise les dates de rapprochement en fonction de la date indiquée et d'un statut aléatoire.
- Si la valeur est `setdemodatesnostatuschange`, la commande réinitialise les dates de rapprochement en fonction de la date indiquée sans modifier le statut de rapprochement.

Tous les autres processus métier : cette commande réinitialise les dates de début et de fin des tâches, ainsi que d'autres informations relatives à la date, afin de préparer les tâches pour une démonstration. Elle calcule les nouvelles dates de tâche en fonction de la valeur de l'attribut `SETDEMODATES` défini dans la planification de la tâche, ainsi que de la valeur `Demo Date` que vous indiquez. Si la valeur `Demo Date` n'est pas indiquée, la commande utilise la date du jour pour calculer les nouvelles dates de tâche.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Remarque**
>
> Les tâches des planifications pour lesquelles la valeur `SETDEMODATES` n'est pas définie restent inchangées.

En fonction de la valeur `Demo Date` que vous indiquez, la commande avance toutes les dates associées à la tâche. Cela inclut les dates d'exécution principales (date de début, date de fin, etc.) et les dates auxiliaires, y compris celles de l'historique, les dates d'échéance de chaque workflow et la date de début (réelle). Cela n'a pas d'influence sur le statut des tâches.

Champ d'application

Account Reconciliation, Enterprise Profitability and Cost Management, Financial Consolidation and Close, FreeForm, Planning, modules Planning, Sales Planning, Strategic Workforce Planning et Tax Reporting

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmautomate setDemoDates [`*`DEMO_DATE`*`]`, où *`DEMO_DATE`* est une date facultative au format `YYYY-MM-DD`. Les rapprochements sont réinitialisés à la date actuelle si vous ne précisez pas cette valeur.

Exemple

```
epmautomate setDemoDates 2020-02-15
```

setEJJournalStatus

Dans Financial Consolidation and Close, définit le résultat d'imputation du journal Enterprise à partir du système ERP. Utilisez cette commande pour mettre à jour le statut d'imputation des journaux ayant le statut `Post in Progress`, quel que soit le statut de leur workflow.

Cette commande utilise un fichier CSV identifiant le statut d'import dans le système ERP. Vous utilisez la commande uploadFile pour charger le fichier d'import dans l'environnement. Le format de fichier CSV est le suivant :

```
Year,Period,Journal ID,Posting Status,Message
2020,Dec,1000001021,Posted,"SUCCESS"
2020,Dec,1000001022,Failed,"Row Header No: 2,10000415 - Linked value 6 does
not exist Application-defined or object-defined error 65171"
2020,Dec,1000001022,Failed,"Row Header No: 7,10000415 - Z_ECS_MSG (001)Enter
a valid account number"
2020,Dec,1000001022,Failed,"Row Header No: 7,10000415 - Z_ECS_MSG (002) Enter
a valid cost center"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

La colonne de message est facultative et peut être omise.

Cette commande n'exporte pas les données de journal Enterprise à partir de Financial Consolidation and Close et ne les importe pas dans le système ERP.

Champ d'application

Financial Consolidation and Close

Rôles requis

Administrateur de service

Syntaxe

`epmautomate setEJJournalStatus FILE_NAME.csv [OPERATION=posting|validation]`, où :

- *`FILE_NAME`* identifie le fichier CSV contenant le statut d'import dans le système ERP.
- `Operation` (facultatif) est l'opération à exécuter. Les valeurs possibles sont `posting` et `validation`. La valeur par défaut est `posting`.
  `operation=posting` fournit un fichier CSV qui contient les résultats de l'imputation des journaux avec les colonnes suivantes : Année, Période, ID de journal, Statut d'imputation (Imputé/Echec), Message, Code d'erreur et Message d'erreur. Les colonnes Message, Code d'erreur et Message d'erreur sont facultatives. Le statut d'imputation des journaux est mis à jour en fonction du statut fourni dans le fichier. Seuls les journaux dont le statut d'imputation est *Imputation en cours* sont mis à jour.

  `operation = validation` fournit un fichier CSV qui contient les résultats de la validation des journaux avec les colonnes suivantes : Année, Période, ID de journal, Statut de validation (Valide/Echec), Message, Code d'erreur et Message d'erreur. Les colonnes Message, Code d'erreur et Message d'erreur sont facultatives. Le statut de validation des journaux est mis à jour en fonction du statut fourni dans le fichier. Seuls les journaux dont le statut de validation est *Validation en cours* seront mis à jour.

Exemple

- Définition des résultats d'imputation des journaux Enterprise à partir d'un fichier :

  ```
  epmautomate setEJJournalStatus JournalStatus.csv
  ```

- Définition des résultats d'imputation des journaux Enterprise à partir d'un fichier et exécution d'une opération de validation :
  `epmautomate esetEJJournalStatus JournalStatus.csv Operation=validation`

setEncryptionKey

Définit une clé de cryptage personnalisée pour l'accès à la base de données.

Cette commande permet aux clients d'utiliser leur propre clé pour inclure Oracle Fusion Cloud Enterprise Performance Management à leur rotation standard de gestion des clés.

La clé de cryptage personnalisée prend effet après la prochaine maintenance quotidienne de l'environnement. Pour l'activer immédiatement, exécutez la commande resetService.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate setEncryptionKey key=`*`key`*, où *key* est une chaîne personnalisée de n'importe quelle longueur à utiliser comme clé de cryptage.

Exemples

- Définition de la clé de cryptage : `epmautomate setencryptionkey key=se!m+a2J`
- Suppression de la clé de cryptage : `epmautomate setEncryptionKey key=`

setEssbaseQryGovExecTime

Définit le délai maximal, en secondes, qu'une requête Oracle Essbase peut utiliser pour extraire et envoyer les informations avant que la requête soit interrompue.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Profitability and Cost Management, Enterprise Profitability and Cost Management, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate setEssbaseQryGovExecTime` *`TIME`*, où `TIME` identifie le nombre de secondes au bout duquel les requêtes Essbase doivent être interrompues. Cette valeur doit être un nombre entier ne dépassant pas 70 000.

Oracle recommande de ne pas définir cette valeur sur 0 (zéro) pour empêcher que les requêtes Essbase ne soient exécutées indéfiniment.

Exemple

```
epmautomate setEssbaseQryGovExecTime 600
```

setIdleSessionTimeout

Modifie le délai d'inactivité de session (en minutes) de l'environnement Oracle Fusion Cloud Enterprise Performance Management. Le nouveau délai d'expiration de session devient actif après la prochaine maintenance quotidienne de l'environnement. Utilisez cette commande pour modifier la valeur du délai d'expiration par défaut (30 minutes). Lorsqu'une session atteint le délai d'inactivité indiqué à l'aide de cette commande, l'utilisateur est redirigé vers la page de connexion.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate setIdleSessionTimeout MINUTES`, où MINUTES indique le nouveau délai d'inactivité de session (15 au minimum et 480 au maximum) en minutes.

Exemple

```
epmautomate setIdleSessionTimeout 30
```

setIPAllowlist (désapprouvé)

Configure une liste d'autorisation des adresses IP et des CIDR (Classless Inter-Domain Routings) autorisés à accéder à Oracle Fusion Cloud Enterprise Performance Management. Cette commande ajoute ou enlève des adresses IPv4 et des CIDR.

> **Note**
>
> Cette commande est désapprouvée et sera enlevée dans une mise à jour future. Au lieu d'utiliser cette commande, configurez un périmètre réseau dans Oracle Identity Cloud Service afin de limiter les connexions à des adresses IP ou des plages spécifiques pour l'ensemble du domaine. Reportez-vous aux sources d'information suivantes :
>
> - Configuration d'un périmètre réseau dans le guide *Guide de mise en route pour les administrateurs*
> - Gestion des périmètres réseau Oracle Identity Cloud Service dans le guide *Administration d'Oracle Identity Cloud Service*

> **Note**
>
> - Lorsque vous configurez une liste d'autorisation d'adresses IP pour un environnement Cloud EPM, vous autorisez les connexions à partir de ces adresses IP spécifiques uniquement. Dans ce scénario, il n'est pas possible de demander l'accès à partir d'un autre environnement Cloud EPM, sauf si vous ajoutez à la liste d'autorisation les adresses IP sortantes de la région où se trouve l'environnement à l'origine de la demande. Pour connaître les adresses IP que vous devez ajouter afin que les autres environnements puissent communiquer avec l'environnement pour lequel la liste d'autorisation d'adresses IP est configurée, reportez-vous à la section Adresses IP sortantes des centres de données et des régions Cloud EPM du *Guide des opérations*.
> - Si vous utilisez le périmètre réseau IDCS dans le service Identity Cloud où se trouve l'environnement, vous devez également ajouter les mêmes adresses IP dans ce périmètre réseau.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmAutomate setIPAllowlist add|remove` *`FILE_NAME`*`.txt` où :

- `add` ajoute à la liste d'autorisation les adresses IP et les CIDR répertoriés dans un fichier texte.
- `remove` supprime de la liste d'autorisation les adresses IP et les CIDR répertoriés dans un fichier texte.
- *`FILE_NAME`* est le nom d'un fichier texte répertoriant les adresses IP et les CIDR à ajouter à la liste d'autorisation ou à enlever de celle-ci. Les entrées du fichier doivent être séparées par un caractère de retour à la ligne. Utilisez la commande uploadFile pour charger ce fichier vers l'environnement. Chaque ligne du fichier doit être une adresse IPv4 ou un CIDR au format suivant :

  ```
  xxx.xxx.xxx.xxx
  xxx.xxx.xxx.xxx/n
  ```

> **Note**
>
> - Seules les adresses IP IPv4 sont prises en charge.
> - Utilisez le format CIDR, plutôt que des adresses IP individuelles, pour indiquer une plage continue d'adresses IP.
> - Pour désactiver une liste d'autorisation afin de permettre l'accès à partir de n'importe quelle adresse IP, utilisez la commande getIPAllowlist (désapprouvé) pour consigner l'ensemble des adresses IP et des CIDR existants dans un fichier. Chargez le fichier dans l'environnement, puis exécutez cette commande avec l'option `remove` comme indiqué dans l'exemple suivant :
>
>   ```
>   epmAutomate getIPAllowlist > myRemoveList.txt
>   epmAutomate uploadFile myRemoveList.txt
>   epmAutomate setIPAllowlist remove myRemoveList.txt
>   ```

Exemples

- Ajout d'adresses IP et de CIDR à une liste d'autorisation :

  ```
  epmAutomate setIPAllowlist add myAddList.txt
  ```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Suppression d'adresses IP d'une liste d'autorisation :

  ```
  epmAutomate setIPAllowlist remove myRemoveList1.txt
  ```

setManualDataAccess

Indique si Oracle est autorisé à accéder manuellement aux bases de données relationnelles et Oracle Essbase d'un environnement dans les situations d'urgence, où l'environnement ne répond pas et où le client n'a pas encore envoyé de demande de service pour enquêter et rendre l'environnement disponible.

En cas d'urgence, Oracle a recours à un processus interne au cours duquel un responsable de développement de haut niveau conduit un processus de vérification indépendant et autorise ensuite l'accès manuel aux bases de données relationnelles et Essbase. Vous pouvez utiliser cette commande pour empêcher Oracle d'accéder à ces bases de données sans votre approbation explicite. Vous avez également la possibilité d'interdire à Oracle d'accéder manuellement aux bases de données relationnelles et Essbase en cas d'urgence même si une demande de service est ouverte.

Le paramètre indiqué à l'aide de cette commande prend effet immédiatement. Par défaut, l'accès manuel aux données par Oracle est autorisé. Utilisez cette commande pour révoquer cet accès.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate setManualDataAccess Allow|Revoke [disableEmergencyAccess=true|false]`, où `disableEmergencyAccess` (facultatif) indique si vous souhaitez interdire tout accès manuel aux bases de données relationnelles et Essbase. Si vous définissez la valeur de cette propriété sur `true`, vous empêchez Oracle d'accéder manuellement à ces bases de données même si une demande de service est ouverte. La valeur par défaut est `false`.
Lorsque vous définissez la valeur de `disableEmergencyAccess` sur `true`, Oracle ne peut pas accéder aux bases de données relationnelles et Essbase si l'accès est requis pour dépanner et corriger un environnement arrêté. Si l'environnement est arrêté, vous n'avez pas la possibilité d'émettre cette commande pour permettre à Oracle d'accéder manuellement à ces bases de données.

Exemples

- Révoquer l'autorisation qui a été accordée pour accéder manuellement aux bases de données relationnelles et Essbase dans les situations d'urgence sans approbation explicite :
  ```
  epmautomate setManualDataAccess revoke
  ```
- Autoriser l'accès manuel aux bases de données relationnelles et Essbase en cas d'urgence :
  ```
  epmautomate setManualDataAccess allow
  ```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Interdire l'accès manuel aux bases de données relationnelles et Essbase même si une demande de service est ouverte :
  `epmautomate setManualDataAccess revoke disableEmergencyAccess=true`

setPeriodStatus

Définit un statut spécifique pour une période.

Champ d'application

Account Reconciliation

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Les utilisateurs dotés des rôles prédéfinis Super utilisateur, Utilisateur et Visualiseur peuvent nécessiter des rôles d'application supplémentaires.

Syntaxe

`epmautomate setPeriodStatus` *`PERIOD STATUS`*, où :

- *`PERIOD`* est le nom d'une période.
- *`STATUS`* est `OPEN`, `CLOSED` ou `LOCKED`.

Exemple

```
epmautomate setPeriodStatus "January 2015" OPEN
```

setRestrictedDataAccess

Configure l'environnement Oracle Fusion Cloud Enterprise Performance Management pour empêcher les administrateurs de service de consentir à la soumission d'un instantané d'application à Oracle lorsqu'ils se servent de l'utilitaire Fournir des commentaires.

Si vous avez pour politique de ne pas autoriser Oracle à accéder à vos données, utilisez cette commande pour gérer le partage des données lors de l'envoi de commentaires à Oracle. La définition d'un accès restreint aux données pour votre environnement désactive l'option Soumettre l'instantané d'application qui est affichée par l'utilitaire Fournir des commentaires.

Champ d'application

Account Reconciliation, Oracle Fusion Cloud Enterprise Data Management, Enterprise Profitability and Cost Management, Financial Consolidation and Close, FreeForm, Planning, modules Planning, Profitability and Cost Management, Narrative Reporting, Sales Planning, Strategic Workforce Planning et Tax Reporting.

Rôles requis

Administrateur de service

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Syntaxe

`epmautomate setRestrictedDataAccess true|false`
Pour afficher le statut en cours des restrictions d'accès aux données, utilisez getRestrictedDataAccess.

Exemples

- Empêcher les administrateurs de service de consentir à la soumission d'un instantané d'application :
  `epmautomate setRestrictedDataAccess true`
- Autoriser les administrateurs de service à consentir à la soumission d'un instantané d'application :
  `epmautomate setRestrictedDataAccess false`

setSubstVars

Crée ou met à jour des variables de substitution au niveau de l'application ou du cube.

Vous ne pouvez pas utiliser cette commande afin de définir plusieurs valeurs et/ou fonctions pour des variables de substitution.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate setSubstVars` *`CUBE_NAME SUBSTVAR=VALUE`* [*`SUBSTVAR=VALUE`*] où :

- *`CUBE_NAME`* est le cube (par exemple, Plan1, Plan2) pour lequel la variable de substitution est créée ou mise à jour. Utilisez `All` au lieu d'un nom de cube pour définir ou mettre à jour une variable de substitution au niveau de l'application.
- *`SUBSTVAR`* est le nom de la variable de substitution pour laquelle une valeur doit être définie ou mise à jour.
- *`VALUE`* est la nouvelle valeur de variable de substitution.

Exemples

- Création ou mise à jour d'une variable de substitution au niveau de l'application :
  `epmautomate setSubstVars ALL CurYear=2015 CurPeriod=Jan`
- Création ou mise à jour de variables de substitution au niveau du cube : `epmautomate setSubstVars Plan2 CurYear=2013 CurPeriod=Jan`

setVirusScanOnFileUploads

Permet à l'environnement d'analyser les fichiers à la recherche de virus avant qu'ils soient chargés vers Oracle Fusion Cloud Enterprise Performance Management.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Les environnements Cloud EPM sont protégés à l'aide d'un programme antivirus. Cette commande fournit une sécurité supplémentaire en vous permettant d'activer l'analyse antivirus sur les fichiers de chargement. L'analyse des fichiers avant leur chargement empêche les risques de chargement de virus dans l'environnement.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate setVirusScanOnFileUploads true|false`
Par défaut, l'analyse antivirus n'est pas activée (définie sur `false`). Lorsque cette valeur est définie sur `true`, Cloud EPM analyse tous les fichiers de chargement. Si un fichier est infecté par un virus, il n'est pas chargé dans l'environnement.

Exemples

- Activer l'analyse antivirus : `epmautomate setVirusScanOnFileUploads true`
- Désactiver l'analyse antivirus : `epmautomate setVirusScanOnFileUploads false`

simulateConcurrentUsage

Exécute différentes opérations simultanées dans un environnement en simulant des utilisateurs.

Cette commande peut être utilisée pour valider les performances de l'environnement et vérifier que le temps de réponse est acceptable lorsque le service est soumis à une charge pendant des opérations spécifiques exécutées par un nombre donné d'utilisateurs. Par exemple, cette commande peut être utilisée pour mesurer les performances si 50 utilisateurs ouvrent simultanément un formulaire à l'aide de différents PDV. Elle permet de tester les environnements dans des conditions de charge, en libre-service.

Cette commande effectue la simulation en exécutant les opérations indiquées pour un nombre donné d'utilisateurs et d'itérations. Elle exécute plusieurs itérations pour calculer les durées minimale, maximale et moyenne d'une opération particulière. Elle prend en charge les opérations ci-après pour effectuer des tests de charge d'utilisation simultanée :

- Ouverture de formulaires
- Enregistrement de formulaires
- Exécution de règles métier
- Exécution d'ensembles de règles métier
- Exécution de règles de données
- Ouverture de grilles ad hoc
- Exécution de rapports
- Exécution de liasses

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Note**
>
> Cette commande ne prend pas en charge les rapports et les liasses Financial Reporting ; elle ne prend en charge que les liasses et les rapports appartenant à Reports (anciennement Management Reports).

> **Caution**
>
> Cette commande exécute les opérations spécifiées sur l'environnement en cours et peut, selon l'opération, mettre à jour les données de l'environnement. Exécutez cette commande sur des environnements de test. Il est déconseillé d'exécuter cette commande dans des environnements de production.

Cette commande accepte en tant qu'entrée un fichier ZIP, qui doit déjà avoir été chargé dans la boîte de réception de l'environnement. Le fichier ZIP contient un fichier `requirement.csv` et les fichiers d'entrée qui prennent en charge les cas d'emploi inclus dans `requirement.csv`. Le fichier ZIP peut éventuellement contenir un fichier `UserVarMemberMapping.csv` afin de fournir le mapping de membre de variable utilisateur, un fichier `options. xml` afin de fournir des options Oracle Smart View for Office pour certains cas d'emploi, et un fichier `users.csv` afin de fournir les noms d'utilisateur et les mots de passe des utilisateurs déjà existants à employer plutôt que de créer d'autres utilisateurs. La commande simule alors les cas d'emploi et crée un rapport pouvant être envoyé par courriel à des destinataires.

> **Note**
>
> Cette commande ne génère pas de soumission Fournir des commentaires. Vous pouvez utiliser la fonctionnalité Fournir des commentaires (reportez-vous à la section Aider Oracle à collecter des informations de diagnostic à l'aide de l'utilitaire Fournir des commentaires du guide *Guide de mise en route pour les administrateurs*) à partir d'un écran de processus métier, de l'API REST Fournir des commentaires ou de la commande feedback pour générer une soumission Fournir des commentaires afin d'obtenir des détails sur l'environnement après l'exécution de la simulation.

Scénario d'utilisation 1 : test d'acceptation des performances de l'application pour 50 utilisateurs ouvrant simultanément un formulaire.

Solution :

1. Créez un fichier `requirement.csv` contenant des entrées semblables aux suivantes, en supposant que vous souhaitez ouvrir un formulaire nommé `Exchange Rates` stocké dans le dossier `Library/Global Assumption/` :

   ```
   # Type of Operation,Artifact Name,Number of Users,Input File,Additional
   Info
   Open Form, Library/Global Assumption/Exchange Rates,50,open_form_input.csv,
   ```

2. Créez un fichier `open_form_input.csv` à l'aide du format indiqué dans la section Fichier d'entrée d'ouverture de formulaire. Ce fichier comportera une entrée qui sera utilisée 50 fois. Si vous souhaitez ouvrir le même formulaire avec différents PDV, vous aurez autant d'entrées que de PDV à utiliser.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

3. Créez `UserVarMemberMapping.csv` à l'aide du format spécifié dans Création du fichier UserVarMemberMapping.csv si le mapping de membre de variable utilisateur doit être défini.
4. Exportez les options Smart View dans le fichier `options.xml`, si les options Smart View doivent être utilisées. Pour plus de détails, reportez-vous à la section Création du fichier options.xml.
5. Créez un fichier ZIP contenant les fichiers des étapes précédentes et chargez-le dans la boîte de réception.
6. Exécutez la commande `simulateConcurrentUsage` en utilisant le fichier ZIP de l'étape précédente comme fichier d'entrée.

Scénario d'utilisation 2 : simulation des performances en cas d'augmentation saisonnière de l'utilisation, par exemple, à la fin de l'exercice financier. Hypothèse : 100 utilisateurs enregistrent un formulaire avec un décalage de six secondes entre chaque utilisateur.

Solution :

1. Créez un fichier `requirement.csv` contenant des entrées semblables aux suivantes, en supposant que vous souhaitez enregistrer un formulaire nommé `Accessories Revenue` stocké dans le dossier `Library/Dashboards/` :

   ```
   # Type of Operation,Artifact Name,Number of Users,Input File,Additional
   Info
   Save Form, Library/Dashboards/Accessories Revenue,100,save_form_input.csv,
   ```

2. Créez un fichier `save_form_input.csv` à l'aide du format indiqué dans la section Fichier d'entrée d'enregistrement de formulaire.
3. Créez `UserVarMemberMapping.csv` à l'aide du format spécifié dans Création du fichier UserVarMemberMapping.csv si le mapping de membre de variable utilisateur doit être défini.
4. Exportez les options Smart View dans le fichier `options.xml`, si les options Smart View doivent être utilisées. Pour plus de détails, reportez-vous à la section Création du fichier options.xml.
5. Créez un fichier ZIP contenant les fichiers des étapes précédentes et chargez-le dans la boîte de réception.
6. Exécutez la commande `simulateConcurrentUsage` en utilisant le fichier ZIP de l'étape précédente comme fichier d'entrée, ainsi que les valeurs de propriété `iteration=1` et `lagTime=6`.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de service. Vous devez également disposer du rôle Administrateur de domaine d'identité pour utiliser testMode 0, 1 et 2.

Syntaxe

`epmautomate simulateConcurrentUsage` *`INPUT_FILE.zip`* `[testName=`*`TEST_NAME`*`] [iterations=`*`COUNT`*`] [notificationEmails="EMAIL_ADDRESS"] [testMode=0|1|2|3|4] [lagTime=`*`LAG_TIME`*`]` où :

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- *INPUT_FILE.zip* est le nom du fichier ZIP qui identifie les cas d'emploi. Utilisez la commande uploadFile (exemple de syntaxe de commande : `epmautomate uploadFile "C:/uploads/INPUT_FILE.zip" inbox`) pour charger ce fichier vers la boîte de réception avant d'exécuter cette commande. Le fichier ZIP doit contenir les fichiers suivants :
  - Un fichier CSV de cas d'emploi nommé `requirement.csv`. Chaque ligne du fichier CSV identifie le type d'opération à effectuer, le nom d'artefact, le nombre d'utilisateurs simultanés, le fichier d'entrée indiquant les détails de l'opération et d'autres informations relatives à chaque cas d'emploi. Reportez-vous à la section Création du fichier requirement.csv.
  - Les fichiers d'entrée qui contiennent les détails de chaque opération. Reportez-vous aux rubriques suivantes :
    * Fichier d'entrée d'ouverture de formulaire
    * Fichier d'entrée d'enregistrement de formulaire
    * Fichier d'entrée d'exécution de règle métier
    * Fichier d'entrée d'exécution d'ensemble de règles métier
    * Fichier d'entrée d'exécution de règle de données
    * Fichier d'entrée de grille ad hoc
    * Fichier d'entrée d'exécution de liasse
    * Fichier d'entrée d'exécution de rapport
  - Ajoutez éventuellement `UserVarMemberMapping.csv` au fichier ZIP d'entrée pour fournir le mapping de membre de variable utilisateur. Reportez-vous à la section Création du fichier UserVarMemberMapping.csv.
  - Ajoutez éventuellement `options.xml` au fichier ZIP d'entrée pour utiliser les options Smart View. Reportez-vous à la section Création du fichier options.xml.
  - Ajoutez éventuellement `users.csv` au fichier ZIP d'entrée pour fournir les noms d'utilisateur et les mots de passe des utilisateurs existants. Reportez-vous à la section Création du fichier users.csv.
- `testName` est un nom (50 caractères au maximum) permettant d'identifier l'appel actuel de ce test. Ce paramètre est requis pour les modes test 0, 1, 2 et 3, et facultatif pour le mode test 4.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Note

Vous devez utiliser le même nom de test si vous fractionnez la simulation de manière séquentielle avec les modes test 1 (pour créer des utilisateurs), 3 (pour exécuter la simulation avec les utilisateurs créés par le mode test 1), puis 2 (pour supprimer les utilisateurs créés par le mode test 1) avec le même fichier d'entrée. Exemple d'utilisation avec le nom de test *Sim02* :

- Créez des utilisateurs.

```
epmautomate simulateConcurrentUsage SimTest.zip testName=Sim02
testMode=1
```

- Exécutez des simulations avec les utilisateurs créés par l'exemple précédent :

```
epmautomate simulateConcurrentUsage SimTest.zip testName=Sim02
testMode=3
```

- Supprimez les utilisateurs créés pour cette simulation :

```
epmautomate simulateConcurrentUsage SimTest.zip testName=Sim02
testMode=2
```

- `iterations` (facultatif) est un nombre positif indiquant le nombre de fois où chaque cas d'emploi identifié dans `requirement.csv` doit être exécuté pour mesurer le temps de réponse. Si aucune valeur n'est indiquée, l'opération n'est exécutée qu'une seule fois.
- `notificationEmails` (facultatif) indique les adresses électroniques auxquelles les résultats de la commande doivent être envoyés. Si vous indiquez plusieurs adresses électroniques, séparez-les à l'aide d'un point-virgule. De plus, placez la liste d'adresses entre guillemets. Si aucune valeur n'est indiquée, les résultats sont envoyés par courriel à l'utilisateur qui a lancé la commande. Pour obtenir des informations détaillées sur ce rapport, reportez-vous à la section Exemple de rapport de simulation d'utilisation simultanée.
- `[testMode]` (facultatif) indique le mode de simulation de l'utilisation simultanée. La valeur par défaut est 0. Les valeurs acceptables sont les suivantes :
  - `0` : mode de simulation par défaut ; il ajoute des utilisateurs simulés à l'environnement et leur affecte le rôle Administrateur de service, exécute la simulation, puis supprime les utilisateurs simulés. Ce mode est utile si vous ne voulez exécuter le test qu'une seule fois.
    Les noms d'utilisateur simulés suivent le modèle <*testName*>Utilisateur<*userNumber*>. Les exemples suivants supposent que le nom de test utilisé est *Simulation* :

    Prénom : SimulationUser1, SimulationUser2, etc.

    Nom : SimulationUser1, Simulationser2, etc.

    Adresse électronique : SimulationUser1@example.oracle.com, SimulationUser2@example.oracle.com, etc.

    Nom d'utilisateur : SimulationUser1, SimulationUser2, etc.
  - `1` : ajoute des utilisateurs simulés à l'environnement et leur affecte le rôle Administrateur de service. N'exécute pas la simulation et ne supprime pas les utilisateurs simulés.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Après avoir utilisé ce mode, exécutez la commande en mode 3 pour lancer la simulation autant de fois que nécessaire. A la fin, exécutez la commande en mode 2 pour supprimer les utilisateurs simulés.

- `2` : supprime les utilisateurs simulés. Ne crée pas d'utilisateurs et n'exécute pas la simulation.
- `3` : exécute la simulation avec des utilisateurs simulés existants sans ajouter ni supprimer d'utilisateurs.
- `4` : se sert des utilisateurs définis dans le fichier `users.csv` inclus dans le fichier ZIP d'entrée pour exécuter la commande. Reportez-vous à la section Création du fichier users.csv. Ce mode ne crée pas d'utilisateurs pour la simulation. A la place, il se sert d'utilisateurs existants.

Si vous ne voulez exécuter l'utilisation simultanée qu'une seule fois, utilisez `testMode=0`. Pour effectuer une série de tests, procédez comme suit :

- Dans un premier temps, exécutez la commande avec `testMode=1` pour ajouter les utilisateurs simulés et leur affecter le rôle Administrateur de service.
- Exécutez ensuite la commande avec `testMode=3` pour lancer la simulation autant de fois que nécessaire.
- A la fin, exécutez la commande avec `testMode=2` pour supprimer les utilisateurs simulés.

- `[lagTime]` (facultatif) indique le nombre de secondes (5 ou plus) d'attente de la commande entre l'exécution de chaque cas d'emploi de `requirement.csv`. La valeur par défaut est 5 secondes. N'utilisez ni nombres négatifs (par exemple, -1), ni fractions (par exemple, 1/2), ni valeurs décimales.
  Après le lancement de l'exécution d'un cas d'emploi de `requirement.csv` par un utilisateur, la commande attend le nombre de secondes indiqué par ce paramètre pour lancer l'exécution du cas d'emploi par l'utilisateur suivant. Les activités utilisateur n'étant généralement pas lancées simultanément, la définition de ce paramètre contribue à créer une simulation plus réaliste du chargement dans un environnement.

Exemple

```
epmautomate simulateConcurrentUsage test_simulation.zip testName=Simulation
iterations=5
notificationEmails="jane.doe@example.com;john.doe@example.com;example@example.
com" lagTime=6
```

skipUpdate

Demande à Oracle d'ignorer l'application des mises à jour mensuelles à un environnement pendant trois cycles consécutifs au maximum ou enlève toutes les demandes de non-prise en compte de mise à jour effectuées précédemment à l'aide de cette commande pour que la mise à jour de l'environnement repasse sur le cycle standard.

Vous pouvez également utiliser cette commande pour répertorier les demandes de non-prise en compte de la mise à jour actuellement spécifiées pour un environnement. La non-prise en compte de la mise à jour du statut de l'environnement est incluse dans le rapport d'activité (dans Métriques opérationnelles) généré après avoir utilisé cette commande pour ignorer les mises à jour d'un environnement. Reportez-vous à la section Métriques opérationnelles du guide *Guide de mise en route pour les administrateurs*.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Les patches hebdomadaires et urgents pour le mois en cours continueront d'être appliqués à l'environnement, le cas échéant. Aucune mise à jour ne sera appliquée pendant les mois correspondant à la demande de délai de mise à niveau.

Vous ne pouvez pas ignorer les mises à jour pour un environnement qui est sur un patch exceptionnel. Par ailleurs, vous ne pouvez pas ignorer les mises à jour mensuelles si elles ont plus de trois mois d'écart avec la mise à jour actuelle de l'environnement. Par exemple, si l'environnement exécute la mise à jour mensuelle 23.12, vous pouvez ignorer les mises à jour 24.01, 24.02 et 24.03 mais pas 24.04. Pour obtenir des informations détaillées sur le fonctionnement des délais de mise à jour, reportez-vous à la section Demande de délais de mise à niveau pour les environnements de production du guide *Guide des opérations*.

> **Remarque**
>
> Si vous ignorez la mise à jour pour un seul de vos environnements (par exemple, vous ignorez la mise à jour de l'environnement de production mais pas celle de l'environnement de test) pendant trois mois, vos environnements auront trois versions d'écart. Dans un tel scénario, vous ne pourrez peut-être pas migrer les instantanés entre ces environnements.
> Par exemple, supposons que vos environnements de test et de production exécutent la mise à jour mensuelle 23.12, et que vous ignorez les mises à jour vers les versions 24.01, 24.02 et 24.03 pour l'environnement de production uniquement. Lorsque la version 24.03 devient disponible, votre environnement de test passe à la version 24.03 tandis que l'environnement de production reste à la version 23.12. Dans ce cas, la migration entre vos environnements de test et de production n'est pas prise en charge.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

```
epmautomate skipUpdate add|remove|list [version=UPDATE_NUMBER comment="COMMENT"]
```

où :

- `add` définit une demande de non-prise en compte de la mise à jour pour une mise à jour mensuelle donnée. Vous devez indiquer les paramètres suivants :
  - `version` : mise à jour mensuelle à ignorer. Vous pouvez ignorer jusqu'à trois des prochaines mises à jour mensuelles. Par exemple, si l'environnement exécute la mise à jour mensuelle 23.12, vous pouvez ignorer les mises à jour 24.01, 24.02 et/ou 24.03. Pour ignorer trois mises à jour mensuelles, exécutez la commande trois fois, en indiquant à chaque fois une mise à jour spécifique à ignorer avec, par exemple, `version=24.01`, `version=24.02`, puis `version=24.03`. Dans ce scénario, la mise à jour de l'environnement repasse sur le cycle standard lors du cycle mensuel 24.04.

    Si le cycle mensuel de la demande de non-prise en compte de la mise à jour et le cycle mensuel en cours ne sont pas consécutifs, Oracle mettra à jour l'environnement

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

comme indiqué, puis ignorera les mises à jour du cycle mensuel spécifié. Par exemple, l'environnement exécute la mise à jour mensuelle 23.12 et vous indiquez d'ignorer la mise à jour vers les versions 24.02 et 24.03. Dans ce cas, l'environnement sera mis à jour vers la version 24.01. Les mises à jour 24.02 et 24.03 seront ignorées. La mise à jour de l'environnement repasse sur le cycle standard lors du cycle mensuel 24.04.

  - `comment` : texte indiquant la raison de la non-prise en compte de la mise à jour. Les commentaires doivent être placés entre guillemets doubles.

- `remove` enlève toutes les demandes de non-prise en compte de la mise à jour pour l'environnement afin que la mise à jour de ce dernier repasse sur le cycle standard lors de la prochaine maintenance quotidienne. S'il existe plusieurs demandes de non-prise en compte de la mise à jour sur un environnement, cette commande les enlève toutes.
- `list` permet d'afficher les demandes de non-prise en compte de la mise à jour (ID de connexion de l'utilisateur ayant effectué la demande, commentaire, version dont les mises à jour sont ignorées et date à laquelle la demande a été effectuée) actuellement définies pour l'environnement, comme l'illustre l'image suivante :

```
c:\Oracle\EPM Automate\bin>epmautomate skipupdate add version=24.01 comment="Some Comment"
skipupdate completed successfully


c:\Oracle\EPM Automate\bin>epmautomate skipupdate add version=24.02 comment="Some Comment"
skipupdate completed successfully


c:\Oracle\EPM Automate\bin>epmautomate skipupdate add version=24.03 comment="Some Comment"
skipupdate completed successfully


c:\Oracle\EPM Automate\bin>epmautomate skipupdate list
skipupdate completed successfully

1] User:  example@example.com | Version: 24.03 | Comments: Some Comment | Timestamp: 2023-11-15T19:17:09Z
2] User:  example@example.com | Version: 24.02 | Comments: Some Comment | Timestamp: 2023-11-15T19:17:01Z
3] User:  example@example.com | Version: 24.01 | Comments: Some Comment | Timestamp: 2023-11-15T19:16:51Z

c:\Oracle\EPM Automate\bin>epmautomate skipupdate remove
skipupdate completed successfully
```

Exemples

- Demande de non-prise en compte de la mise à jour : `epmautomate skipUpdate add version=24.01 comment="Processus de clôture du trimestre en cours"`
- Visualisation des détails de la non-prise en compte de la mise à jour : `epmautomate skipUpdate list`
- Suppression de toutes les demandes de non-prise en compte de la mise à jour : `epmautomate skipUpdate remove`

snapshotCompareReport

Compare deux instantanés et crée le rapport de comparaison d'instantanés qui identifie les différences concernant les règles et les ensembles de règles de calcul, ainsi que les formulaires de données inclus dans les instantanés.

Le rapport affiche les différences dans les définitions de règle et de formulaire. Il repère également si un instantané contient une règle, un ensemble de règles ou un formulaire présent dans l'autre instantané. Les différences dans les ensembles de règles sont identifiées en fonction des règles qu'ils contiennent. Le rapport généré ne signale pas les différences dans les propriétés des cubes ou dimensions.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Dégradation récente des performances dans un environnement. Vous pouvez comparer l'instantané précédent avec l'instantané actuel afin d'identifier les différences qui ont pu entraîner cette dégradation des performances.
- Différence de performances ou de comportement entre deux environnements qui devraient avoir des performances ou un comportement opérationnel identiques. Dans ce cas, vous pouvez comparer les instantanés des deux environnements afin de comprendre les différences entre ceux-ci.
- Suspicion de disparition de règles ou de formulaires d'un environnement. Utilisez ce rapport pour comparer les artefacts qui existaient et les artefacts actuels.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmAutomate snapshotCompareReport` *`SOURCE_SNAPSHOT TARGET_SNAPSHOT`* `[reportName=`*`REPORT_NAME`*`.html]`, où :

- *`SOURCE_SNAPSHOT`* est le nom de l'instantané avec lequel la comparaison doit être effectuée. Le rapport contient des données sur les différences au niveau des règles et des formulaires dans cet instantané.
- *`TARGET_SNAPSHOT`* est le nom de l'instantané à comparer.

> **Note**
>
> – Les noms d'instantané peuvent être indiqués avec ou sans l'extension .zip.
>
> – Les deux instantanés doivent être disponibles dans l'environnement. Utilisez la commande uploadFile, copyFromObjectStorage ou copySnapshotFromInstance pour les charger vers l'environnement.

- *`REPORT_NAME`* (facultatif) est le nom du fichier de rapport. Le nom de rapport par défaut est `SnapshotCompare.html`.
  Utilisez la commande downloadFile pour télécharger le rapport.

Exemples

- `epmAutomate snapshotCompareReport "Instantané d'artefact" Backup_22-09-08.zip reportName=Snapshot_Diffs.html`
- `epmAutomate snapshotCompareReport backup_snapshot_22-Aug-08.zip backup_Snapshot_22-Sep-08.zip reportName=Sep_22_snapshot_compare_report.html`

sortMember

Trie les membres des dimensions Entité, Compte, Scénario, Versions et des dimensions libres.

Cette commande est utile pour trier les membres de dimension après leur chargement dans l'application.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Remarque**
>
> Vous ne pouvez pas utiliser cette commande pour trier les membres des dimensions Période, Années et Devise.

Champ d'application

Planning, modules Planning, FreeForm, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning.

Rôles requis

Administrateur de service

Syntaxe

`epmautomate sortMember` *`Member`* `[type=children|descendants] [order=ascending|descending]` où :

- *`Member`* est le nom du membre parent dont les descendants ou enfants sont à trier.
- `type` indique les membres à trier (facultatif). Les valeurs acceptables sont les suivantes :
  - `descendants` trie tous les sous-membres (enfants et descendants) du membre parent indiqué en tant que valeur de `Member`.
  - `children` (valeur par défaut) trie uniquement les membres appartenant au niveau placé immédiatement en dessous du membre parent indiqué en tant que valeur de `Member`.
- `order` identifie un ordre de tri (facultatif). Les valeurs acceptables sont les suivantes :
  - `ascending` : ordre de tri par défaut.
  - `descending`

Exemples

- Tri des enfants de la dimension Entity par ordre croissant : `epmautomate sortMember Entity`
- Tri de tous les sous-membres de la dimension Entity dans l'ordre décroissant : `epmautomate sortMember Entity type=descendants order=descending`

unassignRole

Enlève un rôle actuellement affecté aux utilisateurs, y compris l'utilisateur qui exécute cette commande, dont les ID de connexion sont inclus dans le fichier CSV ANSI ou UTF-8 utilisé avec cette commande. Vous pouvez utiliser cette commande pour enlever l'affectation d'un rôle prédéfini ou d'un rôle d'application.

Le format de fichier CSV est le suivant :

```
User Login
jane.doe@example.com
jdoe
```

Un administrateur de service utilise la commande uploadFile pour charger le fichier vers l'environnement.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Remarque**
>
> Placez entre guillemets doubles les noms de rôle qui contiennent un espace.

Une fois l'exécution terminée, la commande imprime les informations relatives à chaque entrée en échec dans la console. Reportez-vous à ces informations afin de comprendre pourquoi l'exécution de la commande a échoué pour certaines entrées dans le fichier CSV.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Pour enlever des affectations de rôle prédéfini :
  - Administrateur de service et/ou tout rôle prédéfini
  - Administrateur de domaine d'identité et tout rôle prédéfini
- Pour enlever des rôles d'application :
  - Administrateur de service
  - Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer

Syntaxe

`epmautomate unassignRole` *`FILE_NAME ROLE`*, où :

- *`FILE_NAME`* est le nom du fichier CSV contenant les ID de connexion des utilisateurs dont l'affectation de rôle est à révoquer. Indiquez l'extension CSV en minuscules.
  Les valeurs de connexion utilisateur ne respectent pas la casse. Par exemple, la valeur `jane.doe@example.com` est considérée comme identique à `Jane.Doe@Example.com`, ou à toute autre variation de la casse.
- `ROLE` identifie l'un des éléments suivants. Les noms de rôle ne respectent pas la casse.
  - Si vous enlevez l'affectation de rôles prédéfinis à des utilisateurs, `ROLE` doit identifier un rôle prédéfini applicable au service. Reportez-vous à la section Présentation des rôles prédéfinis du guide *Guide de mise en route pour les administrateurs*.
  - Si vous enlevez l'affectation de rôles d'application à des utilisateurs, `ROLE` doit identifier un rôle appartenant à l'application dans l'environnement en cours. Les rôles d'application sont répertoriés dans l'onglet **Rôles** de Contrôle d'accès . Afin d'obtenir une description des rôles d'application pour chaque processus métier, reportez-vous aux rubriques suivantes dans le guide *Administration du contrôle d'accès pour Oracle Enterprise Performance Management Cloud* :
    * Account Reconciliation
    * Enterprise Profitability and Cost Management
    * Planning, FreeForm, Financial Consolidation and Close et Tax Reporting
    * Profitability and Cost Management
    * Oracle Enterprise Data Management
    * Narrative Reporting

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exemples

- Annulation de l'affectation d'un rôle de domaine d'identité prédéfini à des utilisateurs : `epmautomate unassignRole remove_roles.csv "Administrateur de service"`
- Annulation de l'affection d'un rôle d'application à des utilisateurs : `epmautomate unassignRole example_file.csv "Gestionnaire d'accès à la liste des tâches"`

updateGuidedLearningSettings

Définit les valeurs requises pour activer Oracle Guided Learning (OGL) dans un environnement.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning et Strategic Workforce Planning

Rôles requis

Administrateur de service

Syntaxe

`epmautomate updateGuidedLearningSettings [oglAppId=`*`APP_ID`*`] [oglServerUrl=`*`OGL_SERVER_URL`*`]`, où :

- `OglAppId` (facultatif) est l'ID d'application OGL. Il s'agit de l'ID d'un regroupement logique de guides (contenus) sur le serveur OGL. Cette valeur peut contenir jusqu'à 100 caractères.
- `OglServerUrl` (facultatif) est l'URL du serveur OGL sur lequel vous disposez d'un compte OGL actif. La valeur par défaut est `https://guidedlearning.oracle.com`. Cette URL doit indiquer le protocole et peut contenir 300 caractères au maximum.

> **Note**
>
> Au moins l'un des paramètres précédents est requis.

Exemple

```
epmautomate updateGuidedLearnignSettings "oglAppId=123xyz" "oglServerUrl=https://
guidedlearning.oracle.com"
```

updateUsers

Modifie des attributs, tels que l'adresse électronique, le prénom et le nom de famille des utilisateurs d'Oracle Fusion Cloud Enterprise Performance Management, dans un domaine d'identité à l'aide de nouvelles valeurs identifiées dans un fichier CSV encodé en ANSI ou UTF-8 et chargé vers l'environnement.

Un administrateur de service utilise la commande uploadFile pour charger des fichiers dans un environnement. Toutes les colonnes du fichier CSV sont obligatoires. Vous devez indiquer une entrée valide dans chaque colonne. Cette commande valide chaque définition pour ces valeurs

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

obligatoires et affiche un message d'erreur identifiant chaque valeur manquante ou non valide. Le format de fichier d'entrée est le suivant :

```
First Name,Last Name,Email,User Login
Jane,Doe,jane.doe@example.com,jdoe
John,Doe,john.doe@example.com,john.doe@example.com
```

Si la valeur de connexion utilisateur dans le fichier CSV correspond à un compte existant dans le domaine d'identité, la commande modifie le compte utilisateur pour qu'il corresponde aux valeurs du fichier d'entrée. Les comptes utilisateur étant communs à tous les environnements pris en charge par un domaine d'identité, les informations utilisateur mises à jour sont disponibles dans tous les environnements qui partagent le même domaine d'identité. Cette commande n'a aucune incidence sur les rôles prédéfinis et les rôles propres à une application affectés à l'utilisateur.

> **Note**
>
> - Vous ne pouvez pas utiliser cette commande pour modifier les valeurs de connexion utilisateur.
> - Vous n'êtes pas autorisé à modifier vos propres attributs de compte.
> - Le fichier d'entrée contenant des caractères multi-octets doit utiliser l'encodage de caractères UTF-8.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Strategic Workforce Planning et Sales Planning.

Rôles requis

Administrateur de domaine d'identité et tout rôle prédéfini

Syntaxe

`epmautomate updateUsers` *`FILE_NAME`*, où *`FILE_NAME`* est le nom d'un fichier CSV contenant les informations utilisateur à modifier.

Exemple

```
epmautomate updateUsers update_user_info.csv
```

upgrade

Télécharge automatiquement et installe sans invite la version la plus récente d'EPM Automate.

Une fois que vous avez exécuté la commande login pour lancer une session, EPM Automate identifie la version actuellement installée. Si la version installée n'est pas la plus récente, l'application vous informe qu'une nouvelle version est disponible.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Remarque**
>
> - Sous Windows, par défaut, EPM Automate est installé dans le dossier `C:\Program Files\Oracle\EPM Automate`. Il s'agit d'un dossier protégé pour lequel seuls les utilisateurs appartenant au groupe Windows Administrateurs disposent d'un accès en écriture. Cette commande échoue si vous n'êtes pas membre du groupe Windows Administrateurs.
> - Cette commande ne modifie pas le répertoire d'installation actuel si EPM Automate n'est pas installé dans `C:\Program Files\Oracle\EPM Automate`.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Super utilisateur
- Utilisateur
- Visualiseur

Syntaxe

```
epmautomate upgrade
```

Exemple

```
epmautomate upgrade
```

uploadFile

Charge un fichier de l'ordinateur local vers le service. Utilisez cette commande pour charger des fichiers contenant des données, des métadonnées, des définitions de règle, des définitions de dimension, des transactions mappées, des modèles et des instantanés de sauvegarde.

Cette commande ne remplace pas les fichiers existants dans l'environnement. Elle affiche une erreur si le nom du fichier en cours de chargement est identique à celui d'un fichier dans l'emplacement de chargement.

Cette commande ne vous permet pas de charger des fichiers binaires (par exemple, des fichiers avec des extensions telles que .exe, .com, .cmd, .bin, .bat, .msi et .vbs) ni des fichiers avec des extensions non autorisées.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Migrations - Administrer

Syntaxe

`epmautomate uploadFile "FILE_NAME" [UPLOAD_LOCATION]`, où :

- *`FILE_NAME`* est le nom du fichier, y compris le chemin absolu si le fichier ne se trouve pas dans le répertoire à partir duquel vous exécutez EPM Automate.
- *`UPLOAD_LOCATION`* (facultatif) est l'emplacement Oracle Fusion Cloud Enterprise Performance Management vers lequel charger le fichier. N'indiquez pas d'emplacement de chargement si vous voulez charger les fichiers vers l'emplacement de chargement par défaut. Pour obtenir des informations détaillées, reportez-vous à Emplacements de fichier par défaut. Les valeurs prises en charge sont les suivantes :
  - `inbox`, pour charger les fichiers dans la boîte de réception. A l'exception de Profitability and Cost Management, les processus métier Cloud EPM recherchent d'éventuels fichiers à traiter à cet emplacement.
  - `profitinbox`, pour charger les fichiers devant être traités par Profitability and Cost Management.
  - `to_be_imported`, pour charger un instantané Narrative Reporting qui doit être importé lors de la prochaine maintenance quotidienne de l'environnement.
  - `inbox/`*`directory_name`*, pour charger des fichiers vers un répertoire de la boîte de réception à des fins de traitement par Data Management.
  - `outbox`, pour charger des fichiers dans la boîte d'envoi utilisée par des processus métier autres que Profitability and Cost Management.
  - `profitoutbox`, pour charger des fichiers dans la boîte d'envoi utilisée par Profitability and Cost Management.

Exemples

- Chargement d'un instantané dans l'emplacement par défaut :
  ```
  epmautomate uploadFile "C:/snapshots/backup_snapshot.zip"
  ```
- Chargement d'un fichier dans la boîte de réception de la gestion des données :
  ```
  epmautomate uploadFile "C:/pbcsdata/quarterlydata.csv" inbox
  ```
- Chargement d'un fichier dans un dossier de la boîte de réception (pour la gestion des données) :
  ```
  epmautomate uploadFile "C:/fdmee_data/data.zip" inbox/dm_folder
  ```
- Chargement d'un fichier dans profitinbox (Profitability and Cost Management) :
  ```
  epmautomate uploadFile "C:/profitability_data/data.zip" profitinbox
  ```
- Chargement d'un instantané Narrative Reporting du répertoire `C:\temp` vers l'emplacement to_be_imported :
  ```
  epmautomate uploadFile "C:\temp\EPRCS_Backup.tar.gz" to_be_imported
  ```

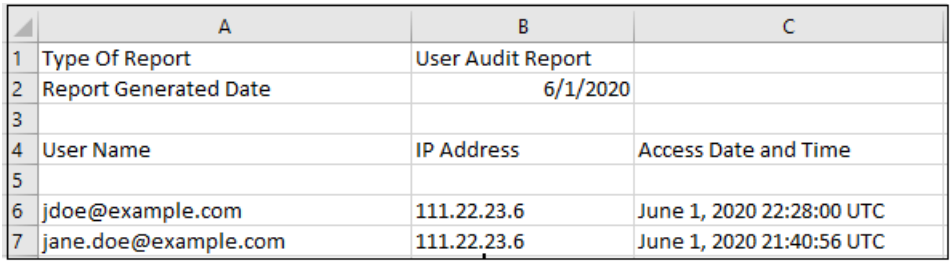
userAuditReport (désapprouvé)

Génère un rapport d'audit d'utilisateur (fichier CSV) et le stocke à l'emplacement de téléchargement par défaut.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Le rapport d'audit d'utilisateur contient des informations sur les utilisateurs qui se sont connectés à un environnement au cours d'une période donnée (120 derniers jours au maximum). Il répertorie l'ID de connexion utilisateur, l'adresse IP de l'ordinateur à partir duquel l'utilisateur s'est connecté, ainsi que la date et l'heure (par exemple, 28 juillet 2022 18:43:21 UTC) auxquelles il a accédé à l'environnement.

> **Remarque**
>
> Le rapport d'audit d'utilisateur ne consigne qu'une seule entrée de connexion pour un utilisateur qui s'est connecté à un environnement Oracle Fusion Cloud Enterprise Performance Management plusieurs fois dans un intervalle de cinq minutes.



	A	B	C
1	Type Of Report	User Audit Report	
2	Report Generated Date	6/1/2020	
3			
4	User Name	IP Address	Access Date and Time
5			
6	jdoe@example.com	111.22.23.6	June 1, 2020 22:28:00 UTC
7	jane.doe@example.com	111.22.23.6	June 1, 2020 21:40:56 UTC

Utilisez la commande downloadFile pour télécharger le rapport sur votre ordinateur.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Afficher

Syntaxe

`epmautomate userAuditReport` *`FROM_DATE TO_DATE REPORT_NAME`*, où :

- *`FROM_DATE`* indique la date de début (au format `YYYY-MM-DD`) de la période pour laquelle le rapport d'audit doit être généré.
- *`TO_DATE`* indique la date de fin (au format `YYYY-MM-DD`) de la période pour laquelle le rapport d'audit doit être généré.
- *`REPORT_NAME`* est le nom du fichier de rapport.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Remarque**
>
> Cloud EPM s'assure que seule une plage de dates valide est utilisée lors de la génération du rapport. Les validations suivantes sont effectuées pour les dates *de début* et *de fin* :
>
> - La date *FROM_DATE* ne peut pas être antérieure à la période de conservation maximale autorisée (120 jours) à compter de la date actuelle.
> - La date *TO_DATE* ne peut pas être postérieure à la période de conservation maximale à compter de la date *FROM_DATE*.
> - La date *TO_DATE* ne peut pas être antérieure à la date *FROM_DATE*.

> **Remarque**
>
> Ce rapport ne peut être généré que pour les 120 derniers jours.

Exemple

```
epmautomate userAuditReport 2016-10-15 2016-12-15 myAuditReport.CSV
```

userGroupReport

Génère un rapport (fichier CSV) qui répertorie les groupes auxquels les utilisateurs sont affectés dans Contrôle d'accès et le stocke à l'emplacement de téléchargement par défaut.

Le rapport indique si l'affectation de l'utilisateur au groupe est directe (membre du groupe) ou indirecte (membre du groupe qui est un enfant d'un groupe imbriqué).

Le rapport identifie l'ID de connexion, le prénom, le nom, l'adresse électronique, le groupe affecté et le type d'affectation de l'utilisateur au format suivant. Ce rapport est identique à la version CSV du rapport créé à partir de l'onglet Rapport sur le groupe d'utilisateurs dans Contrôle d'accès. Supposons par exemple que l'utilisateur `jdoe` est un membre du groupe `Test1`, qui est un enfant du groupe imbriqué `Test2`. Dans ce scénario, le rapport affiche les informations suivantes pour `jdoe` :

```
User Login,First Name,Last Name,Email,Direct,Group
jdoe, John, Doe, jdoe@example.com, Yes, test1
jdoe, John, Doe, jdoe@example.com, No, test2
```

Utilisez la commande downloadFile pour télécharger le rapport sur votre ordinateur.

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Afficher

Syntaxe

`epmautomate userGroupReport` *`REPORT_NAME`*, où *`REPORT_NAME`* est le nom du fichier de rapport.

Exemple

```
epmautomate userGroupReport UsrGrpReport.CSV
```

userLoginReport

Génère le rapport sur les connexions utilisateur contenant des informations sur les utilisateurs qui ont accédé à l'environnement dans une plage de dates donnée. Le rapport répertorie le nom d'utilisateur, l'adresse IP de l'appareil employé pour accéder à l'environnement, et la date et l'heure (UTC) auxquelles l'utilisateur a accédé à l'environnement.

Vous pouvez créer ce rapport pour n'importe quelle plage de dates des 120 derniers jours.

	A	B	C
1	**User Login**	**IP Address**	**Access Date and Time**
2	jdoe	111.222.22.4	September 11, 2025 19:18:06 UTC
3	jane.doe@example.com	222.333.44.5	September 11, 2025 17:56:46 UTC
4	example@example.com	222.333.55.7	September 11, 2025 17:31:13 UTC

Champ d'application

Planning, modules Planning, FreeForm, Financial Consolidation and Close, Tax Reporting, Account Reconciliation, Profitability and Cost Management, Enterprise Profitability and Cost Management, Oracle Fusion Cloud Enterprise Data Management, Narrative Reporting, Sales Planning et Strategic Workforce Planning.

Rôles requis

- Administrateur de service
- Tout rôle prédéfini et rôle d'application Contrôle d'accès - Gérer

Syntaxe

`epmautomate userLoginReport` *`FROM_DATE TO_DATE REPORT_NAME`* où :

- *`FROM_DATE`* indique la date de début (au format `YYYY-MM-DD`) de la période pour laquelle le rapport doit être généré
- *`TO_DATE`* indique la date de fin (au format `YYYY-MM-DD`) de la période pour laquelle le rapport doit être généré
- *`REPORT_NAME`* est le nom du fichier de rapport (format .CSV)

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

> **Note**
>
> Oracle Fusion Cloud Enterprise Performance Management s'assure que seule une plage de dates valide est utilisée lors de la génération du rapport. Les validations suivantes sont effectuées pour les dates *de début* et *de fin* :
>
> - La date *FROM_DATE* ne peut pas être antérieure à la période de conservation maximale autorisée (120 jours) à compter de la date du jour.
> - La date *TO_DATE* ne peut pas être postérieure à la période de conservation maximale à compter de la date *FROM_DATE*.
> - La date *TO_DATE* ne peut pas être antérieure à la date *FROM_DATE*.

> **Note**
>
> Ce rapport ne peut être généré que pour les 120 derniers jours car Cloud EPM tient à jour l'historique d'audit de connexion utilisateur des 120 derniers jours uniquement.

Exemple

```
epmautomate userLoginReport 2025-08-01 2025-09-25 myEPMLoginReport.CSV
```

validateConsolidationMetadata

Valide les métadonnées de l'environnement pour garantir une actualisation et une consolidation de base de données sans erreur.

Après avoir importé les métadonnées à l'aide de la commande importMetadata, exécutez cette commande pour valider les métadonnées afin de garantir une actualisation de base de données sans erreur lorsque vous exécutez la commande refreshCube. Si les métadonnées de consolidation ne sont pas correctes, la consolidation peut également échouer.

Cette commande affiche 0 (zéro) ou le nombre d'erreurs de validation dans la console à partir de laquelle vous l'exécutez. Elle écrit les erreurs de validation dans un fichier CSV dont vous pouvez vous servir pour corriger les erreurs de métadonnées. Employez la commande downloadFile pour télécharger le fichier CSV généré sur un serveur local.

Champ d'application

Financial Consolidation and Close

Rôles requis

Administrateur de service

Syntaxe

`epmautomate validateConsolidationMetadata` *`LOG_FILE_NAME`* où `LOG_FILE_NAME` indique le nom du fichier contenant les informations sur les erreurs identifiées par la commande.

Exemples

```
epmautomate validateConsolidationMetadata validation_error.csv
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

validateModel

Valide un modèle Enterprise Profitability and Cost Management et consigne le résultat dans un fichier.

Champ d'application

Enterprise Profitability and Cost Management

Rôles requis

Administrateur de service

Syntaxe

> **Note**
>
> Si des variables de substitution existent, vous pouvez les utiliser à la place des valeurs de paramètre réelles. Pour obtenir des informations détaillées, reportez-vous à la section Utilisation de variables de substitution avec certaines commandes Enterprise Profitability and Cost Management.

`epmautomate validateModel "modelName" FILE_NAME.txt [messageType=All|Warning|Error] [ruleStatus=All|Enabled|Disabled]` où :

- `modelName` est le nom du modèle Enterprise Profitability and Cost Management à valider. Cette valeur doit être placée entre guillemets doubles.
- *`FILE_NAME`* est le nom unique d'un fichier texte dans lequel la commande doit écrire le résultat de validation de modèle. Ce fichier, créé dans la boîte d'envoi, peut être téléchargé à l'aide de la commande downloadFile.
- `messageType` (facultatif) est le statut des informations à inclure dans la sortie de validation de modèle. Valeurs de paramètres possibles :
  - `All`, qui écrit les erreurs et les avertissements dans le fichier de sortie de validation.
  - `Error`, qui enregistre uniquement les erreurs dans le fichier de sortie de validation. Il s'agit de la valeur par défaut.
  - `Warning`, qui enregistre uniquement les avertissements de validation de modèle dans le fichier de sortie de validation.
- `ruleStatus` (facultatif) est l'un des statuts suivants :
  - `All` valide toutes les règles pour le modèle sélectionné. Il s'agit de la valeur par défaut.
  - `All` valide uniquement les règles activées.
  - `Disabled` valide uniquement les règles désactivées.

Exemple

```
epmautomate validateModel "10 Actuals Allocation Process" results.txt
messageType=All ruleStatus=Enabled
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Codes de sortie

EPM Automate renvoie un code de sortie et un message pour indiquer le statut de l'opération. Les codes de sortie sont regroupés sous cinq numéros de code ; chaque code peut indiquer de nombreuses conditions d'erreur. Passez en revue le message associé pour identifier la condition spécifique ayant causé l'erreur.

De plus, EPM Automate crée un fichier journal *(`COMMANDNAME_TIMESTAMP.log`*, par exemple, `uploadfile_16_11_2016_11_27_10.log`) pour chaque échec d'exécution de commande. Les fichiers journaux sont créés sur l'ordinateur à partir duquel vous exécutez EPM Automate.

Erreurs du code de sortie 1

`Command failed to execute` EPM Automate utilise ce code de sortie pour afficher les messages liés aux codes de statut HTTP 200 et 400. Ces codes sont renvoyés par les API REST qu'EPM Automate utilise.

`Insufficient privileges to perform the operation` Cette erreur est affichée si l'utilisateur dont les informations d'identification sont utilisées pour se connecter au service ne dispose pas des privilèges suffisant pour effectuer l'opération que vous avez tentée.

Connectez-vous avec un compte qui dispose des privilèges suffisants pour effectuer l'opération. En général, seuls les administrateurs de service sont autorisés à effectuer des opérations dans le service.

`Resource does not exist` Cette erreur est affichée si le fichier ou l'instantané que vous voulez supprimer ou télécharger n'existe pas dans le service.

Utilisez la commande `listfiles` pour vérifier le nom du fichier et son emplacement.

`Instantané non valide SNAPSHOT` Cette erreur est affichée lorsque le service ne peut pas valider l'instantané que vous avez indiqué pour l'opération d'export ou d'import.

Vérifiez que vous utilisez un instantané valide.

`Erreur de serveur interne. Impossible de supprimer le fichier : FILE_NAME Utilisez la fonctionnalité Fournir des commentaires et donnez des détails` Cette erreur est affichée si le fichier ou l'instantané n'a pas pu être supprimé du service en raison d'une erreur de serveur.

Signalez le problème à Oracle à l'aide de la commande feedback ou de la fonctionnalité Fournir des commentaires.

`Fichier non valide : FILE_NAME` Cette erreur est affichée si le fichier ou l'instantané que vous voulez supprimer ou télécharger n'existe pas dans le service, ou si le nom du fichier n'a pas le format requis.

Utilisez la commande `listfiles` pour vérifier le nom du fichier et son emplacement.

`L'opération de recréation est en cours d'exécution depuis longtemps. Contactez le support technique` Cette erreur est affichée si l'opération de recréation que vous avez lancée n'est pas terminée dans un délai d'une heure.

Signalez le problème à Oracle à l'aide de la commande feedback ou de la fonctionnalité Fournir des commentaires.

`L'opération de réinitialisation du service est en cours d'exécution depuis longtemps. Contactez le support technique` Cette erreur est affichée si

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

l'opération de réinitialisation du service que vous avez lancée n'est pas terminée dans un délai d'une heure.

Signalez le problème à Oracle à l'aide de la commande feedback ou de la fonctionnalité Fournir des commentaires.

`Impossible d'effectuer l'opération. Une autre instance est en cours. Réessayez ultérieurement` Cette erreur est affichée si vous essayez d'exécuter la commande `copysnapshotfrominstance` alors qu'une autre instance de la commande est active.

Attendez que la commande `copysnapshotfrominstance` soit terminée avant d'essayer de réexécuter la commande.

`Impossible d'effectuer l'opération. Un autre script de maintenance est en cours. Réessayez ultérieurement` Cette erreur est affichée si vous essayez d'exécuter la commande `copysnapshotfrominstance`, `recreate` ou `resetservice` lorsque la maintenance quotidienne ou le processus de réinitialisation du service est en cours d'exécution.

Réexécutez l'opération une fois la maintenance ou le processus de réinitialisation terminé.

`Echec de la connexion à l'instance source : SOURCE_URL` Cette erreur est affichée si EPM Automate ne peut pas se connecter à l'environnement source pour lancer la commande `copysnapshotfrominstance`.

Vérifiez que les informations d'identification, le domaine d'identité et l'URL utilisée pour accéder à l'environnement source sont valides.

`Erreur de serveur interne. Echec de la copie de l'instantané de l'instance source. Utilisez la fonctionnalité Fournir des commentaires et donnez des détails` Cette erreur est affichée lorsqu'EPM Automate rencontre un problème inattendu lors de l'exécution de la commande `copysnapshotfrominstance`.

Signalez le problème à Oracle à l'aide de la commande `feedback` ou de la fonctionnalité Fournir des commentaires.

`Erreur de serveur interne. Utilisez la fonctionnalité Fournir des commentaires et donnez des détails` Cette erreur est affichée pour indiquer plusieurs conditions de serveur interne qui exigent des actions correctives de la part d'Oracle.

Signalez le problème à Oracle à l'aide de la commande `feedback` ou de la fonctionnalité Fournir des commentaires.

`L'instantané SNAPSHOT_NAME existe déjà. Supprimez l'instantané et réexécutez la commande` Cette erreur est affichée lorsque vous téléchargez ou chargez un instantané dans un emplacement où se trouve un autre instantané du même nom.

Supprimez ou enlevez l'instantané existant, puis réessayez la commande.

`Erreur lors de l'extraction de l'instantané. Réessayez avec un instantané approprié` Cette erreur est affichée lorsqu'EPM Automate ne peut pas extraire le contenu de l'instantané lors de l'exécution de la commande `importsnapshot`.

Vérifiez que l'instantané est valide.

`Erreur de serveur interne. Impossible d'ouvrir le fichier pour écriture. Utilisez la fonctionnalité Fournir des commentaires et donnez des détails` Cette erreur est affichée si des erreurs entraînent la création ou la mise à jour de fichiers CSV, par exemple lors de la génération du rapport d'audit.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Signalez le problème à Oracle à l'aide de la commande `feedback` ou de la fonctionnalité Fournir des commentaires.

`No matching records found, please select a different date range` Cette erreur est affichée si vous exécutez la commande `userauditreport` afin de générer le rapport d'audit pour une plage de dates pour laquelle les données d'audit ne sont pas disponibles.

Indiquez une plage de dates valide, puis réexécutez la commande `userauditreport`. Le service conserve un historique d'audit pour les 365 derniers jours uniquement.

`Un fichier portant le même nom existe : FILE_NAME. Choisissez un autre nom de fichier` Cette erreur est affichée si un rapport portant le nom du rapport d'audit que vous avez indiqué existe déjà dans le service.

Supprimez le fichier existant du service ou indiquez un autre nom de fichier, puis réexécutez la commande `userauditreport`.

`Echec de l'opération avec le statut $1. Utilisez la fonctionnalité Fournir des commentaires` Ce message indique une erreur de serveur interne qui entraîne l'échec du processus de réinitialisation ou de recréation du service.

Signalez le problème à Oracle à l'aide de la commande `feedback` ou de la fonctionnalité Fournir des commentaires.

`EPMCSS-20643 : Echec de l'ajout des utilisateurs. Le fichier FILE_NAME.csv est introuvable. Indiquez un nom de fichier valide` Cette erreur est affichée si le fichier CSV indiqué et contenant les informations sur les utilisateurs à ajouter n'est pas disponible dans la boîte de réception.

Utilisez la commande `listfiles` pour vérifier le nom du fichier et son emplacement. Si le fichier ne se trouve pas dans la boîte de réception, utilisez la commande uploadFile pour le charger.

`EPMCSS-20644 : Echec de la suppression des utilisateurs. Le fichier FILE_NAME.csv est introuvable. Indiquez un nom de fichier valide` Cette erreur est affichée si le fichier CSV indiqué et contenant les informations sur les utilisateurs à supprimer n'est pas disponible dans la boîte de réception.

Utilisez la commande `listfiles` pour vérifier le nom du fichier et son emplacement. Si le fichier ne se trouve pas dans la boîte de réception, utilisez la commande uploadFile pour le charger.

`20645 : Echec de l'affectation du rôle aux utilisateurs. Nom de rôle role non valide. Indiquez un nom de rôle valide` Cette erreur est affichée si le rôle indiqué dans le fichier CSV n'est pas pris en charge.

Vérifiez que le nom de rôle indiqué dans le fichier est `Administrateur de service`, `Super utilisateur`, `Utilisateur` ou `Visualiseur`.

Utilisez la commande `listfiles` pour vérifier le nom du fichier et son emplacement. Si le fichier ne se trouve pas dans la boîte de réception, utilisez la commande uploadFile pour le charger.

Erreurs du code de sortie 6

`Service Unavailable` Le service n'est pas disponible à cause de l'erreur HTTP 404.

Vérifiez la disponibilité du service en accédant à celui-ci à partir d'un navigateur sur l'ordinateur d'où vous exécutez EPM Automate. Si le serveur est arrêté pour une quelconque raison, attendez un peu et réessayez, ou contactez le support technique Oracle.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

`Read/Write timeout` Cette erreur est affichée si le socket client expire (le délai d'expiration du socket est de 15 minutes) lors d'une opération de lecture/écriture en raison d'un réseau lent ou de problèmes de pare-feu.

Réexécutez la commande ayant échoué lorsque le débit du réseau est élevé. Si l'échec est dû à des paramètres de pare-feu, contactez votre administrateur réseau.

Erreurs du code de sortie 7

EPM Automate affiche cette erreur s'il ne peut pas exécuter une commande. Le message d'erreur (par exemple, `Invalid command`) indique la raison pour laquelle l'erreur s'est produite. **`Impossible d'ouvrir le fichier de mots de passe`** ***`FILE_NAME`*** Fichier de mots de passe crypté non valide, par exemple : *`PASSWORD_FILE`*`.EPW`. EPM Automate n'a pas trouvé le fichier à l'emplacement que vous avez indiqué ou le fichier n'est pas au format requis.

Vérifiez le nom du fichier et le chemin. Si le fichier ne peut pas être analysé en raison d'un format non valide, utilisez la commande encrypt pour le recréer.

`Impossible d'analyser le fichier de mots de passe` ***`FILE_NAME`*** EPM Automate n'est pas parvenu à analyser le fichier de mots de passe crypté car il a été endommagé ou son format n'était pas valide.

Utilisez la commande encrypt pour recréer le fichier.

`Impossible de se connecter à` ***`URL`*****`. Cause racine :`** ***`MESSAGE`*** Cette erreur est affichée si une connexion ne peut pas être établie en raison d'une URL incorrecte. Le message affiché comme cause racine détaille l'échec sous-jacent dû à l'utilisation d'une URL incorrecte.

- Vérifiez que vous utilisez une URL valide.
- Si vos paramètres de proxy requièrent que vous vous authentifiiez auprès du serveur proxy pour vous connecter à Internet, spécifiez le nom d'utilisateur d'un serveur proxy, son domaine et son mot de passe (ou utilisez un fichier de mots de passe cryptés qui contient le mot de passe de serveur proxy) pour vous connecter. Contactez votre administrateur réseau si vous avez besoin d'aide.

`Impossible de se connecter à` ***`URL`*** **`Protocole non pris en charge`** La commande login a échoué car l'URL indiquée utilise un protocole non pris en charge. Le message d'erreur associé identifie le protocole non pris en charge.

Assurez-vous que l'URL que vous utilisez avec la commande login utilise le protocole sécurisé (HTTPS).

`La session n'est pas authentifiée. Exécutez la commande "login" avant d'exécuter toute autre commande` Vous avez tenté d'exécuter une commande avant d'établir une session à l'aide de la commande `login`.

Exécutez la commande login pour établir une connexion sécurisée à un environnement avant de tenter d'exécuter d'autres commandes.

`Invalid parameter` Ce message indique une erreur de syntaxe dans une commande causée par une séquence incorrecte de paramètres de commande ou l'absence de certaines valeurs de paramètre de commande requises.

Passez en revue et corrigez les paramètres de commande et la séquence dans laquelle ils sont indiqués.

La commande ***`COMMAND_NAME`*** **`n'est pas prise en charge par`** ***`SERVICE_TYPE`*** EPM Automate n'est pas parvenu à exécuter la commande dans l'environnement auquel vous êtes connecté car le processus métier ne prend pas en charge la commande.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Reportez-vous à la section Référence de commande pour obtenir les listes de commandes prises en charge par chaque processus métier.

`Le fichier n'exsite pas dans l'emplacement : PATH` EPM Automate n'est pas parvenu à trouver le fichier à traiter, à l'aide de la commande `upload` ou `replay` par exemple.

Assurez-vous que le nom et le chemin de fichier sont exacts.

`Impossible d'ouvrir le fichier pour lecture : PATH` EPM Automate n'a pas pu procéder à la lecture à partir du fichier indiqué.

Assurez-vous que le fichier est au format requis. Vérifiez que l'utilisateur exécutant EPM Automate dispose d'un accès en lecture au fichier.

`Impossible d'ouvrir le fichier pour écriture : PATH` EPM Automate n'est pas parvenu à écrire dans le fichier indiqué.

Assurez-vous que le fichier n'est pas verrouillé par un autre processus. Vérifiez que l'utilisateur exécutant EPM Automate dispose d'un accès en écriture au fichier.

`Invalid command` EPM Automate a détecté une commande non prise en charge.

Vérifiez qu'EPM Automate prend en charge la commande. Assurez-vous également que le nom de la commande est orthographié correctement.

`Invalid date format` L'outil a détecté un format de date non valide.

Indiquez les dates de génération de rapport dans un format de date pris en charge.

`FROMDATE DATE ne peut pas être postérieur à TODATE DATE` EPM Automate a détecté une date de fin antérieure à la date de début.

Assurez-vous que `to date` dans une plage de dates indiquée est une date antérieure à `from date`.

`Exceeded maximum number of feedbacks (6) for a day` Cette erreur est affichée lorsque vous dépassez le nombre de commentaires que vous pouvez soumettre à l'aide de la commande `feedback`.

`Un fichier portant le même nom existe déjà dans le chemin de téléchargement PATH`. Supprimez le fichier et réexécutez la commande Cette erreur est affichée lorsque vous essayez de télécharger un fichier dans un emplacement qui contient déjà un fichier portant le même nom.

Supprimez, renommez ou déplacez le fichier existant, puis réexécutez la commande.

`Le fichier est vide : PATH` Cette erreur est affichée si le fichier de réexécution n'a aucun contenu.

Assurez-vous que le fichier de réexécution (fichier CSV) répertorie les informations d'identification (nom d'utilisateur et mot de passe) et le nom des fichiers HAR qui doivent être utilisés pour exécuter la commande `replay`.

`Impossible de crypter le mot de passe car localhost ne peut pas être résolu. Assurez-vous que les noms d'hôte sont mappés correctement avec les adresses IP` Cette erreur est affichée si EPM Automate ne parvient pas à résoudre la définition de localhost en une adresse MAC car le fichier hosts sur votre ordinateur contient un nom de serveur au lieu de `localhost` pour l'adresse `127.0.0.1`.

Assurez-vous que le fichier d'hôtes indique `localhost` comme nom de serveur pour `127.0.0.1`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

`Snapshot Name is invalid` Cette erreur est affichée si vous n'indiquez pas le nom de l'instantané à renommer.

Indiquez le nom d'un instantané disponible dans l'environnement.

`New Snapshot Name is invalid` Cette erreur est affichée si vous n'indiquez pas de nouveau nom pour l'instantané.

Indiquez le nouveau nom de l'instantané.

`Nom d'instantané non valide : {0}. Les caractères \\/*?"<>| ne sont pas autorisés` Cette erreur est affichée si le nom de l'instantané contient des caractères spéciaux tels que les suivants : espace, \ (barre oblique inverse), / (barre oblique), * (astérisque), ? (point d'interrogation), " (guillemet), < (inférieur à) et > (supérieur à).

Spécifiez un nouveau nom d'instantané qui ne contient pas ces caractères spéciaux.

`Impossible de renommer l'instantané : {0}. Il est possible qu'un autre processus l'utilise. Réessayez ultérieurement` Cette erreur est affichée si EPM Automate ne peut pas obtenir de verrou exclusif sur l'instantané car ce dernier est déjà en cours d'utilisation par un autre processus.

Attendez que l'opération en cours qui utilise l'instantané se termine, puis réessayez.

`L'instantané {0} existe déjà. Supprimez l'instantané et réexécutez la commande` Cette erreur est affichée si le nom du nouvel instantané est identique à celui d'un instantané existant dans l'environnement.

Utilisez un nom d'instantané différent ou supprimez l'instantané existant à l'aide de la commande `deletefile`.

Erreurs du code de sortie 9

`Informations d'identification non valides` Cette erreur est affichée lorsque le nom d'utilisateur ou le mot de passe employé avec la commande login n'est pas correct.

Indiquez des informations d'identification valides pour l'environnement auquel vous essayez de vous connecter.

`Echec de l'authentification lors de l'exécution de la commande. Veuillez réessayer` Cette erreur est affichée lorsque l'authentification de base échoue lors de l'exécution d'une commande autre que login. Cette erreur peut également se produire pour des appels HTTP en cas de nouvelle tentative d'exécution d'une commande (jusqu'à trois fois).

Erreurs du code de sortie 11

Erreur de serveur interne. En raison de la réinitialisation manuelle du service, votre environnement Oracle EPM Cloud Service est actuellement indisponible. Cette erreur est affichée si les commandes EPM Automate sont exécutées pendant la réinitialisation de l'environnement.

`Internal server error MESSAGE` Cette erreur est affichée si EPM Automate rencontre des exceptions inconnues non liées aux connexions HTTP. Inclut les erreurs de serveur 503 et 500.

`Unable to connect to URL: MESSAGE` Cette erreur est affichée lorsque le serveur n'est pas disponible. Le message d'erreur indique l'exception ayant provoqué l'échec de la commande.

Si le serveur n'est pas disponible, contactez le support technique Oracle. Si le message indique des problèmes avec l'URL, vérifiez que l'URL que vous utilisez est valide.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

3

Scénarios d'exemple d'exécution de commandes

EPM Automate peut être utilisé pour automatiser de nombreuses tâches d'administration Oracle Fusion Cloud Enterprise Performance Management courantes.

- Exemples de scénarios pour tous les services
- Exemples de scénarios pour Planning, Consolidation, Tax Reporting et Enterprise Profitability and Cost Management
- Exemples de scénarios pour Account Reconciliation
- Exemples de scénarios pour Profitability and Cost Management
- Exemples de scénarios pour Cloud EDM

A propos de la copie d'exemples de script

Ne copiez pas les exemples de script à partir de la version PDF de ce document. Pour éviter les retours à la ligne et les informations de pied de page qui rendent les scripts inutilisables, Oracle vous recommande de copier les exemples de script à partir de la version HTML de *Utilisation d'EPM Automate*.

Exemples de scénarios pour tous les services

Ces scénarios présentent une séquence classique de commandes pouvant être utilisées pour réaliser des opérations spécifiques dans les environnements Oracle Fusion Cloud Enterprise Performance Management.

Voir aussi :

- Sauvegarde d'un instantané d'application sur un ordinateur
  Ce scénario explique comment automatiser le processus de sauvegarde de l'instantané créé lors de la maintenance de service quotidienne sur un ordinateur local.
- Communication de la fin de la maintenance quotidienne aux utilisateurs
  La maintenance quotidienne des environnements Oracle Fusion Cloud Enterprise Performance Management prend généralement bien moins de temps que l'heure qui lui est affectée.
- Copie d'un instantané à partir d'Oracle Object Storage ou vers celui-ci
- Création d'utilisateurs et affectation de rôles prédéfinis
  Utilisez les scripts de cette section pour créer des utilisateurs et les affecter à des rôles prédéfinis du domaine d'identité.
- Décompte des utilisateurs disposant d'une licence (utilisateurs affectés à des rôles)
  Utilisez le script de cette section pour générer le rapport sur l'affectation de rôle permettant de compter le nombre d'utilisateurs dans un environnement.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Création de rapports d'audit sur les utilisateurs affectés à des rôles
  Utilisez les scripts de cette section afin d'automatiser le processus de création d'un rapport d'audit pour les utilisateurs affectés à des rôles prédéfinis dans un environnement, et éventuellement afin d'envoyer le rapport par courriel à un destinataire.
- Création d'un rapport d'audit sur la révocation et l'affectation de rôle
  Utilisez le script PowerShell de cette section pour automatiser le processus de création d'un rapport d'audit qui décrit en détail l'affectation et la révocation de rôle dans un environnement.
- Masquage des journaux d'accès et du rapport d'activité pour respecter les lois sur la confidentialité
  Utilisez les scripts de cette section pour automatiser le processus de masquage d'informations dans le rapport d'activité ou les journaux d'accès afin de respecter les lois sur la confidentialité, et éventuellement pour envoyer le rapport par courriel à un destinataire.
- Automatisation des téléchargements de rapport d'activité sur un ordinateur local
  Utilisez le script de cette section pour automatiser le téléchargement des rapports d'audit à partir d'un environnement vers un ordinateur local.
- Téléchargement des journaux d'accès à partir d'un environnement
  Utilisez le script de cette section pour automatiser le processus de téléchargement des journaux d'accès à partir d'un environnement vers un ordinateur local.
- Automatisation du clonage d'environnements
  Utilisez le script de cette section pour automatiser le clonage des environnements.
- Clonage quotidien de l'environnement principal vers l'environnement de secours une fois la maintenance quotidienne de l'environnement principal effectuée
  Pour que l'environnement de secours reste à jour par rapport à l'environnement principal, utilisez ces scripts afin de cloner l'environnement principal Oracle Fusion Cloud Enterprise Performance Management vers l'environnement de secours peu après la fin de la maintenance quotidienne de l'environnement principal.
- Suppression des fichiers inutiles d'un environnement
  Utilisez ces scripts pour enlever les fichiers inutiles d'un environnement.
- Recherche et téléchargement de fichiers à partir d'un environnement
  Utilisez l'exemple de script de cette section pour automatiser le processus de téléchargement de fichiers à partir d'un environnement Oracle Fusion Cloud Enterprise Performance Management à l'aide d'une chaîne de texte comme caractère générique.
- Recréation d'un ancien environnement Cloud EPM pour les audits
  Utilisez le script de cette section pour créer une solution libre-service permettant de maintenir une bibliothèque d'instantanés à jour pour votre environnement Oracle Fusion Cloud Enterprise Performance Management. Vous avez besoin d'un environnement dédié à la mise à niveau et à la maintenance d'une bibliothèque d'instantanés à jour.
- Automatisation de l'audit et de la conformité de l'accès à la base de données
  Les scripts PowerShell et d'interpréteur de commandes Bash de cette section vous permettent d'exploiter les commandes EPM Automate en vue de collecter des données d'audit et de conformité concernant l'accès manuel à la base de données.
- Réplication des utilisateurs et des affectations de rôle prédéfini
  Les scripts de cette section vous aident à migrer les utilisateurs et les affectations de rôle prédéfini d'un environnement vers un autre.
- Création d'une cadence de mise à niveau Cloud EPM trimestrielle
  Utilisez ces scripts pour créer une solution libre-service permettant d'ignorer les mises à jour de sorte que les environnements Cloud EPM soient mis à jour chaque trimestre avec

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

un cycle de test de deux semaines. Dans ce cas, les environnements de production sont mis à jour deux semaines après les environnements de test.

- Création d'une cadence de mise à niveau Cloud EPM trimestrielle avec des cycles de test de six semaines
  Utilisez le script de cette section pour créer une solution libre-service permettant d'ignorer les mises à jour de sorte que les environnements Oracle Fusion Cloud Enterprise Performance Management soient mis à jour chaque trimestre avec un cycle de test de six semaines. Dans ce cas, les environnements de production sont mis à jour six semaines après les environnements de test.

Sauvegarde d'un instantané d'application sur un ordinateur

Ce scénario explique comment automatiser le processus de sauvegarde de l'instantané créé lors de la maintenance de service quotidienne sur un ordinateur local.

- Télécharge l'instantané d'application (`Artifact Snapshot`) créé lors de la fenêtre de maintenance.
- Renomme l'instantané téléchargé en ajoutant l'horodatage.
- Tient à jour 10 sauvegardes en supprimant la sauvegarde la plus ancienne, si nécessaire.

> **Note**
>
> - Ce script ne peut pas être utilisé pour sauvegarder Narrative Reporting
> - Si vous réadaptez ce script à votre utilisation personnelle, modifiez les valeurs des paramètres d'exécution (`url`, `user`, `password` et `NumberOfBackups`) à votre convenance.

Reportez-vous à la section Automatisation de l'exécution de scripts pour plus d'informations sur la planification du script à l'aide du planificateur de tâches Windows.

Exemple de script Windows

Créez un fichier batch (`.bat`) ou d'interpréteur de commandes (`.sh`) qui contient un script semblable aux scripts suivants pour automatiser les téléchargements d'instantanés.

```
@echo off
rem Sample script to download and maintain 10 maintenance backups
rem Update the following  parameters

SET url=https://example.oraclecloud.com
SET user=ServiceAdmin
SET password=Example.epw
SET SnapshotName="Artifact Snapshot"
SET NumberOfBackups=10

rem EPM Automate commands
call epmautomate login %user% %password% %url%
     IF %ERRORLEVEL% NEQ 0 goto :ERROR
       call epmautomate downloadfile %SnapshotName%
     IF %ERRORLEVEL% NEQ 0 goto :ERROR
       call epmautomate logout
     IF %ERRORLEVEL% NEQ 0 goto :ERROR
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
rem Rename downloaded Artifact Snapshot, keep the last 10 backups
Set Timestamp=%date:~4,2%_%date:~7,2%_%date:~10,2%%
Set Second=%time:~0,2%%time:~3,2%
ren %SnapshotName%.zip %SnapshotName%_%Timestamp%_%Second%.zip

SET Count=0
FOR %%A IN (%SnapshotName%*.*) DO SET /A Count += 1
IF %Count% gtr %NumberOfBackups% FOR %%A IN (%SnapshotName%*.*) DO del "%%A"
&& GOTO EOF
:EOF

echo Scheduled Task Completed successfully
exit /b %errorlevel%
:ERROR
echo Failed with error #%errorlevel%.
exit /b %errorlevel%
```

Exemple de script Linux/UNIX

Créez un fichier d'interpréteur de commandes (`.sh`) contenant un script semblable à ce qui suit pour automatiser les téléchargements d'instantanés. Si votre mot de passe contient des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux.

```
#!/bin/sh
# Sample script to download and maintain 10 maintenance backups
# Update the following seven parameters

url=https://example.oraclecloud.com
user=serviceAdmin
password=/home/user1/epmautomate/bin/example.epw
snapshotname="Artifact Snapshot"
numberofbackups=10
epmautomatescript=/home/user1/epmautomate/bin/epmautomate.sh
javahome=/home/user1/jdk_17.0.12/

export JAVA_HOME=${javahome}

printResult()
    {
    op="$1"
    opoutput="$2"
    returncode="$3"

    if [ "${returncode}" -ne 0 ]
    then
        echo "Command failed. Error code: ${returncode}. ${opoutput}"
    else
        echo "${opoutput}"
    fi
}

processCommand()
{
    op="$1"
    date=`date`
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    echo "Running ${epmautomatescript} ${op}"
    operationoutput=`eval "$epmautomatescript $op"`
    printResult "$op" "$operationoutput" "$?"
}

op="login ${user} ${password} ${url}"
processCommand "${op}"

op="downloadfile \"${snapshotname}\""
processCommand "${op}"

op="logout"
processCommand "${op}"

# Renames the downloaded artifacts, keeps the last 10 backups
timestamp=`date +%m_%d_%Y_%I%M`
mv "${snapshotname}.zip" "${snapshotname}_${timestamp}.zip"

((numberofbackups+=1))
ls -tp ${snapshotname}*.zip | grep -v '/$' | tail -n +${numberofbackups} |
xargs -d '\n' -r rm --
```

Communication de la fin de la maintenance quotidienne aux utilisateurs

La maintenance quotidienne des environnements Oracle Fusion Cloud Enterprise Performance Management prend généralement bien moins de temps que l'heure qui lui est affectée.

La durée de maintenance quotidienne réelle de l'environnement est enregistrée comme valeur de la métrique Durée de la maintenance quotidienne en minutes dans la section Métriques opérationnelles du rapport d'activité. Pour éviter que les utilisateurs n'attendent une heure entière avant de travailler dans l'environnement, recourez à une version personnalisée de ce script afin de les informer de la fin de la maintenance quotidienne et de leur permettre de reprendre leurs activités.

Script Windows

Créez `daily_maintenance_completed.ps1` en copiant le script PowerShell suivant. Reportez-vous à la section Exécution du script afin d'obtenir des informations sur la mise à jour du script pour votre utilisation.

```
# Daily Maintenance Completed Notification script
#
# Update the following parameters
# -------------------------------
$emailaddresses=user1@oracle.com,user2@oracle.com
# -------------------------------

$username=$args[0]
$password=$args[1]
$url=$args[2]

if ($($args.count) -ne 3) {
    echo "Usage: ./daily_maintenance_completed.ps1 <USERNAME> <PASSWORD>
<URL>"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    exit 1
}

$amw_time=""

function getDailyMaintenanceStartTime {
    $amwstring=$(epmautomate.bat getDailyMaintenanceStartTime)
    $elements=$amwstring.split(' ')
    $amwtime=$elements[0]
    return $amwtime
}

function goToSleep ($amw_time){
    $current_mdy=Get-Date -AsUTC -UFormat "%m/%d/%Y"
    $current_date_time=Get-Date -AsUTC -UFormat "%m/%d/%Y %H:%M:%S"
    $current_epoch=Get-Date -Date $current_date_time -UFormat "%s"
    $target_date_time=[DateTime]"${current_mdy} ${amw_time}"
    $target_epoch=Get-Date -Date $target_date_time -UFormat "%s"
    $sleep_seconds=$target_epoch - $current_epoch

    # Today's AMW start time has already passed, so add 24 hours to
sleep_seconds
    if ($sleep_seconds -lt 0) {
        $sleep_seconds=$sleep_seconds + 86400
    }

    $sleep_ts=New-TimeSpan -Seconds ${sleep_seconds}
    $sleep_hms="${sleep_ts}" -replace '^\d+?\.'

    echo "Current time is ${current_date_time}. Sleeping for ${sleep_hms},
until daily maintenance start time of ${amw_time}."
    Start-Sleep -Seconds $sleep_seconds
}

function attemptLogin {
    $serverdown=$False
    while ($true) {
        epmautomate.bat login ${username} ${password} ${url}
        if ($?) { # login succeeded
        if ($serverdown) { # server has been brought down
                echo "Daily maintenance processing has completed ..."
                break
        } else { # server has not yet been brought down
                echo "Daily maintenance processing has not yet started.
Sleeping for 2 minutes before the next check ..."
                Start-Sleep -Seconds 120
            }
    } else { # login failed
        if ($serverdown) { # server has been brought down
                echo "Waiting for daily maintenance processing to complete.
Sleeping for 2 minutes before the next check ..."
                Start-Sleep -Seconds 120
        } else { # server has not yet been brought down
                echo "Daily maintenance processing is now beginning. Sleeping
for 2 minutes before the next check ..."
                Start-Sleep -Seconds 120
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
            $serverdown=$True
        }
        }
    }
}

function sendNotification {
    $servername=$url.split("/")[2];
    $subject="Daily maintenance processing has completed"
    $formattedmessage="Daily maintenance processing has completed for server $
{servername}"
    $emailaddresses=${emailaddresses}.replace(',',';')

    echo "Mailing report"
    epmautomate.bat sendmail "${emailaddresses}" "${subject}" Body="$
{formattedmessage}"
}

echo "Beginning daily maintenance completion notification script."
echo "Logging into server ..."
epmautomate.bat login ${username} ${password} ${url}
$amwtime=getDailyMaintenanceStartTime
goToSleep ($amwtime)
attemptLogin
sendNotification
echo "Logging out of server ..."
epmautomate.bat logout
echo "Script processing has completed."
```

Script Linux/UNIX

Créez `daily_maintenance_completed.sh` en copiant le script suivant. Reportez-vous à la section Exécution du script afin d'obtenir des informations sur la mise à jour du script pour votre utilisation.

```
#!/bin/bash
# Update the following parameters
# -------------------------------
epmautomatescript="LOCATION_EPM_AUTOMATE_EXECUTABLE"
javahome="LOCATION_JAVA_HOME"
emailaddresses=EMAIL_ADDRESS_1,EMAIL_ADDRESS_2,EMAIL_ADDRESS_N
# -------------------------------

username="$1"
password="$2"
url="$3"

export JAVA_HOME=${javahome}

if [ "$#" -ne 3 ]; then
    echo "Usage: ./daily_maintenance_completed.sh <USERNAME> <PASSWORD> <URL>"
    exit 1
fi

amw_time=""
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
getDailyMaintenanceStartTime() {
    amw_time=$(${epmautomatescript} getDailyMaintenanceStartTime | cut -d' ' -
f1)
}

goToSleep() {
    current_mdy=$(date -u +%m/%d/%Y)
    current_date_time=$(date -u)
    current_epoch=$(date +%s)
    target_epoch=$(date -d "${current_mdy} ${amw_time}" +%s)
    sleep_seconds=$(($target_epoch - $current_epoch))

    # Today's AMW start time has already passed, so add 24 hours to
sleep_seconds
    if [[ ${sleep_seconds} -lt 0 ]]
    then
        sleep_seconds=$((sleep_seconds + 86400))
    fi

    sleep_hms=$(date -d@${sleep_seconds} -u +%H:%M:%S)

    echo "Current time is ${current_date_time}. Sleeping for ${sleep_hms},
until daily maintenance start time of ${amw_time}."
    sleep $sleep_seconds
}

attemptLogin() {
    local serverdown=1
    while true
    do
        ${epmautomatescript} login ${username} ${password} ${url}
        if [[ $? -eq 0 ]] # login succeeded
        then
            if [[ ${serverdown} -eq 0 ]] # server has been brought down
            then
                echo "Daily maintenance processing has completed"
                break
            else # server has not yet been brought down
                echo "Daily maintenance processing has not yet started.
Sleeping for 2 minutes before the next check ..."
                sleep 120
            fi
        else # login failed
            if [[ ${serverdown} -eq 0 ]] # server has been brought down
            then
                echo "Waiting for daily maintenance processing to complete.
Sleeping for 2 minutes before the next check ..."
                sleep 120
            else # server has not yet been brought down
                echo "Daily maintenance processing is now beginning. Sleeping
for 2 minutes before the next check ..."
                sleep 120
                serverdown=0
            fi
        fi
    done
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
}

sendNotification()
{
    local servername=$(echo "${url}" | cut -d '/' -f3- | rev | cut -d':' -f2-
| rev)
    local subject="Daily maintenance processing has completed"
    local formattedmessage="Daily maintenance processing has completed for
server ${servername}"
    local emailaddresses=$(echo ${emailaddresses} | sed "s/,/;/g")

    echo "Mailing report"
    ${epmautomatescript} sendmail "${emailaddresses}" "${subject}" Body="$
{formattedmessage}"
}

echo "Beginning daily maintenance completion notification script."
echo "Logging into server ..."
${epmautomatescript} login ${username} ${password} ${url}
getDailyMaintenanceStartTime
goToSleep
attemptLogin
sendNotification
echo "Logging out of server ..."
${epmautomatescript} logout
echo "Script processing has completed."
```

Script Groovy côté serveur

Créez le script Groovy `daily_maintenance_completed` en copiant le code suivant. Reportez-vous à la section Exécution du script afin d'obtenir des informations sur la mise à jour du script pour votre utilisation.

```
// Daily Maintenance Completed Notification script

// Update the following parameters
// ------------------------------
String username="USERNAME"
String password="PASSWORD"
String url="URL OF THE ENVIRONMENT"
String emailaddresses="EMAIL_ADDRESS_1,EMAIL_ADDRESS_2,EMAIL_ADDRESS_N"
// ------------------------------

def LogMessage(String message) {
    def date = new Date()
    def sdf = new SimpleDateFormat("MM/dd/yyyy HH:mm:ss")
    println('[' + sdf.format(date) + '] ' + message);
}

def LogOperationStatus(EpmAutomateStatus opstatus) {
    def returncode = opstatus.getStatus()
    if (returncode != 0){
        LogMessage(opstatus.getOutput())
    }
    LogMessage('return code: ' + returncode)
}
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
def getDailyMaintenanceStartTime(EpmAutomate automate) {
    LogMessage("Operation: getDailyMaintenanceStartTime")
    EpmAutomateStatus amwtimestatus =
automate.execute('getDailyMaintenanceStartTime')
    LogOperationStatus(amwtimestatus)
    def amwstring=(amwtimestatus.getOutput())
    def elements=amwstring.split(' ')
    def amwtime=elements[0]
    return amwtime
}

def goToSleep(String amw_time){
    def date = new Date()
    def current_mdy = new SimpleDateFormat("MM/dd/yyyy")
    def current_date_time = new SimpleDateFormat("MM/dd/yyyy HH:mm:ss")
    float current_epoch = date.getTime() / 1000
    def pattern = "MM/dd/yyyy HH:mm:ss"
    def input = current_mdy.format(date) + " " + amw_time + ":00"
    def target_date_time = Date.parse(pattern, input)
    float target_epoch = target_date_time.getTime() / 1000
    int sleep_seconds = Math.round(target_epoch - current_epoch)

    //Today's AMW start time has already passed, so add 24 hours to
sleep_seconds
    if (sleep_seconds < 0) {
        sleep_seconds = sleep_seconds + 86400
    }

    def sleep_milliseconds = sleep_seconds * 1000
    LogMessage("Current time is " + current_date_time.format(date) + ".
Sleeping until daily maintenance start time of " + amw_time + ":00.")
    sleep(sleep_milliseconds)
}

def attemptLogin(EpmAutomate automate, String username, String password,
String url) {
    def serverdown=1
    while (true) {
        LogMessage("Operation: login " + username + " " + password + " " +
url)
        EpmAutomateStatus status =
automate.execute('login',username,password,url)
        def returncode = status.getStatus()
        if (returncode == 0) {
            if (serverdown == 0){
                LogMessage("Daily maintenance processing has completed ...")
                break
            } else {
                LogMessage("Daily maintenance processing has not yet started.
Sleeping for 2 minutes before the next check ...")
                sleep(120000)
            }
        } else {
            if (serverdown == 0){
                LogMessage("Waiting for daily maintenance processing to
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
complete. Sleeping for 2 minutes before the next check ...")
                sleep(120000)
            } else {
                LogMessage("Daily maintenance processing is now beginning.
Sleeping for 2 minutes before the next check ...")
                sleep(120000)
                serverdown=0
            }
        }
    }
}

def sendNotification(EpmAutomate automate, String url, String emailaddresses)
{
    def servername=url.tokenize("/")[-1];
    def subject="Daily maintenance processing has completed"
    def formattedmessage="Daily maintenance processing has completed for
server " + servername
    def emailaddressesformatted = emailaddresses.replaceAll(',',';')

    LogMessage("Operation: sendmail " + emailaddressesformatted + " " +
subject + " Body=" + formattedmessage)
    EpmAutomateStatus status =
automate.execute('sendmail',emailaddressesformatted,subject,'Body=' +
formattedmessage)
    LogOperationStatus(status)
}

LogMessage("Beginning daily maintenance completion notification script.")

EpmAutomate automate = getEpmAutomate()

LogMessage("Operation: login " + username + " " + password + " " + url)
EpmAutomateStatus status = automate.execute('login',username,password,url)
LogOperationStatus(status)

String amwtime = getDailyMaintenanceStartTime(automate)
goToSleep (amwtime)
attemptLogin(automate,username,password,url)
sendNotification(automate,url,emailaddresses)

LogMessage("Operation: logout ")
status = automate.execute('logout')
LogOperationStatus(status)

LogMessage ("Script processing has completed.")
```

Exécution du script

Windows et Linux/UNIX

1. Créez `daily_maintenance_completed.ps1` ou `daily_maintenance_completed.sh` en copiant le script d'une section précédente.
2. Mettez le script à jour :

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- **Windows :** mettez à jour la valeur d'`emailaddresses` avec la liste des adresses électroniques (séparées par une virgule) des utilisateurs à avertir une fois la maintenance quotidienne terminée.
- **Linux/UNIX :** mettez à jour les variables suivantes :
  - `epmautomatescript` avec l'emplacement de l'exécutable EPM Automate. Exemple : `epmautomatescript="/home/utils/EPMAutomate/bin/epmautomate.sh"`
  - `javahome` avec le répertoire dans lequel le JDK utilisé par EPM Automate est installé. Par exemple : `"/home/user1/jdk_17.0.12"`
  - `emailaddresses` avec la liste des adresses électroniques (séparées par une virgule) des utilisateurs à avertir une fois la maintenance quotidienne terminée. Par exemple : `jdoe@example.com,jane_doe@example.com`

3. Dans une console ou une fenêtre de commande, accédez au dossier de stockage du script `daily_maintenance_completed`.
4. Exécutez la commande suivante :
   - **Windows :** `./daily_maintenance_completed.ps1 USERNAME PASSWORD URL`
   - **Linux/UNIX :** `./daily_maintenance_completed.sh USERNAME PASSWORD URL`, où :
     - `USERNAME` est le nom d'utilisateur d'un administrateur de service
     - `PASSWORD` est le mot de passe de l'administrateur de service
     - `URL` est l'URL de l'environnement Cloud EPM

Groovy côté serveur :

1. Créez le script Groovy `daily_maintenance_completed.groovy` en le copiant à partir d'une section précédente.
2. Mettez à jour les valeurs suivantes.
   - `username` avec le nom d'utilisateur d'un administrateur de service.
   - `password` avec le mot de passe de l'administrateur de service.
   - `url` avec l'URL de l'environnement Cloud EPM pour lequel la notification de fin de maintenance quotidienne doit être effectuée. Par exemple : Exemple : `https://testExample-idDomain.pbcs.us1.oraclecloud.com`
   - `emailaddresses` avec la liste des adresses électroniques (séparées par une virgule) des utilisateurs à avertir une fois la maintenance quotidienne terminée.
3. Utilisez l'écran Groovy dans un processus métier Cloud EPM ou automatisez l'exécution du script via runBusinessRule. Pour plus d'informations, consultez ces sources d'information :
   - Exécution de commandes sans installer EPM Automate
   - Utilisation de règles Groovy dans le guide *Administration de Planning*

Copie d'un instantané à partir d'Oracle Object Storage ou vers celui-ci

Cette rubrique contient des exemples de script permettant d'effectuer les tâches suivantes :

- Copier l'instantané `Artifact Snapshot` (instantané de maintenance) d'Oracle Fusion Cloud Enterprise Performance Management vers un bucket Oracle Object Storage et le renommer en ajoutant la date à laquelle l'instantané a été copié.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Copier un instantané de sauvegarde d'un bucket Oracle Object Storage vers Cloud EPM.

Les scripts de cette section supposent que vous avez déjà créé un bucket dans Oracle Object Storage pour contenir l'instantané. Avant d'exécuter ces scripts, personnalisez-les en mettant à jour les espaces réservés suivants :

Table 3-1 Paramètres et leurs valeurs

Espace réservé	Valeur attendue
`JAVA_HOME`	Répertoire dans lequel le JDK utilisé par EPM Automate est installé. **Exemple :** `./home/JDK/bin`
`epmautomateExe`	Répertoire dans lequel EPM Automate est installé. **Exemple :** `./home/utils/EPMAutomate/bin`
`cloudServiceUser`	ID utilisateur d'un administrateur de service Cloud EPM. **Exemple :** `John.doe@example.com`
`cloudServicePassword`	Mot de passe de l'administrateur de service ou emplacement du fichier de mots de passe. Si le mot de passe contient des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux. **Exemple :** `ex_PWD_213`
`cloudServiceUrl`	URL de l'environnement Cloud EPM à partir duquel l'instantané `Artifact Snapshot` doit être copié. **Exemple :** `https//test-cloud-id_Dom.pbcs.us1.oraclecloud.com`
`objectStorageUser`	ID d'un utilisateur dans Oracle Object Storage. Pour copier un instantané vers Object Storage, cet utilisateur doit avoir accès en écriture au bucket dans lequel l'instantané est copié. Pour copier un instantané à partir d'Object Storage, cet utilisateur doit avoir accès en lecture au bucket à partir duquel l'instantané est copié. **Exemple :** `jDoe`
`objectStoragePassword`	Mot de passe pour `objectStorageUser`. **Exemple :** `example_PWD`
`objectStorageBucketUrl`	URL du bucket Oracle Object Storage dans lequel l'instantané doit être copié. Reportez-vous aux sources d'informations ci-dessous concernant le format de l'URL : • copyToObjectStorage • copyFromObjectStorage **Exemple :** `https//swiftobjectstorage.us-ashburn-1.oraclecloud.com/v1/axaxnpcrorw5/bucket-20210301-1359`
`snapshot`	Nom de l'instantané que vous voulez copier à partir du bucket Oracle Object Storage. **Exemple :** `Artifact Snapshot20210429.zip`

Exemple de script EPM Automate pour copier un instantané de Cloud EPM vers Oracle Object Storage

Personnalisez et exécutez ce script pour renommer l'instantané `Artifact Snapshot`, puis le copier de Cloud EPM vers un bucket Oracle Object Storage.

```
#!/bin/sh
export JAVA_HOME=<path_to_jdk>
epmautomateExe=<path_to_epmautomate_executable>
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
cloudServiceUser=<cloud_service _user>
cloudServicePassword=<cloud_service_password>
cloudServiceUrl=<cloud_service_url>
# User with write access to Object Storage bucket
objectStorageUser=<object_storage_user>
objectStoragePassword=<object_storage_password>
objectStorageBucketUrl=<object_storage_bucket>
currentDate=`date +'%Y%m%d'`
sourceSnapshot="Artifact Snapshot"
targetSnapshot="${sourceSnapshot} ${currentDate}"
$epmautomateExe login ${cloudServiceUser} ${cloudServicePassword} $
{cloudServiceUrl}
$epmautomateExe renamesnapshot "${sourceSnapshot}" "${targetSnapshot}"
$epmautomateExe copyToObjectStorage "${targetSnapshot}" ${objectStorageUser} $
{objectStoragePassword} "${objectStorageBucketUrl}/${targetSnapshot}"
$epmautomateExe logout
exit 0
```

Exemple de script EPM Automate pour copier un instantané d'Oracle Object Storage vers Cloud EPM

Personnalisez et exécutez ce script pour copier l'instantané `Artifact Snapshot` d'une date spécifique d'un bucket Oracle Object Storage vers Cloud EPM.

```
#!/bin/sh
export JAVA_HOME=<path_to_jdk>
epmautomateExe=<path_to_epmautomate_executable>
cloudServiceUser=<cloud_service _user>
cloudServicePassword=<cloud_service_password>
cloudServiceUrl=<cloud_service_url>
# User with read access to Object Storage bucket
objectStorageUser=<object_storage_user>
objectStoragePassword=<object_storage_password>
objectStorageBucketUrl=<object_storage_bucket>
snapshot=<desired_snapshot>
$epmautomateExe login ${cloudServiceUser} ${cloudServicePassword} $
{cloudServiceUrl}
$epmautomateExe copyFromObjectStorage ${objectStorageUser} $
{objectStoragePassword} "${objectStorageBucketUrl}/${snapshot}"
$epmautomateExe logout
exit 0
```

Création d'utilisateurs et affectation de rôles prédéfinis

Utilisez les scripts de cette section pour créer des utilisateurs et les affecter à des rôles prédéfinis du domaine d'identité.

Les scripts utilisent des commandes EPM Automate pour effectuer les activités suivantes :

- Connectez-vous à l'environnement en tant qu'administrateur de service disposant du rôle Administrateur de domaine d'identité.
- Exportez les informations sur les groupes et l'appartenance à partir de l'environnement afin de régénérer l'instantané indiqué, par exemple : `example_snapshot.zip`. Nous partons du principe que vous avez précédemment exporté les artefacts de groupe et d'appartenance à l'aide de Migration pour créer l'instantané.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Téléchargez l'instantané (`example_snapshot.zip`) vers un répertoire local.
- Supprimez l'instantané (`example_snapshot.zip`) de l'environnement.
- Déconnectez-vous de l'environnement.
- Effectuez les opérations pour lesquelles vous avez ajouté du code personnalisé. Il peut s'agir des opérations suivantes :
  - Extraction du contenu d'`example_snapshot.zip`.
  - Ajout d'informations sur les nouveaux utilisateurs dans `HSS-Shared Services\resource\External Directory\Users.csv` au format Prénom,Nom,Adresse électronique,Connexion utilisateur. Pour obtenir des informations détaillées, reportez-vous à la section Import d'un lot de comptes utilisateur dans le guide *Mise en route d'Oracle Cloud*.
  - Ajout d'informations sur les affectations de rôle des nouveaux utilisateurs (au format Prénom,Nom,Adresse électronique,Connexion utilisateur) aux fichiers de rôle appropriés dans `HSS-Shared Services\resource\External Directory\Roles\`. Par exemple, l'affectation au rôle Administrateur de service doit être ajoutée au fichier `<service_name> Service Administrator.csv` alors que les affectations au rôle Visualiseur doivent être ajoutées à `HSS-Shared Services\resource\External Directory\Roles\<service_name> Viewer.csv`. Pour obtenir des informations détaillées, reportez-vous à la section Affectation d'un rôle à plusieurs utilisateurs dans le guide *Mise en route d'Oracle Cloud*.
  - Recréation de l'instantané avec les fichiers mis à jour via la compression du répertoire `HSS-Shared Services` et de son contenu.
- Connectez-vous à l'environnement en tant qu'administrateur de service disposant également du rôle Administrateur de domaine d'identité.
- Téléchargez le fichier modifié `example_snapshot.zip` vers l'environnement.
- Importez le fichier `example_snapshot.zip` vers l'environnement.
- Supprimez le fichier `example_snapshot.zip` téléchargé de l'environnement.
- Déconnectez-vous.

Exemple de script Windows

Créez un fichier nommé `createUsersAndAssignRoles.ps1` en copiant le script suivant. Stockez-le dans un répertoire local.

```
$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -raw)
$username="$($inputproperties.username)"
$passwordfile="$($inputproperties.passwordfile)"
$serviceURL="$($inputproperties.serviceURL)"
$snapshotName="$($inputproperties.snapshotName)"
$userspassword="$($inputproperties.userspassword)"
$resetPassword="$($inputproperties.resetPassword)"

epmautomate login ${username} ${passwordfile} ${serviceURL}
epmautomate exportsnapshot ${snapshotName}
epmautomate downloadfile ${snapshotName}.zip
epmautomate deletefile ${snapshotName}.zip
epmautomate logout

# Add custom code to extract the contents of example_snapshot.zip
# Add custom code to append new user information to HSS-Shared
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
Services\resource\External Directory\Users.csv
# Optional: Add custom code to append role information to the appropriate
role file(s) in HSS-Shared Services\resource\External Directory\Roles\
# Add custom code to zip HSS-Shared Services and its contents as
example_snapshot.zip

epmautomate login ${username} ${passwordfile} ${serviceURL}
epmautomate uploadfile ${snapshotName}.zip
epmautomate importsnapshot ${snapshotName} userPassword=${userspassword}
resetPassword=${resetPassword}
epmautomate deletefile ${snapshotName}.zip
epmautomate logout
```

Exemple de script Linux/UNIX

Créez un fichier nommé `createUsersAndAssignRoles.sh` en copiant le script suivant. Stockez-le dans un répertoire local.

```
#!/bin/bash

. ./input.properties
export JAVA_HOME=${javahome}
${epmautomatescript} login "${username}" "${passwordfile}" "${serviceURL}"
${epmautomatescript} exportsnapshot "${snapshotName}"
${epmautomatescript} downloadfile "${snapshotName}.zip"
${epmautomatescript} deletefile "${snapshotName}.zip"
${epmautomatescript} logout

# Add custom code to extract the contents of example_snapshot.zip
# Add custom code to append new user information to HSS-Shared
Services\resource\External Directory\Users.csv
# Optional: Add custom code to append role information to the appropriate
role file(s) in HSS-Shared Services\resource\External Directory\Roles\
# Add custom code to zip HSS-Shared Services and its contents as
example_snapshot.zip

${epmautomatescript} login "${username}" "${passwordfile}" "${serviceURL}"
${epmautomatescript} uploadfile "${snapshotName}.zip"
${epmautomatescript} importsnapshot "${snapshotName}" "userPassword=$
{userspassword}" "resetPassword=${resetPassword}"
${epmautomatescript} deletefile "${snapshotName}.zip"
${epmautomatescript} logout
```

Exemple de fichier input.properties

Pour exécuter les scripts `createUsersAndAssignRoles`, créez le fichier `input.properties` et mettez-le à jour avec les informations de votre environnement. Enregistrez le fichier dans le répertoire dans lequel `createUsersAndAssignRoles.ps1` ou `createUsersAndAssignRoles.sh` est stocké.

Windows

```
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
snapshotName=SNAPSHOT_NAME
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
userspassword=TEMP_IDM_PASSWORD
resetPassword=true
```

Linux/UNIX

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
snapshotName=SNAPSHOT_NAME
userspassword=TEMP_IDM_PASSWORD
resetPassword=true
```

Tableau 3-2 Paramètres du fichier input.properties

Paramètre	Description
`javahome`	Emplacement de `JAVA_HOME`. Pour Linux/UNIX uniquement.
`epmautomatescript`	Chemin absolu de l'exécutable EPM Automate (`epmautomate.sh`). Pour Linux/UNIX uniquement.
`username`	Nom d'utilisateur d'un administrateur de service disposant également du rôle Administrateur de domaine d'identité.
`password`	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
`serviceURL`	URL de l'environnement à partir duquel générer l'instantané.
`snapshotName`	Nom de l'instantané à générer. Nous partons du principe que vous avez précédemment exporté les artefacts de groupe et d'appartenance à l'aide de Migration pour créer l'instantané.
`userspassword`	Mot de passe initial des nouveaux utilisateurs.
`resetPassword`	Indique si les nouveaux utilisateurs doivent réinitialiser le mot de passe lors de la première connexion. Définissez cette valeur sur `true` pour forcer les nouveaux utilisateurs à modifier leur mot de passe lorsqu'ils se connectent pour la première fois.

Exécution du script

1. Créez `createUsersAndAssignRoles.ps1` ou `createUsersAndAssignRoles.sh` en copiant le script d'une section précédente.
2. Ajoutez du code personnalisé pour effectuer les opérations suivantes :
   - Extrayez le contenu de l'instantané.
   - Ajoutez des informations sur les nouveaux utilisateurs dans `HSS-Shared Services\resource\External Directory\Users.csv`.
   - Ajoutez éventuellement des informations sur les affectations de rôle des nouveaux utilisateurs (au format Prénom,Nom,Adresse électronique,Connexion utilisateur) aux fichiers de rôle appropriés dans `HSS-Shared Services\resource\External Directory\Roles\`.
   - Recréez l'instantané avec les fichiers mis à jour.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

3. Créez le fichier `input.properties` et enregistrez-le dans le répertoire où se trouve le script `createUsersAndAssignRoles`. Le contenu de ce fichier varie en fonction du système d'exploitation. Reportez-vous à la section Exemple de fichier input.properties. Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter le script.
4. Lancez le script.
   - **Windows PowerShell :** exécutez `createUsersAndAssignRoles.ps1`.
   - **Linux/UNIX :** exécutez `./createUsersAndAssignRoles.sh`.

Décompte des utilisateurs disposant d'une licence (utilisateurs affectés à des rôles)

Utilisez le script de cette section pour générer le rapport sur l'affectation de rôle permettant de compter le nombre d'utilisateurs dans un environnement.

Créez `provisionedUsersCount.bat` en copiant le script suivant.

> **Remarque**
>
> - Paramètres d'entrée pour l'exécution de `provisionedUsersCount.bat` : `username`, `password/password_file`, `service_url` et `report_file_name`. Par exemple, à l'invite de commande, entrez une commande semblable à ce qui suit : `provisionedUsersCount.bat jdoe password.epw https://example.oraclecloud.com myRole_assign.CSV`
> - Si le mot de passe contient des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux.

```
@echo off

set paramRequiredMessage=Syntax: provisionedUsersCount.bat USERNAME PASSWORD/
PASSWORD_FILE URL REPORT_FILE_NAME

if "%~1" == "" (
      echo User Name is missing.
    echo %paramRequiredMessage%
    exit /b 1
  )

if "%~2" == "" (
    echo Password or Password_File is missing.
    echo %paramRequiredMessage%
    exit /b 1
  )

if "%~3" == "" (
    echo URL is missing.
    echo %paramRequiredMessage%
    exit /b 1
  )
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
if "%~4" == "" (
    echo Role Assignment Report File Name is missing.
    echo %paramRequiredMessage%
    exit /b 1
  )

call epmautomate.bat login %~1 "%~2" %~3
REM call epmautomate.bat login %~1 "%~2" %~3

if %errorlevel% NEQ 0 exit /b 1
   call epmautomate.bat roleAssignmentReport "%4"
if %errorlevel% NEQ 0 exit /b 1
   call epmautomate.bat downloadFile "%4"
if %errorlevel% NEQ 0 exit /b 1

set filePath="%cd%\%4"

if exist %filePath% (
    SETLOCAL EnableDelayedExpansion
    set /a lineCount=0
    set /a userCount=0
    set userHeaderFound=false
    for /f "tokens=*" %%A in ( 'type %filePath%' ) do (
        set /a lineCount+=1
        set line=%%A

        REM Fetch username from role assignment information row
        if !userHeaderFound!==true (
            for /f "delims=," %%i in ("!line!") do (
                set userName=%%i
          )
        if NOT !userName! == "" (
          if !userCount! gtr 0 if NOT !userName! == !lastUserName! (
              set /a userCount+=1
          set users[!userCount!]=!userName!
                )
          if !userCount! == 0 (
                set /a userCount+=1
                set users[!userCount!]=!userName!
                )
      set lastUserName=!userName!
            )
        )

        REM Check for headers of Role Assignment Report
        if "!line!"=="User Login,First Name,Last Name,Email,Role,Granted
through Group" (
                set userHeaderFound=true
            )
    if "!line!"=="User Login,First Name,Last Name,Email,Roles,Granted
Through" (
        set userHeaderFound=true
         )
    )
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    echo Number of Users: !userCount!
    for /l %%n in (1,1,!userCount!) do (
  REM echo !users[%%n]!
    )
    endlocal

) else (
      echo Invalid provisioning report file path - %filePath%.
)
```

Création de rapports d'audit sur les utilisateurs affectés à des rôles

Utilisez les scripts de cette section afin d'automatiser le processus de création d'un rapport d'audit pour les utilisateurs affectés à des rôles prédéfinis dans un environnement, et éventuellement afin d'envoyer le rapport par courriel à un destinataire.

Ce rapport d'audit présente les utilisateurs affectés à des groupes ou rôles prédéfinis qui ont été modifiés depuis la dernière génération du rapport. Pour créer un rapport d'audit quotidien, exécutez ce script tous les jours.

Créez `provisioningAuditReport.bat` en copiant le script suivant. Ce script de lot d'enveloppe appelle le script de PowerShell `provisioningAuditReport.ps1`, le code source qui est fourni plus loin dans ce scénario.

> **Remarque**
>
> - Les paramètres d'entrée pour l'exécution de `provisioningAuditReport.bat` sont les suivants : `username`, `password` ou `password_file`, `service_url` et `report_email_to_address` (facultatif ; requis uniquement si vous voulez envoyer le rapport par courriel).
> - Si le mot de passe contient des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux.

```
@echo off
set paramRequiredMessage=Syntax: provisioningAuditReport.bat USERNAME
PASSWORD/PASSWORD_FILE URL [REPORT_EMAIL_TO_ADDRESS]

if "%~1" == "" (
  echo User Name is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )
if "%~2" == "" (
  echo Password or Password_File is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )
if "%~3" == "" (
  echo URL is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
PowerShell.exe -File provisioningAuditReport.ps1 %*
```

`provisioningAuditReport.bat` appelle `provisioningAuditReport.ps1`, que vous créez en copiant le script suivant.

`provisioningAuditReport.ps1` crée le rapport d'audit. Placez-le dans le même répertoire que `provisioningAuditReport.bat`.

```
$username=$args[0]
$password=$args[1]
$url=$args[2]
$reportemailtoaddress=$args[3]

$date=$(get-date -f dd_MM_yy_HH_mm_ss)
$datedefaultformat=$(get-date)
$logdir="./logs/"
$logfile="$logdir/epmautomate-provisionauditreport-" + $date + ".log"
$reportdir="./reports/"
$provisionreport="provreport-audittest-" + $date + ".csv"
$provisionreporttemp="./provreport-audittest-temp.csv"
$provisionreportunique="./provreport-audittest-unique.csv"
$provisionreportbaselineunique="./provreport-audittest-baseline-unique.csv"

function EchoAndLogMessage
{
  $message=$args[0]
  echo "$message"
  echo "$message" >> $logfile
}

function Init
{
  $logdirexists=Test-Path $logdir
  if (!($logdirexists)) {
    mkdir $logdir 2>&1 | out-null
    }

  $logfileexists=Test-Path $logfile
  if ($logfileexists) {
    rm $logfile 2>&1 | out-null
    }

  $reportdirexists=Test-Path $reportdir
  if (!($reportdirexists)) {
    mkdir $reportdir 2>&1 | out-null
    }
}

function PostProcess
{
  rm $provisionreporttemp
  mv -Force $provisionreportunique $provisionreportbaselineunique
}

function ProcessCommand
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
{
  $op=$args
  echo "EPM Automate operation: epmautomate.bat $op" >> $logfile
  epmautomate.bat $op >> $logfile 2>&1
  if ($LASTEXITCODE -ne 0) {
    echo "EPM Automate operation failed: epmautomate.bat $op. See $logfile
for details."
    exit
    }
}

function RunEpmAutomateCommands
{
  EchoAndLogMessage "Running EPM Automate commands to generate the
provisioning report."
  ProcessCommand login $username $password $url
  ProcessCommand provisionreport $provisionreport
  ProcessCommand downloadfile $provisionreport
  ProcessCommand deletefile $provisionreport
  ProcessCommand logout
}

function CreateProvisionReportTempFile
{
  # Loop through iteration csv file and parse
  Get-Content $provisionreport | ForEach-Object {
  $elements=$_.split(',')
  echo "$($elements[0]),$($elements[2])" >> $provisionreporttemp
  }
}

function CreateUniqueElementsFile
{
  gc $provisionreporttemp | sort | get-unique > $provisionreportunique
}

function CheckBaselineAndCreateAuditReport
{
  $provisionreportbaselineuniqueexists=Test-
Path $provisionreportbaselineunique
  if (!($provisionreportbaselineuniqueexists)) {
    EchoAndLogMessage "No existing provisioning report, so comparison with a
baseline is not possible. Audit report will be created at the next test run."
  } else {
    CreateAuditReport
    }
}

function EmailAuditReport
{
  $auditreport=$args[0]
  $elements=$auditreport.split('/')
  $auditreportname=$elements[2]

  if (${reportemailtoaddress} -match "@") {
    EchoAndLogMessage "Emailing audit report"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    ProcessCommand login $username $password $url
    ProcessCommand uploadFile $auditreport
    ProcessCommand sendMail $reportemailtoaddress "Provisionining Audit
Report" Body="Provisioning Audit Report is attached."
Attachments=$auditreportname
    ProcessCommand deleteFile $auditreportname
    ProcessCommand logout
  }
}

function CreateAuditReport
{
  $auditreport=$reportdir + "auditreport-"+ $date + ".txt"
  $additions = @()
  $deletions = @()

  EchoAndLogMessage "Comparing previous provisioning report with the current
report."
  $compare=compare-object (get-content $provisionreportunique) (get-
content $provisionreportbaselineunique)

  $compare | foreach  {
    if ($_.sideindicator -eq '<=')
     {
        $additions += $_.inputobject
     } elseif ($_.sideindicator -eq '=>') {
        $deletions += $_.inputobject
     }
  }

  echo "Provisioning Audit Report for $datedefaultformat" > $auditreport
  echo "-----------------------------------------------" >> $auditreport

  if ($additions.count -ne 0)
  {
    echo " "          >> $auditreport
    echo "Additions:" >> $auditreport
    foreach($element in $additions) { echo "$element" >> $auditreport }
    }

  if ($deletions.count -ne 0)
  {
    echo " "          >> $auditreport
    echo "Deletions:" >> $auditreport
    foreach($element in $deletions) { echo "$element" >> $auditreport }
  }

  if (($additions.count -eq 0) -and ($deletions.count -eq 0))
  {
    echo " "                                    >> $auditreport
    echo "No changes from last audit report." >> $auditreport
  }

  EchoAndLogMessage "Provisioning audit report has been
generated: $auditreport."
  EmailAuditReport $auditreport
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
}

Init
EchoAndLogMessage "Starting EPMAutomate provisioning audit reporting"
RunEpmAutomateCommands
CreateProvisionReportTempFile
CreateUniqueElementsFile
CheckBaselineAndCreateAuditReport
PostProcess
EchoAndLogMessage "EPMAutomate provisioning audit reporting completed"
```

Création d'un rapport d'audit sur la révocation et l'affectation de rôle

Utilisez le script PowerShell de cette section pour automatiser le processus de création d'un rapport d'audit qui décrit en détail l'affectation et la révocation de rôle dans un environnement.

Créez `AuditReportRoleAssignment.bat` en copiant le script suivant. Ce script de lot d'enveloppe appelle le script de PowerShell `AuditReportRoleAssignment.ps1`, le code source qui est fourni plus loin dans ce scénario.

> **Remarque**
>
> - Paramètres d'entrée pour l'exécution de `AuditReportRoleAssignment.bat` : `username`, `password` ou `password_file`, et `service_url`.
> - Si le mot de passe contient des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux.

Script : AuditReportRoleAssignment.bat

```
@echo off
set paramRequiredMessage=Syntax: AuditReportRoleAssignment.bat USERNAME
PASSWORD/PASSWORD_FILE URL

if "%~1" == "" (
   echo User Name is missing.
   echo %paramRequiredMessage%
   exit /b 1
   )
if "%~2" == "" (
   echo Password or Password_File is missing.
   echo %paramRequiredMessage%
   exit /b 1
   )
if "%~3" == "" (
   echo URL is missing.
   echo %paramRequiredMessage%
   exit /b 1
   )

PowerShell.exe -File AuditReportRoleAssignment.ps1 %*
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Script : AuditReportRoleAssignment.ps1

```
# EPM Automate Role Assignment Audit Report Script
$username=$args[0]
$password=$args[1]
$url=$args[2]

# Generic variables
$date=$(get-date -f dd_MM_yy_HH_mm_ss)
$datedefaultformat=$(get-date)
$logdir="./logs/"
$logfile="$logdir/epmautomate-provisionauditreport-" + $date + ".log"
$reportdir="./reports/"
$provisionreport="provreport-audittest-" + $date + ".csv"
$provisionreporttemp="./provreport-audittest-temp.csv"
$provisionreportunique="./provreport-audittest-unique.csv"
$provisionreportbaselineunique="./provreport-audittest-baseline-unique.csv"

function EchoAndLogMessage
{
    $message=$args[0]
    echo "$message"
    echo "$message" >> $logfile
}
function Init
{
    $logdirexists=Test-Path $logdir
    if (!($logdirexists)) {
        mkdir $logdir 2>&1 | out-null
    }
    $logfileexists=Test-Path $logfile
    if ($logfileexists) {
        rm $logfile 2>&1 | out-null
    }
    $reportdirexists=Test-Path $reportdir
    if (!($reportdirexists)) {
        mkdir $reportdir 2>&1 | out-null
    }
}

function PostProcess
{
    rm $provisionreporttemp
    mv -Force $provisionreportunique $provisionreportbaselineunique
}

function ProcessCommand
{
    $op=$args
    echo "EPM Automate operation: epmautomate.bat $op" >> $logfile
    epmautomate.bat $op >> $logfile 2>&1
    if ($LASTEXITCODE -ne 0) {
        echo "EPM Automate operation failed: epmautomate.bat $op.
See $logfile for details."
        exit
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    }
}

function RunEpmAutomateCommands
{
    EchoAndLogMessage "Running EPM Automate commands to generate the audit
report."
    ProcessCommand login $username $password $url
    ProcessCommand provisionreport $provisionreport
    ProcessCommand downloadfile $provisionreport
    ProcessCommand deletefile $provisionreport
    ProcessCommand logout
}
function CreateProvisionReportTempFile
{
    # Loop through iteration csv file and parse
    Get-Content $provisionreport | ForEach-Object {
        $elements=$_.split(',')
        echo "$($elements[0]),$($elements[2])" >> $provisionreporttemp
    }
}

function CreateUniqueElementsFile
{
    gc $provisionreporttemp | sort | get-unique > $provisionreportunique
}

function CheckBaselineAndCreateAuditReport
{
    $provisionreportbaselineuniqueexists=Test-
Path $provisionreportbaselineunique
    if (!($provisionreportbaselineuniqueexists)) {
        EchoAndLogMessage "Could not find a baseline audit report to compare
with. Audit report will be created next time you run test."
    } else {
        CreateAuditReport
    }
}

function CreateAuditReport
{
    $auditreport=$reportdir + "auditreport-"+ $date + ".txt"
    $additions = @()
    $deletions = @()
    EchoAndLogMessage "Comparing previous audit report with the current one."
    $compare=compare-object (get-content $provisionreportunique) (get-
content $provisionreportbaselineunique)
    $compare | foreach  {
        if ($_.sideindicator -eq '<=')
        {
            $additions += $_.inputobject
        } elseif ($_.sideindicator -eq '=>') {
            $deletions += $_.inputobject
        }
    }
    echo "Provisioning Audit Report for $datedefaultformat" > $auditreport
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    echo "-----------------------------------------------" >> $auditreport
    if ($additions.count -ne 0)
    {
        echo " "          >> $auditreport
        echo "Additions:" >> $auditreport
        foreach($element in $additions) { echo "$element" >> $auditreport }
    }
    if ($deletions.count -ne 0)
    {
        echo " "          >> $auditreport
        echo "Deletions:" >> $auditreport
        foreach($element in $deletions) { echo "$element" >> $auditreport }
    }
    if (($additions.count -eq 0) -and ($deletions.count -eq 0))
    {
        echo " "                                    >> $auditreport
        echo "No changes from last audit report." >> $auditreport
    }
    EchoAndLogMessage "Role audit report generated: $auditreport."
}

Init
EchoAndLogMessage "Starting EPMAutomate role audit report generation"
RunEpmAutomateCommands
CreateProvisionReportTempFile
CreateUniqueElementsFile
CheckBaselineAndCreateAuditReport
PostProcess
EchoAndLogMessage "EPMAutomate role audit report completed"
```

Masquage des journaux d'accès et du rapport d'activité pour respecter les lois sur la confidentialité

Utilisez les scripts de cette section pour automatiser le processus de masquage d'informations dans le rapport d'activité ou les journaux d'accès afin de respecter les lois sur la confidentialité, et éventuellement pour envoyer le rapport par courriel à un destinataire.

En raison des lois exigeantes de certains pays en matière de confidentialité, les informations disponibles dans les rapports d'activité et les journaux d'accès doivent peut-être être masquées des administrateurs de service pour protéger la vie privée des utilisateurs.

Utilisez `anonymizeData.bat` pour masquer des informations dans le rapport d'activité ou les journaux d'accès afin de respecter les lois sur la confidentialité, ou pour envoyer le rapport par courriel. Pour masquer des informations, planifiez ce script ou une variation de ce dernier en utilisant le planificateur Windows de sorte qu'il soit exécuté tous les jours peu après le processus de maintenance quotidienne pour chaque environnement.

Utilisez les sources d'information suivantes :

- Surveillance de l'utilisation à l'aide de rapports d'activité et de journaux d'accès dans le guide *Guide de mise en route pour les administrateurs*
- Automatisation de l'exécution de scripts

Créez manuellement `anonymizeData.bat` en copiant le script Windows fourni dans la procédure suivante et planifiez-le à l'aide du planificateur Windows. Vous pouvez créer et

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

exécuter des scripts semblables adaptés à la plate-forme si vous n'utilisez pas Windows pour la planification.

`anonymizeData.bat` est un script d'enveloppe exécutant le script `anonymizeData.ps1`, que vous pouvez créer et mettre à jour comme indiqué dans la procédure suivante.

Si le mot de passe utilisé contient des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux.

1. Créez un fichier batch (BAT) nommé `anonymizeData.bat` contenant le script suivant et enregistrez-le à un emplacement pratique, par exemple `C:\automate_scripts`.

```
@echo off
set paramRequiredMessage=Syntax: anonymizeData.bat USERNAME PASSWORD/
PASSWORD_FILE URL [EMAIL_TO_ADDRESS]

if "%~1" == "" (
  echo User Name is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )
if "%~2" == "" (
  echo Password or Password_File is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )
if "%~3" == "" (
  echo URL is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )

PowerShell.exe -File anonymizeData.ps1 %*
```

2. Créez un script PowerShell (PS1) nommé `anonymizeData.ps1` contenant le script suivant et enregistrez-le à un emplacement pratique, par exemple `C:\automate_scripts`.

```
# Anonymize data script

$username=$args[0]
$password=$args[1]
$url=$args[2]
$emailtoaddress=$args[3]

# Generic variables
$date=$(get-date -f dd_MM_yy_HH_mm_ss)
$datedefaultformat=$(get-date)
$logdir="./logs/"
$logfile="$logdir/anonymize-data-" + $date + ".log"
$filelist="filelist.txt"

function LogMessage
{
  $message=$args[0]

  echo "$message" >> $logfile
}
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
function EchoAndLogMessage
{
  $message=$args[0]

  echo "$message"
  echo "$message" >> $logfile
}
function Init
{
   $logdirexists=Test-Path $logdir
   if (!($logdirexists)) {
       mkdir $logdir 2>&1 | out-null
    }

   $logfileexists=Test-Path $logfile
   if ($logfileexists) {
       rm $logfile 2>&1 | out-null
    }

   $filelistexists=Test-Path $filelist
   if ($filelistexists) {
       rm $filelist 2>&1 | out-null
    }
}

function ProcessCommand
{
   $op=$args
   echo "EPM Automate operation: epmautomate.bat $op" >> $logfile
   if ($op -eq 'listfiles') {
      epmautomate.bat $op | where {$_ -like ' apr/*/access_log.zip'} | Tee-
Object -FilePath $filelist | Out-File $logfile -Append 2>&1
    } else {
      epmautomate.bat $op >> $logfile 2>&1
        if ($LASTEXITCODE -ne 0) {
           echo "EPM Automate operation failed: epmautomate.bat $op.
See $logfile for details."
           #exit
           }
    }
}

function RunEpmAutomateCommands
{
    EchoAndLogMessage "Running EPM Automate commands to anonymize data in
the access logs and activity reports."
    ProcessCommand login $username $password $url
    ProcessCommand listfiles
    ProcessFiles
    ProcessCommand logout
}

function ProcessActivityReport
{
    $activityreport=$args[0]
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    $user=$args[1]

    $activityreportexists=Test-Path "$activityreport"
    if ($activityreportexists) {
        LogMessage "Removing User ID: $user from activity
report $activityreport"
        (Get-Content "$activityreport").replace("$user", 'XXXXX') | Set-
Content "$activityreport"
        $txt = [io.file]::ReadAllText("$activityreport") -replace
"`r`n","`n"
        [io.file]::WriteAllText("$activityreport", $txt)
        #Get-ChildItem -File -Recurse | % { $x = get-content -raw -
path $activityreport; $x -replace "`r`n","`n" | set-content -
path $activityreport }
    }
}

function AnonymizeData
{
    $aprdir=$args[0]
    $datestampdir=$args[1]
    $path="$aprdir/$datestampdir"
    $accesslogzipped="access_log.zip"
    $accesslog="access_log.csv"
    $accesslogupdated=$accesslog + ".tmp"
    $activityreportfile="$datestampdir" + ".html"
    $userArray = @()

    expand-Archive -Path "$path/$accesslogzipped" -DestinationPath $path
    rm $path/$accesslogzipped 2>&1 | out-null
    $accesslogexists=Test-Path "$path/$accesslog"
    if ($accesslogexists) {
        EchoAndLogMessage "Processing access log: $path/$accesslog"
        Get-Content $path/$accesslog | ForEach-Object {
            $elements=[regex]::Split( $_ , ',(?=(?:[^"]|"[^"]*")*$)' )
            $date=$elements[0]
            $time=$elements[1]
            $uri=$elements[2]
            $duration=$elements[3]
            $bytes=$elements[4]
            $ip=$elements[5]
            $user=$elements[6]
            $screen=$elements[7]
            $action=$elements[8]
            $object=$elements[9]
            if ($date -like 'Date') {
                echo "$_" >> $path/$accesslogupdated
            } else {
                if ($user -notlike '-') {
                    LogMessage "Removing instance of User ID: $user
from $path/$accesslog."
                    echo
"$date,$time,$uri,$duration,$bytes,$ip,XXXXX,$screen,$action,$object"
>> $path/$accesslogupdated
                    $userArray += $user
                } else {
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
                    echo
"$date,$time,$uri,$duration,$bytes,$ip,$user,$screen,$action,$object"
>> $path/$accesslogupdated
                }
            }
        }
        #Get-ChildItem -File -Recurse | % { $x = get-content -raw -
path $path/$accesslogupdated; $x -replace "`r`n","`n" | set-content -
path $path/$accesslogupdated }
        $txt = [io.file]::ReadAllText("$path/$accesslogupdated") -replace
"`r`n","`n"
        [io.file]::WriteAllText("$path/$accesslogupdated", $txt)
        mv -Force $path/$accesslogupdated $path/$accesslog
        Compress-Archive -Path $path/$accesslog $path/$accesslogzipped
        rm $path/$accesslog 2>&1 | out-null
    }

    EchoAndLogMessage "Processing activity
report: $path/$activityreportfile"
    $userArray = $userArray | Select-Object -Unique
    foreach ($element in $userArray) {
        ProcessActivityReport "$path/$activityreportfile"
"$element"
    }
}

function ProcessFiles
{
    # Loop through iteration csv file and parse
    Get-Content $filelist | ForEach-Object {
        $fullpath=$_.trim()
        $elements=$fullpath.split('/')
        $aprdir=$elements[0]
        $datestampdir=$elements[1]
        $accesslogfile="access_log.zip"
        $activityreportfile="$datestampdir" + ".html"
        $datestampdirexists=Test-Path "$aprdir/$datestampdir"
        $accesslog="$aprdir/$datestampdir/$accesslogfile"
        $activityreport="$aprdir/$datestampdir/$activityreportfile"

        echo "fullpath: $fullpath" >> $logfile
        echo "aprdir: $aprdir, datestampdir: $datestampdir" >> $logfile
        if (!($datestampdirexists)) {
            mkdir "$aprdir/$datestampdir" -ea 0 2>&1 | out-null
            ProcessCommand downloadfile "$accesslog"
            ProcessCommand downloadfile "$activityreport"
            mv "$accesslogfile" "$aprdir/$datestampdir"
            mv "$activityreportfile" "$aprdir/$datestampdir"
            AnonymizeData "$aprdir" "$datestampdir"
            ProcessCommand deletefile "$accesslog"
            ProcessCommand deletefile "$activityreport"
            ProcessCommand uploadfile "$accesslog" "$aprdir/$datestampdir"
            ProcessCommand uploadfile "$activityreport"
"$aprdir/$datestampdir"
        } else {
            EchoAndLogMessage "Files in directory $aprdir/$datestampdir
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
were processed earlier. Skipping these files."
        }
    }
}

function callSendMail
{
    $elements=$logfile.split('/')
    $logfilename=$elements[3]

    if (${emailtoaddress} -match "@") {
        epmautomate.bat login ${username} ${password} ${url}
        epmautomate.bat uploadFile "$logfile"
        epmautomate.bat sendMail $emailtoaddress "Mask Access Logs and
Activity Reports results" Body="The results of running the Mask Access
Logs and Activity Reports script are attached." Attachments=$logfilename
        epmautomate.bat deleteFile "$logfilename"
        epmautomate.bat logout
    }
}

Init
EchoAndLogMessage "Starting the anonymize data script"
RunEpmAutomateCommands
EchoAndLogMessage "Anonymize data script completed"
EchoAndLogMessage "Refer to logfile: $logfile for details."
callSendMail
```

3. A l'aide du planificateur Windows, planifiez `anonymizeData.bat`. Reportez-vous à la section Automatisation de l'exécution de scripts pour obtenir les étapes détaillées.

   Vous devez fournir les valeurs de paramètre suivantes pour exécuter `anonymizeData.bat` :

   - Nom d'utilisateur d'un administrateur de service
   - Mot de passe de l'administrateur de service ou emplacement du fichier de mot de passe crypté
   - URL de l'environnement de service dans lequel les journaux d'accès et les rapports d'activité doivent être masqués
   - **Facultatif :** adresse électronique à laquelle le rapport doit être envoyé. Le rapport n'est envoyé par courriel que si cette valeur est indiquée.

Automatisation des téléchargements de rapport d'activité sur un ordinateur local

Utilisez le script de cette section pour automatiser le téléchargement des rapports d'audit à partir d'un environnement vers un ordinateur local.

Utilisez `syncAprReports.bat` pour télécharger les rapports d'activité. Vous pouvez planifier le fichier batch à l'aide du planificateur Windows afin d'automatiser le téléchargement des rapports d'activité. Reportez-vous à la section Surveillance de l'utilisation à l'aide de rapports d'activité et de journaux d'accès du guide *Guide de mise en route pour les administrateurs* pour obtenir des informations détaillées sur le rapport d'activité.

Créez manuellement `syncAprReports.bat` en copiant le script fourni dans la procédure suivante, puis en mettant à jour les paramètres de connexion. Ce script vérifie l'environnement

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

et télécharge uniquement les rapports qui sont plus récents que ceux disponibles dans le répertoire de téléchargement sur l'ordinateur local.

> **Remarque**
>
> - Le script doit être exécuté à partir d'un ordinateur Windows uniquement.
> - Ce script ne télécharge pas le rapport d'activité des commentaires qui est généré lorsque les utilisateurs soumettent un commentaire.
> - Si le mot de passe utilisé contient des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux.

1. Créez un fichier batch (`.BAT`) nommé `syncAprReports.bat` contenant le script suivant et enregistrez-le à un emplacement pratique, par exemple `C:\automate_scripts`.

```
@echo off
title APR
setlocal DisableDelayedExpansion

REM To hardcode the values in the script replace %1, %2, %3, and %4, with
the actual values.
REM Example:
REM set apr_dir="C:\Oracle\apr"
REM set username="serviceAdmin"
REM set password="Ex@mple!"
REM set url="https://test-example.stg-pbcs.us1.oraclecloud.com"
set apr_dir=%1
set username=%2
set password=%3
set url=%4
setlocal EnableDelayedExpansion
set epmautomate_dir=%cd%
set lastfile=
set argC=0
for %%x in (%*) do Set /A argC+=1
if %argC% neq 0 (
    if %argC% neq 3 (
        if %argC% neq 4 (
            goto :usage
        )
    )
)
goto :login
:usage
echo.
echo Invalid syntax. Please check the parameters.
echo Syntax:
echo 1) syncAprReports.bat APR_FolderPath_on_client username password url
echo      or
echo 2) set the parameters in the file and use below syntax
echo    syncAprReports.bat
goto :end

:login
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
setlocal DisableDelayedExpansion
for /f "delims=" %%i in ('epmautomate login %username% %password% %url%')
do set result=%%i
if "Login successful" neq "%result%" (
    echo Login Failed
    goto :end
)

if not exist %apr_dir% (
echo.
echo apr folder does not exist
GOTO :end
)
cd /D %apr_dir%
for /f "delims=" %%D in ('dir /a:d /b /o:-n') do (
REM AFTER: for /f "delims=" %%D in ('dir /a-d /b /s /o:-n') do (
    set "lastFile=%%~nD"
    goto :next
)

:next
setlocal EnableDelayedExpansion
echo.
echo Most Recent APR on client is %lastFile%

set "output_cnt=0"
cd /D %epmautomate_dir%
for /F "delims=" %%f in ('epmautomate listfiles') do (

    cd /D !apr_dir!
    set "line=%%f"
    for /f "tokens=* delims= " %%a in ("!line!") do set line=%%a
    if "!line:~0,3!" equ "apr" (

        if "!line:~4,8!" neq "Feedback" (

            set isValidFile=false
            if "!line:~-5!" equu ".html" set isValidFile=true
            if "!line:~-5!" equ ".json" set isValidFile=true

            if "!isValidFile!" equ "true" (

                if "%lastFile%" lss "!line:~4,19!" (

                        if "!line:~4,19!" neq "!dirname!" (

                            set apr_dir=!apr_dir:"=!
                            set /a output_cnt+=1
                            set "output[!output_cnt!]=!apr_dir!\!
line:~4,19!"
                            set "dirname=!line:~4,19!"

                            REM start downloading
                            mkdir "!dirname!"
                            cd /D !dirname!
                            echo downloading !line!
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
                                set "downloadDir=!apr_dir!\!dirname!"

                                cd /D %epmautomate_dir%
                                for /f "delims=" %%i in ('epmautomate
downloadfile "!line!"') do set result1=%%i
                                move "!line:~24!" "!downloadDir!" > nul
                                echo !result1!
                                REM end downloading

                            ) else (

                                REM start downloading
                                cd /D !dirname!
                                echo downloading !line!
                                set apr_dir=!apr_dir:"=!
                                set "downloadDir=!apr_dir!\!dirname!"
                                cd /D %epmautomate_dir%
                                for /f "delims=" %%i in ('epmautomate
downloadfile "!line!"') do set result1=%%i
                                move "!line:~24!" "!downloadDir!" > nul
                                echo !result1!
                                REM end downloading

                            )
                        ) else (

                            REM TO-DO

                        )
                    )
                )
            )
        )
    )

    echo.
    echo %output_cnt% APR's downloaded
    for /L %%n in (1 1 !output_cnt!) DO echo !output[%%n]!
    goto :end

    :end
    cd /D %epmautomate_dir%
    endlocal
```

2. Modifiez `syncAprReports.bat` pour définir les valeurs des paramètres du tableau suivant. Ces valeurs permettent d'accéder à l'environnement pour télécharger des rapports d'activité.

Tableau 3-3 Valeurs de paramètre à inclure dans `syncAprReports.bat`

Paramètre	Valeur attendue
`set apr_dir=%1`	Spécifiez un répertoire existant dans lequel les rapports d'activité doivent être téléchargés. **Exemple :** `set apr_dir="C:\Oracle\apr"`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 3-3 (suite) Valeurs de paramètre à inclure dans `syncAprReports.bat`

Paramètre	Valeur attendue
`set username=%2`	Nom d'utilisateur Oracle Fusion Cloud Enterprise Performance Management à employer pour se connecter à l'environnement afin de télécharger des rapports d'activité. **Exemple :** `set username="ServiceAdmin"`

`set password=%3`	Nom et emplacement du fichier qui stocke le mot de passe crypté de l'utilisateur spécifié par la variable `username`. Vous pouvez également indiquer le mot de passe en texte brut de l'utilisateur (non recommandé). Pour obtenir des informations sur la création d'un fichier de mots de passe cryptés, reportez-vous à la commande encrypt.
	Exemples :
	`set password="C:\mySecuredir\password.epw"`
	`set password="Ex@mple1"`

`set url=%4`	URL de l'environnement. **Exemple :** `set url="https://test-example.stg-pbcs.us1.oraclecloud.com"`

3. A l'aide du planificateur Windows, planifiez `syncAprReports.bat`. Reportez-vous à la section Automatisation de l'exécution de scripts pour obtenir les étapes détaillées.

Téléchargement des journaux d'accès à partir d'un environnement

Utilisez le script de cette section pour automatiser le processus de téléchargement des journaux d'accès à partir d'un environnement vers un ordinateur local.

Vous pouvez planifier `syncAccessLog.bat` à l'aide du planificateur Windows afin d'automatiser le téléchargement des fichiers journaux. Reportez-vous à la section Affichage et téléchargement des rapports d'activité et des journaux d'accès du guide *Guide de mise en route pour les administrateurs* pour obtenir les procédures de téléchargement des journaux d'accès à l'aide de la gestion des applications.

Le script suivant vérifie l'environnement et télécharge uniquement les fichiers journaux qui sont plus récents que ceux disponibles dans le répertoire de téléchargement sur l'ordinateur local. Il s'agit d'un script Windows, mais vous pouvez créer un script d'interpréteur de commandes similaire pour les environnements Linux/UNIX.

1. Créez un fichier batch (.BAT) nommé `syncAccessLog.bat` contenant le script suivant et enregistrez-le à un emplacement pratique, par exemple `C:\automate_scripts`.

> **Remarque**
>
> Si votre mot de passe contient des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux.

```
@echo off
title APR
setlocal DisableDelayedExpansion

REM To hardcode the values in the script replace %1, %2, %3, and %4 with
the actual values.
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
REM Example:
REM set apr_dir="C:\Oracle\apr"
REM set username="serviceAdmin"
REM set password="C:\mySecuredir\password.epw"
REM set url="https://test-cloudpln.pbcs.us1.oraclecloud.com"
set apr_dir=%1
set username=%2
set password=%3
set url=%4

setlocal EnableDelayedExpansion
set epmautomate_dir=%cd%
set lastfile=
REM if [%1]==[] goto :usage
REM if [%2]==[] goto :usage
REM if [%3]==[] goto :usage

set argC=0
for %%x in (%*) do Set /A argC+=1
if %argC% neq 0 (
  if %argC% neq 3 (
    if %argC% neq 4 (
      goto :usage
      )
  )
)
goto :login

:usage
echo.
echo Invalid syntax. Please check the parameters.
echo Syntax:
echo 1) syncAccessLog.bat APR_FolderPath_on_client username password url
echo     or
echo 2) set the parameters in the file and use below syntax
echo syncAccessLog.bat
goto :end

:login
setlocal DisableDelayedExpansion
REM for /f "delims=" %%i in ('epmautomate login %2 %3 %4') do set result=%
%i
for /f "delims=" %%i in ('epmautomate login %username% %password% %url%')
do set result=%%i

if not exist %apr_dir% (
echo.
echo apr folder does not exist
GOTO :end
)
cd /D %apr_dir%
for /f "delims=" %%D in ('dir /a:d /b /o:-n') do (
REM AFTER: for /f "delims=" %%D in ('dir /a-d /b /s /o:-n') do (
  set "lastFile=%%~nD"
  goto :next
)
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
:next
setlocal EnableDelayedExpansion
echo.
echo Most Recent Access Log on client is %lastFile%

set "output_cnt=0"
cd /D %epmautomate_dir%
for /F "delims=" %%f in ('epmautomate listfiles') do (

  cd /D !apr_dir!
  set "line=%%f"
  for /f "tokens=* delims= " %%a in ("!line!") do set line=%%a
  if "!line:~0,3!" equ "apr" (
    if "!line:~-4!" equ ".zip" (
      if "%lastFile%" lss "!line:~4,19!" (
          if "!line:~4,19!" neq "!dirname!" (
            set apr_dir=!apr_dir:"=!
            set /a output_cnt+=1
            set "output[!output_cnt!]=!apr_dir!\!line:~4,19!"
            set "dirname=!line:~4,19!"

            REM start downloading
            mkdir "!dirname!"
            cd /D !dirname!
            echo downloading !line!
            set "downloadDir=!apr_dir!\!dirname!"
            cd /D %epmautomate_dir%
            for /f "delims=" %%i in ('epmautomate downloadfile "!line!"')
do set result1=%%i
            move "!line:~24!" "!downloadDir!" > nul
            echo !result1!
            REM end downloading

          ) else (
            REM start downloading
            cd /D !dirname!
            echo downloading !line!
            set apr_dir=!apr_dir:"=!
            set "downloadDir=!apr_dir!\!dirname!"
            cd /D %epmautomate_dir%
            for /f "delims=" %%i in ('epmautomate downloadfile "!line!"')
do set result1=%%i
            move "!line:~24!" "!downloadDir!" > nul
            echo !result1!
            REM end downloading
          )
      ) else (
        REM TO-DO
      )
    )
  )
)

echo.
echo %output_cnt% access logs downloaded
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
for /L %%n in (1 1 !output_cnt!) DO echo !output[%%n]!
goto :end

:end
cd /D %epmautomate_dir%
endlocal
```

2. Modifiez `syncAccessLog.bat` pour définir les valeurs des paramètres du tableau suivant. Ces valeurs permettent d'accéder à l'environnement pour télécharger les journaux d'accès.

Tableau 3-4 Valeurs de variable à inclure dans `syncAccessLog.bat`

Variable	Valeur attendue
`set apr_dir=%1`	Spécifiez le répertoire dans lequel les journaux d'accès doivent être téléchargés. **Exemple :** `set apr_dir="C:\Oracle\apr"`
`set username=%2`	Nom d'utilisateur Oracle Fusion Cloud Enterprise Performance Management à employer pour se connecter à l'environnement afin de télécharger des journaux d'accès. **Exemple :** `set username="ServiceAdmin"`

`set password=%3`	Nom et emplacement du fichier qui stocke le mot de passe crypté de l'utilisateur spécifié par la variable `username`. Vous pouvez également indiquer le mot de passe en texte brut de l'utilisateur (non recommandé). Pour obtenir des informations sur la création d'un fichier de mots de passe cryptés, reportez-vous à la commande encrypt.
	Exemples :
	`set password="C:\mySecuredir\password.epw"`
	`set password="P@ssword1"`

`set url=%4`	URL de l'environnement. **Exemple :** `set url="https://test-cloudpln.pbcs.us1.oraclecloud.com"`

3. A l'aide du planificateur Windows, planifiez `syncAccessLog.bat`. Reportez-vous à la section Automatisation de l'exécution de scripts pour obtenir les étapes détaillées.

Automatisation du clonage d'environnements

Utilisez le script de cette section pour automatiser le clonage des environnements.

Créez un fichier de lot (`.bat`) ou d'interpréteur de commande (`.sh`) contenant un script semblable à celui ci-dessous pour cloner un environnement. Les exemples de scripts traitent les activités suivantes ::

- Connectez-vous à l'environnement source.
- Utilisez `Artifact Snapshot` (l'instantané créé lors de la dernière maintenance quotidienne de l'environnement source) ou un autre instantané disponible dans l'environnement source pour convertir l'environnement cible en un clone de l'environnement source.
- Créez éventuellement des utilisateurs et leurs affectations de rôle d'application et prédéfini correspondant à ceux de l'environnement source.
- Vous pouvez éventuellement modifier l'heure de début de la maintenance quotidienne de sorte qu'elle corresponde à celle de l'environnement source.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Déconnectez-vous.

Pour obtenir des informations détaillées sur le processus de clonage, reportez-vous à la section "Clonage d'environnements Cloud EPM " du guide *Administration de Migration*.

Reportez-vous à la section Automatisation de l'exécution de scripts pour plus d'informations sur la planification du script à l'aide du planificateur de tâches Windows.

Windows

1. Créez un fichier batch (.BAT) nommé `cloneEnvironment.bat` contenant le script suivant et enregistrez-le à un emplacement pratique, par exemple `C:\automate_scripts`.

```
@echo off
set paramRequiredMessage=Syntax: cloneEnvironment.bat "SOURCE USERNAME"
"SOURCE PASSWORD FILE" "SOURCE URL" "TARGET USERNAME" "TARGET PASSWORD
FILE" "TARGET URL"

set usersandpredefinedroles="false"
set snapshotname="Artifact Snapshot"
set dailymaintenancestarttime="true"
set dirpath=%~dp0
cd %dirpath:~0,-1%

if "%~1" == "" (
  echo Source User Name is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )
if "%~2" == "" (
  echo Source Password File is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )
if "%~3" == "" (
  echo Source URL is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )
if "%~4" == "" (
  echo Target User Name is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )
if "%~5" == "" (
  echo Target Password File is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )
if "%~6" == "" (
  echo Target URL is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )

PowerShell.exe -File cloneEnvironment.ps1 %~1 %~2 %~3 %~4 %~5 %~6
%usersandpredefinedroles% %snapshotname% %dailymaintenancestarttime%
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

2. Modifiez `cloneEnvironment.bat` pour définir les valeurs des paramètres suivants :

Tableau 3-5 Paramètres à définir dans cloneEnvironment.bat

Paramètre	Description
`usersandpredefinedroles`	Définissez la valeur de ce paramètre sur `true` pour cloner les utilisateurs et leurs affectations de rôle d'application et prédéfini. Pour cloner des utilisateurs et des affectations de rôle, l'utilisateur qui exécute le script doit disposer des rôles Administrateur de service et Administrateur de domaine d'identité dans l'environnement cible.
`snapshotname`	Nom de l'instantané dans l'environnement source à utiliser pour le clonage.
`dailymaintenancestarttime`	Définissez la valeur de ce paramètre sur `true` pour modifier l'heure de début de la maintenance quotidienne de l'environnement cloné de sorte qu'elle corresponde à celle de l'environnement source. Définissez cette valeur sur `false` pour conserver l'heure de début de maintenance quotidienne actuelle de l'environnement cloné.

3. Créez un script PowerShell nommé `cloneEnvironment.ps1` contenant le script suivant, puis enregistrez-le dans le répertoire où vous avez enregistré `cloneEnvironment.bat`, par exemple : `C:\automate_scripts`.

```
# Clone Environment script

$source_username=$args[0]
$source_password=$args[1]
$source_url=$args[2]
$target_username=$args[3]
$target_password=$args[4]
$target_url=$args[5]
$usersandpredefinedroles=$args[6]
$snapshotname=$args[7]
$dailymaintenancestarttime=$args[8]

epmautomate.bat login "${source_username}" "${source_password}" "$
{source_url}"
epmautomate.bat cloneEnvironment "${target_username}" "${target_password}"
"${target_url}" UsersAndPreDefinedRoles="${usersandpredefinedroles}"
SnapshotName="${snapshotname}" DailyMaintenanceStartTime="$
{dailymaintenancestarttime}"
epmautomate.bat logout
```

4. Exécutez `cloneEnvironment.bat` à l'aide de la commande suivante :

```
cloneEnvironment.bat "SOURCE USERNAME" "SOURCE PASSWORD FILE" "SOURCE URL"
"TARGET USERNAME" "TARGET PASSWORD FILE" "TARGET URL"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Par exemple :

```
cloneEnvironment.bat jdoe@example.com C:\mySecuredir\example_pwd.epw
https://source_example.oraclecloud.com jdoe@example.com
C:\mySecuredir\example_pwd2.epw https://target_example.oraclecloud.com.
```

Linux

1. Créez un script shell nommé `cloneEnvironment.sh` contenant le script suivant, puis enregistrez-le à un emplacement pratique.

```
#!/bin/bash

# Update the following parameters
# ------------------------------
epmautomatescript=/home/user1/epmautomate/bin/epmautomate.sh
javahome=/home/user1/jdk_17.0.12/
usersandpredefinedroles="false"
snapshotname="Artifact Snapshot"
dailymaintenancestarttime="true"
# ------------------------------

source_username="$1"
source_password="$2"
source_url="$3"
target_username="$4"
target_password="$5"
target_url="$6"

export JAVA_HOME=${javahome}

if [ "$#" -ne 6 ]; then
    echo "Usage: ./cloneEnvironment.sh <SOURCE USERNAME> <SOURCE PASSWORD
FILE> <SOURCE URL> <TARGET USERNAME> <TARGET PASSWORD FILE> <TARGET URL>"
    exit 1
fi

${epmautomatescript} login "${source_username}" "${source_password}" "$
{source_url}"
${epmautomatescript} cloneEnvironment "${target_username}" "$
{target_password}" "${target_url}" UsersAndPreDefinedRoles="$
{usersandpredefinedroles}" SnapshotName="${snapshotname}"
DailyMaintenanceStartTime="${dailymaintenancestarttime}"
${epmautomatescript} logout
```

2. Modification de `cloneEnvironment.sh` pour définir les valeurs des paramètres suivants :

Tableau 3-6 Paramètres à définir dans cloneEnvironment.sh

Paramètre	Description
`epmautomatescript`	Chemin absolu de l'exécutable EPM Automate (`epmautomate.sh`).
`javahome`	Emplacement de JAVA_HOME.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 3-6 (suite) Paramètres à définir dans cloneEnvironment.sh

Paramètre	Description
`usersandpredefinedroles`	Définissez la valeur de ce paramètre sur `true` pour cloner les utilisateurs et leurs affectations de rôle d'application et prédéfini. Pour cloner des utilisateurs et des affectations de rôle, l'utilisateur qui exécute le script doit disposer des rôles Administrateur de service et Administrateur de domaine d'identité dans l'environnement cible.
`snapshotname`	Nom de l'instantané dans l'environnement source à utiliser pour le clonage.
`dailymaintenancestarttime`	Définissez la valeur de ce paramètre sur `true` pour modifier l'heure de début de la maintenance quotidienne de l'environnement cloné de sorte qu'elle corresponde à celle de l'environnement source. Définissez cette valeur sur `false` pour conserver l'heure de début de maintenance quotidienne actuelle de l'environnement cloné.

3. Exécutez `cloneEnvironment.sh`.

```
./cloneEnvironment.sh "SOURCE USERNAME" "SOURCE PASSWORD FILE" "SOURCE
URL" "TARGET USERNAME" "TARGET PASSWORD FILE" "TARGET URL"
```

Par exemple :

```
./cloneEnvironment.sh jdoe@example.com ./home/secure/example_pwd.epw
https://source_example.oraclecloud.com jdoe@example.com ./home/secure/
example_pwd.epw2 https://target_example.oraclecloud.com.
```

Clonage quotidien de l'environnement principal vers l'environnement de secours une fois la maintenance quotidienne de l'environnement principal effectuée

Pour que l'environnement de secours reste à jour par rapport à l'environnement principal, utilisez ces scripts afin de cloner l'environnement principal Oracle Fusion Cloud Enterprise Performance Management vers l'environnement de secours peu après la fin de la maintenance quotidienne de l'environnement principal.

Ces scripts personnalisés déterminent si la maintenance quotidienne planifiée pour la journée est terminée et clonent ensuite l'environnement.

Script Windows

Créez `dailyclone.ps1` en copiant le script PowerShell suivant.

```
# Clone Environment script
#
# Update the following parameters
# -------------------------------
$users_and_predefined_roles="false"
$daily_maintenance_start_time="true"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
$data_management="true"
$app_audit="true"
$job_console="true"
$stored_snapshots_and_files="false"
# ------------------------------

$source_username=$args[0]
$source_password=$args[1]
$source_url=$args[2]
$target_username=$args[3]
$target_password=$args[4]
$target_url=$args[5]

if ($($args.count) -ne 6) {
    echo "Usage: ./dailyclone.ps1 <SOURCE USERNAME> <SOURCE PASSWORD> <SOURCE
URL> <TARGET USERNAME> <TARGET PASSWORD> <TARGET URL>"
    exit 1
}

$amw_time=""

function getDailyMaintenanceStartTime {
    $amwstring=$(epmautomate.bat getDailyMaintenanceStartTime)
    $elements=$amwstring.split(' ')
    $amwtime=$elements[0]
    return $amwtime
}

function goToSleep ($amw_time){
    $current_mdy=Get-Date -AsUTC -UFormat "%m/%d/%Y"
    $current_date_time=Get-Date -AsUTC -UFormat "%m/%d/%Y %H:%M:%S"
    $current_epoch=Get-Date -Date $current_date_time -UFormat "%s"
    $target_date_time=[DateTime]"${current_mdy} ${amw_time}"
    $target_epoch=Get-Date -Date $target_date_time -UFormat "%s"
    $sleep_seconds=$target_epoch - $current_epoch

    # Today's AMW start time has already passed, so add 24 hours to
sleep_seconds
    if ($sleep_seconds -lt 0) {
        $sleep_seconds=$sleep_seconds + 86400
    }

    $sleep_ts=New-TimeSpan -Seconds ${sleep_seconds}
    $sleep_hms="${sleep_ts}" -replace '^\d+?\.'

    echo "Current time is ${current_date_time}. Sleeping for ${sleep_hms},
until daily maintenance start time of ${amw_time}."
    Start-Sleep -Seconds $sleep_seconds
}

function deleteArtifactSnapshotIfExists {
    if (artifactSnapshotExists) {
        $command_del=$(epmautomate.bat deletefile "Artifact Snapshot")
    }
}
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
function artifactSnapshotExists {
    $filelist=$(epmautomate.bat listfiles)
    if ("$filelist".contains("Artifact Snapshot")) {
        return $true
    else
        return $false
    }
}

function cloneEnvironment {
    echo "Checking to see if daily maintenance processing has completed ..."
    while ($true) {
        if (artifactSnapshotExists) {
            echo "Daily maintenance processing has completed ..."
            break
    } else {
            echo "Sleeping for 30 seconds before the next check to see if
daily maintenance processing has completed ..."
            Start-Sleep -Seconds 30
        }
    }

    echo "Encrypting target password ..."
    epmautomate.bat encrypt "${target_password}" "oracleKey"
"target_password.epw"

    echo "Cloning environment ..."
    epmautomate.bat cloneEnvironment "${target_username}"
"target_password.epw" "${target_url}" "SnapshotName=Artifact Snapshot"
"UsersAndPreDefinedRoles=${users_and_predefined_roles}" "DataManagement=$
{data_management}" "appAudit=${app_audit}" "jobConsole=${job_console}"
"storedSnapshotsAndFiles=${stored_snapshots_and_files}"
"dailyMaintenanceStartTime=${daily_maintenance_start_time}"
}

echo "Beginning clone environment script."
echo "Logging into server ..."
epmautomate.bat login ${source_username} ${source_password} ${source_url}
$amwtime=getDailyMaintenanceStartTime
goToSleep ($amwtime)
deleteArtifactSnapshotIfExists
cloneEnvironment
echo "Logging out of server ..."
epmautomate.bat logout
echo "Clone environment script processing has completed."
```

Script Linux/UNIX

Créez `dailyclone.sh` en copiant le script suivant.

```
#!/bin/bash

# Update the following parameters
# -------------------------------
epmautomatescript="LOCATION_EPM_AUTOMATE_EXECUTABLE"
javahome="LOCATION_JAVA_HOME"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
users_and_predefined_roles="false"
data_management="true"
app_audit="true"
job_console="true"
stored_snapshots_and_files="false"
daily_maintenance_start_time="true"
# ------------------------------

source_username="$1"
source_password="$2"
source_url="$3"
target_username="$4"
target_password="$5"
target_url="$6"

export JAVA_HOME=${javahome}

if [ "$#" -ne 6 ]; then
    echo "Usage: ./dailyclone.sh SOURCE_USERNAME SOURCE_PASSWORD SOURCE_URL
TARGET_USERNAME TARGET_PASSWORD TARGET_URL"
    exit 1
fi

amw_time=""

getDailyMaintenanceStartTime() {
    amw_time=$(${epmautomatescript} getDailyMaintenanceStartTime | cut -d' ' -
f1)
}

goToSleep() {
    current_mdy=$(date -u +%m/%d/%Y)
    current_date_time=$(date -u)
    current_epoch=$(date +%s)
    target_epoch=$(date -d "${current_mdy} ${amw_time}" +%s)
    sleep_seconds=$(($target_epoch - $current_epoch))

    # Today's AMW start time has already passed, so add 24 hours to
sleep_seconds
    if [[ ${sleep_seconds} -lt 0 ]]
    then
        sleep_seconds=$((sleep_seconds + 86400))
    fi

    sleep_hms=$(date -d@${sleep_seconds} -u +%H:%M:%S)

    echo "Current time is ${current_date_time}. Sleeping for ${sleep_hms},
until daily maintenance start time of ${amw_time}."
    sleep $sleep_seconds
}

deleteArtifactSnapshotIfExists() {
    found=1
    filelist=$(${epmautomatescript} listfiles)
    if [[ ${filelist} == *"Artifact Snapshot"* ]]
    then
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        command_del=$(${epmautomatescript} deletefile "Artifact Snapshot")
    fi
}

artifactSnapshotExists() {
    found=1

    filelist=$(${epmautomatescript} listfiles)
    if [[ ${filelist} == *"Artifact Snapshot"* ]]
    then
        found=0
    else
        found=1
    fi

    echo ${found}
}

cloneEnvironment() {
    local found=1

    while true
    do
        found=$(artifactSnapshotExists)
        if [[ ${found} -eq 0 ]]
        then
            echo "Daily maintenance processing has completed ..."
            break
        else
            echo "Sleeping for 30 seconds before the next check to see if
daily maintenance processing has completed ..."
            sleep 30
        fi
    done

    echo "Encrypting target password ..."
    ${epmautomatescript} encrypt "${target_password}" "oracleKey"
"target_password.epw"

    echo "Cloning environment ..."
    ${epmautomatescript} cloneEnvironment "${target_username}"
"target_password.epw" "${target_url}" "SnapshotName=Artifact Snapshot"
"UsersAndPreDefinedRoles=${users_and_predefined_roles}" "DataManagement=$
{data_management}" "appAudit=${app_audit}" "jobConsole=${job_console}"
"storedSnapshotsAndFiles=${stored_snapshots_and_files}"
"dailyMaintenanceStartTime=${daily_maintenance_start_time}"
}

echo "Beginning clone environment script."
echo "Logging into server ..."
${epmautomatescript} login ${source_username} ${source_password} ${source_url}
getDailyMaintenanceStartTime
goToSleep
deleteArtifactSnapshotIfExists
cloneEnvironment
echo "Logging out of server ..."
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
${epmautomatescript} logout
echo "Clone environment script processing has completed."
```

Script Groovy

Créez le script Groovy `dailyclone` en copiant le code suivant.

```
// Clone Environment script

// Update the following parameters
String source_username="USERNAME"
String source_password="PASSWORD!"
String source_url="URL OF THE SOURCE ENVIRONMENT"
String target_username="USERNAME"
String target_password="PASSWORD"
String target_url="URL OF THE TARGET ENVIRONMENT"
String snapshot_name="Artifact Snapshot"

// ------------------------------

// Do not update the following parameters
String users_and_predefined_roles="false"
String data_management="true"
String app_audit="true"
String job_console="true"
String stored_snapshots_and_files="false"
String daily_maintenance_start_time="true"
// ------------------------------

def LogMessage(String message) {
    def date = new Date()
    def sdf = new SimpleDateFormat("MM/dd/yyyy HH:mm:ss")
    println('[' + sdf.format(date) + '] ' + message);
}

def LogOperationStatus(EpmAutomateStatus opstatus) {
    def returncode = opstatus.getStatus()
    if (returncode != 0){
        LogMessage(opstatus.getOutput())
    }
    LogMessage('return code: ' + returncode)
}

def getDailyMaintenanceStartTime(EpmAutomate automate) {
    LogMessage("Operation: getDailyMaintenanceStartTime")
    EpmAutomateStatus amwtimestatus =
automate.execute('getDailyMaintenanceStartTime')
    LogOperationStatus(amwtimestatus)
    def amwstring=(amwtimestatus.getOutput())
    def elements=amwstring.split(' ')
    def amwtime=elements[0]
    return amwtime
}

def goToSleep(String amw_time){
    def date = new Date()
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    def current_mdy = new SimpleDateFormat("MM/dd/yyyy")
    def current_date_time = new SimpleDateFormat("MM/dd/yyyy HH:mm:ss")
    float current_epoch = date.getTime() / 1000
    def pattern = "MM/dd/yyyy HH:mm:ss"
    def input = current_mdy.format(date) + " " + amw_time + ":00"
    def target_date_time = Date.parse(pattern, input)
    float target_epoch = target_date_time.getTime() / 1000
    int sleep_seconds = Math.round(target_epoch - current_epoch)

    //Today's AMW start time has already passed, so add 24 hours to
sleep_seconds
    if (sleep_seconds < 0) {
        sleep_seconds = sleep_seconds + 86400
    }

    def sleep_milliseconds = sleep_seconds * 1000
    LogMessage("Current time is " + current_date_time.format(date) + ".
Sleeping until daily maintenance start time of " + amw_time + ":00.")
    sleep(sleep_milliseconds)
}

boolean artifactSnapshotExists(EpmAutomate automate,String snapshot_name) {
    LogMessage("Operation: listfiles")
    EpmAutomateStatus listfilesstatus = automate.execute('listfiles')
    String filelist=(listfilesstatus.getItemsList())
    LogOperationStatus(listfilesstatus)

    if (filelist.contains(snapshot_name)) {
        return true
    } else {
        return false
    }
}

def deleteArtifactSnapshotIfExists(EpmAutomate automate,String snapshot_name){
    if (artifactSnapshotExists(automate,snapshot_name)) {
        LogMessage("Operation: deletefile " + snapshot_name)
        EpmAutomateStatus deletefilestatus =
automate.execute('deletefile',snapshot_name)
        LogOperationStatus(deletefilestatus)
    }
}

def waitForDailyMaintenanceToComplete(EpmAutomate automate,String
snapshot_name) {
    LogMessage("Checking to see if daily maintenance processing has
completed ...")
    while (true) {
        if (artifactSnapshotExists(automate,snapshot_name)) {
            LogMessage("Daily maintenance processing has completed ...")
            break
        } else {
            LogMessage("Sleeping for 30 seconds before the next check to see
if daily maintenance processing has completed ...")
            sleep(30000)
        }
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    }
}

LogMessage("Clone environment script processing beginning ...")

EpmAutomate automate = getEpmAutomate()

LogMessage("Operation: login " + source_username + " " + source_password + "
" + source_url)
EpmAutomateStatus status =
automate.execute('login',source_username,source_password,source_url)
LogOperationStatus(status)

String amwtime = getDailyMaintenanceStartTime(automate)
goToSleep (amwtime)
deleteArtifactSnapshotIfExists(automate,snapshot_name)
waitForDailyMaintenanceToComplete(automate,snapshot_name)

LogMessage("Operation: encrypt " + target_password + " oracleKey
target_password.epw")
status =
automate.execute('encrypt',target_password,'oracleKey','target_password.epw')
LogOperationStatus(status)

LogMessage("Operation: cloneEnvironment " + target_username + "
target_password.epw " + target_url + " SnapshotName=" + snapshot_name + "
UsersAndPreDefinedRoles=" + users_and_predefined_roles + " DataManagement=" +
data_management + " appAudit=" + app_audit + " jobConsole=" + job_console + "
storedSnapshotsAndFiles=" + stored_snapshots_and_files + "
dailyMaintenanceStartTime=" + daily_maintenance_start_time)
EpmAutomateStatus cloneenvironmentstatus =
automate.execute('cloneEnvironment',target_username,"target_password.epw",targ
et_url,"SnapshotName=" + snapshot_name,"UsersAndPreDefinedRoles=" +
users_and_predefined_roles,"DataManagement=" + data_management,"appAudit=" +
app_audit,"jobConsole=" + job_console,"storedSnapshotsAndFiles=" +
stored_snapshots_and_files,"dailyMaintenanceStartTime=" +
daily_maintenance_start_time)
LogOperationStatus(cloneenvironmentstatus)

LogMessage("Operation: logout ")
status = automate.execute('logout')
LogOperationStatus(status)

LogMessage ("Clone environment script processing has completed.")
```

Exécution du script

1. Créez `dailyclone.ps1`, `dailyclone.sh` ou le script Groovy `dailyclone` en copiant un script de l'une des sections précédentes.
2. Mettez les scripts à jour :
   Mise à jour pour `dailyclone.sh` :
   - `epmautomatescript` avec l'emplacement de l'exécutable EPM Automate. Exemple : `epmautomatescript="/home/utils/EPMAutomate/bin/epmautomate.sh"`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- `javahome` avec le répertoire dans lequel le JDK utilisé par EPM Automate est installé. Par exemple : `"/home/user1/jdk_17.0.12"`

Mise à jour pour `dailyclone.ps1` et `dailyclone.sh` :

- `users_and_predefined_roles` : définissez cette valeur sur `true` pour cloner les utilisateurs et leurs affectations de rôle prédéfini (les groupes Contrôle d'accès sont toujours clonés).
- `data_management` : définissez cette valeur sur `false` pour ne pas cloner les enregistrements d'intégration des données. Les enregistrements d'intégration des données ne peuvent être clonés que si les environnements source et cible utilisent la même mise à jour mensuelle, ou si l'environnement cible repose sur une mise à jour plus récente que l'environnement source. Par exemple, vous pouvez cloner les enregistrements de gestion des données 22.01 vers un autre environnement 22.01 ou un environnement 22.02 uniquement.
  Ignoré pour les environnements Narrative Reporting et Oracle Fusion Cloud Enterprise Data Management.
- `app_audit` : définissez cette valeur sur `false` si vous ne souhaitez pas cloner les données d'audit d'application pour les applications Planning, FreeForm et Enterprise Profitability and Cost Management.
  Les informations d'audit de Financial Consolidation and Close et Tax Reporting sont toujours clonées.
- `job_console` : définissez cette valeur sur `false` si vous ne voulez pas cloner les données de la console de jobs.
- `stored_snapshots_and_files` : définissez cette valeur sur `true` si vous voulez cloner le contenu des dossiers de niveau supérieur dans la boîte de réception et la boîte d'envoi (les sous-dossiers ne sont jamais clonés) de l'environnement source.
- `daily_maintenance_start_time` : définissez cette valeur sur `false` si vous ne voulez pas réinitialiser l'heure de début de maintenance de l'environnement cible cloné de sorte qu'elle corresponde à celle de l'environnement source.

Mises à jour pour le script Groovy `dailyclone` :

- `source_username` est le nom d'utilisateur d'un administrateur de service. Le rôle Administrateur de domaine d'identité est requis pour cloner des utilisateurs et des rôles prédéfinis.
- `source_password` est le mot de passe de l'utilisateur identifié par *`SOURCE_USERNAME`*.
- `source_url` est l'URL de l'environnement à cloner.
- `target_username` est le nom d'utilisateur d'un administrateur de service. Le rôle Administrateur de domaine d'identité est requis pour cloner des utilisateurs et des rôles prédéfinis.
- `target_password` est le mot de passe de l'utilisateur identifié par *`TARGET_USERNAME`*.
- `target_URL` est l'URL de l'environnement cible.

3. Exécutez `dailyclone.ps1`, `dailyclone.sh` ou le script Groovy `dailyclone` :
   Windows
   - Dans une fenêtre de commande, accédez au dossier de stockage du script `dailyclone.ps1`.
   - Exécutez la commande `./dailyclone.ps1` *`SOURCE_USERNAME SOURCE_PASSWORD SOURCE_URL TARGET_USERNAME TARGET_PASSWORD TARGET_URL`*. Reportez-vous au tableau suivant pour obtenir une description de ces valeurs.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Table 3-7 Description des paramètres

Paramètre	Description
SOURCE_USERNAME	Nom d'utilisateur d'un administrateur de service. Cet utilisateur doit également disposer du rôle Administrateur de domaine d'identité pour cloner les utilisateurs et les rôles prédéfinis.
SOURCE_PASSWORD	Mot de passe de l'utilisateur identifié par *SOURCE_USERNAME*.
SOURCE_URL	URL de l'environnement à cloner.
TARGET_USERNAME	Nom d'utilisateur de l'administrateur de service dans l'environnement cible. Cet utilisateur doit également disposer du rôle Administrateur de domaine d'identité pour cloner les utilisateurs et les rôles prédéfinis.
TARGET_PASSWORD	Mot de passe de l'utilisateur identifié par *TARGET_USERNAME*.
TARGET_URL	URL de l'environnement cible.

Linux/UNIX

- Dans un interpréteur de commandes UNIX ou Linux, accédez au répertoire de stockage de `dailyclone.sh`.
- Exécutez la commande `./dailyclone.sh` *SOURCE_USERNAME SOURCE_PASSWORD SOURCE_URL TARGET_USERNAME TARGET_PASSWORD TARGET_URL*. Reportez-vous au tableau précédent pour obtenir une description de ces valeurs.

Groovy
Reportez-vous à la section Exécution de commandes sans installer EPM Automate.

Suppression des fichiers inutiles d'un environnement

Utilisez ces scripts pour enlever les fichiers inutiles d'un environnement.

Les scripts effectuent les étapes suivantes :

- Connexion à l'environnement.
- Enumération des fichiers et instantanés présents dans l'environnement.
- Suppression des fichiers indiqués dans `input.properties`.
- Déconnexion.

Exemple de script Windows

Créez un fichier nommé `removeUnnecessaryFiles.ps1` en copiant le script suivant. Stockez-le dans un répertoire local.

```
$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -raw)
$username="$($inputproperties.username)"
$passwordfile="$($inputproperties.passwordfile)"
$serviceURL="$($inputproperties.serviceURL)"
$file1="$($inputproperties.file1)"
$file2="$($inputproperties.file2)"

epmautomate login ${username} ${passwordfile} ${serviceURL}
epmautomate listfiles
epmautomate deletefile ${file1}
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
epmautomate deletefile ${file2}
epmautomate logout
```

Exemple de script Linux/UNIX

Créez un fichier nommé `removeUnnecessaryFiles.sh` en copiant le script suivant. Stockez-le dans un répertoire local.

```
#!/bin/bash
. ./input.properties
export JAVA_HOME=${javahome}
${epmautomatescript} login "${username}" "${passwordfile}" "${serviceURL}"
${epmautomatescript} listfiles
${epmautomatescript} deletefile "${file1}"
${epmautomatescript} deletefile "${file2}"
${epmautomatescript} logout
```

Création du fichier input.properties

Pour exécuter les scripts `removeUnnecessaryFiles`, créez le fichier `input.properties` et mettez-le à jour avec les informations de votre environnement. Enregistrez le fichier dans le répertoire dans lequel `removeUnnecessaryFiles.ps1` ou `removeUnnecessaryFiles.sh` est stocké.

Windows

```
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
file1=FILE_NAME
file2=FILE_NAME
```

Linux/UNIX

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
file1=FILE_NAME
file2=FILE_NAME
```

Tableau 3-8 Paramètres du fichier input.properties

Paramètre	Description
`javahome`	Emplacement de `JAVA_HOME`. Pour Linux/UNIX uniquement.
`epmautomatescript`	Chemin absolu de l'exécutable EPM Automate (`epmautomate.sh`). Pour Linux/UNIX uniquement.
`username`	Nom d'utilisateur d'un administrateur de service disposant également du rôle Administrateur de domaine d'identité.
`password`	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
`serviceURL`	URL de l'environnement à partir duquel générer l'instantané.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 3-8 (suite) Paramètres du fichier input.properties

Paramètre	Description
`file1` et `file2`	Nom d'un fichier ou d'un instantané à supprimer de l'environnement. Indiquez le chemin et le nom du fichier si ce dernier ne se trouve pas dans la boîte d'envoi.

Exécution du script

1. Créez `removeUnnecessaryFiles.ps1` ou `removeUnnecessaryFiles.sh` en copiant le script d'une section précédente.
2. Créez le fichier `input.properties` et enregistrez-le dans le répertoire où se trouve le script `removeUnnecessaryFiles`. Le contenu de ce fichier varie en fonction du système d'exploitation. Reportez-vous à la section Création du fichier input.properties. Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter le script.
3. Lancez le script.
   - **Windows PowerShell :** exécutez `removeUnnecessaryFiles.ps1`.
   - **Linux/UNIX :** exécutez `./removeUnnecessaryFiles.sh`.

Recherche et téléchargement de fichiers à partir d'un environnement

Utilisez l'exemple de script de cette section pour automatiser le processus de téléchargement de fichiers à partir d'un environnement Oracle Fusion Cloud Enterprise Performance Management à l'aide d'une chaîne de texte comme caractère générique.

Le script suivant vous permet de mettre en correspondance la chaîne que vous spécifiez en tant que valeur du paramètre `FILENAME` avec les noms de fichier affichés à l'aide de la commande `listfiles`, puis de télécharger automatiquement les fichiers qui correspondent à la chaîne.

Veillez à affecter la chaîne de recherche appropriée au paramètre `FILENAME`. Par exemple, `FILENAME="Scheduler Output/epm"` correspondra à la chaîne `Scheduler Output/epm` auprès des noms de fichier dans la sortie de commande `listfiles` dans votre environnement pour identifier les fichiers à télécharger.

Paramètres d'entrée pour l'exécution de ce script : `username`, `password` ou `password_file`, et `service_url`.

> **Remarque**
>
> Si le mot de passe contient des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux.

Windows

```
@echo off
    setlocal EnableExtensions EnableDelayedExpansion
    set USERNAME="username"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    set PASSWORD="password"
    set URL="url"

call epmautomate login %USERNAME% %PASSWORD% %URL%
    set FILENAME="Scheduler Output/epm"
    for /f "tokens=*" %%i in ('epmautomate listfiles ^| findstr /b /r /c:"^
*%FILENAME%" ') do (
    call epmautomate downloadfile "%%i"
    )
call epmautomate logout
endlocal
```

Linux/UNIX

```
#!/bin/sh
    USERNAME="username"
    PASSWORD="password"
    URL="url"

./epmautomate.sh login $USERNAME $PASSWORD $URL
    FILENAME='Scheduler Output/epm'
    #echo $FILENAME
./epmautomate.sh listfiles | grep "^ $FILENAME" | while read -r line ; do
    echo "Processing $line"
    ./epmautomate.sh downloadfile "$line"
    done
./epmautomate.sh logout
```

Recréation d'un ancien environnement Cloud EPM pour les audits

Utilisez le script de cette section pour créer une solution libre-service permettant de maintenir une bibliothèque d'instantanés à jour pour votre environnement Oracle Fusion Cloud Enterprise Performance Management. Vous avez besoin d'un environnement dédié à la mise à niveau et à la maintenance d'une bibliothèque d'instantanés à jour.

Cloud EPM prend en charge la compatibilité des instantanés pour un seul cycle mensuel. Vous pouvez migrer les instantanés de maintenance de l'environnement de test vers l'environnement de production, et inversement. Cependant, les exigences d'audit de certains clients peuvent nécessiter la restauration des instantanés de plusieurs années sur le dernier environnement, ainsi que l'accès à l'application sur une courte période.

Vous devez planifier l'exécution de ce script une fois par mois pour convertir les instantanés disponibles et les rendre compatibles avec le dernier niveau de patch Cloud EPM. Oracle vous recommande d'exécuter le script après le troisième vendredi du mois afin de s'assurer que tous les problèmes de l'environnement de production aient été résolus.

> **Remarque**
>
> Vous ne pouvez pas utiliser ce script pour mettre à jour les instantanés Narrative Reporting, Account Reconciliation et Oracle Fusion Cloud Enterprise Data Management.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Fonctionnement du script

Pour chaque instantané stocké par le client, le script de mise à niveau réalise les tâches suivantes à l'aide d'EPM Automate :

1. Connexion à un environnement à l'aide des informations du fichier `input.properties`.
2. Utilisation de la commande `recreate` pour restaurer l'environnement.
3. Import de l'instantané dans l'environnement.
4. Exécution de la maintenance quotidienne sur l'environnement, qui entraîne la conversion de l'instantané au format compatible avec le niveau de patch Cloud EPM en cours.
5. Téléchargement de l'instantané de maintenance `Artifact Snapshot` dans un dossier. Si vous avez recréé un environnement 18.05 en chargeant des instantanés à partir de `snapshots/18.05`, `Artifact Snapshot` est téléchargé dans `snapshots/18.06`.
6. Envoi par courriel des résultats de la recréation d'anciens environnements à une adresse électronique, si elle est indiquée.

Exécution du script

1. Créez le fichier `input.properties` et mettez-le à jour avec les informations relatives à votre environnement. Enregistrez le fichier dans un répertoire local. Ce répertoire est nommé `parentsnapshotdirectory` dans cette discussion. Le contenu de ce fichier varie en fonction du système d'exploitation.
   Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter les scripts.
2. Créez le script `upgradeSnapshots.ps1` (Windows PowerShell) ou `upgradeSnapshots.sh` (Linux/UNIX), puis enregistrez-le dans le répertoire `parentsnapshotdirectory`, où se trouve `input.properties`.
3. Créez un sous-répertoire, par exemple `snapshots`, dans le répertoire `parentsnapshotdirectory`.
4. Dans le répertoire créé à l'étape précédente (`snapshots`), créez un sous-répertoire pour l'instantané mensuel à convertir afin de le rendre compatible avec le niveau de patch Cloud EPM en cours. Nommez le répertoire en respectant le format `YY.MM`. Par exemple, `18.05` pour le répertoire stockant les instantanés de mai 2018.
5. Copiez les instantanés dans le sous-répertoire approprié. Par exemple, copiez les instantanés de mai 2018 dans `snapshots/18.05`.
6. Lancez le script.
   - Linux/UNIX : exécutez `./upgradeSnapshots.sh`.
   - Windows PowerShell : exécutez `upgradeSnapshots.ps1`.

Windows

Créez le fichier `input.properties` et le script `upgradeSnapshots.ps1` en copiant les scripts de cette section.

Création du fichier `input.properties`

```
username=exampleAdmin
userpassword=examplePassword
serviceurl=exapleURL
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
proxyserverusername=proxyServerUserName
proxyserverpassword=proxyPassword
proxyserverdomain=proxyDoamin
parentsnapshotdirectory=C:/some_directory/snapshots
emailtoaddress=exampleAdmin@oracle.com
```

Mise à jour du fichier `input.properties`

> **Remarque**
>
> Si l'`authentification au niveau du serveur proxy` n'est pas activée pour votre environnement réseau Windows, enlevez les propriétés `proxyserverusername`, `proxyserverpassword` et `proxyserverdomain` du fichier `input.properties`.

Tableau 3-9 Paramètres du fichier input.properties

Paramètre	Description
`username`	Nom d'utilisateur d'un administrateur de service.
`userpassword`	Mot de passe de l'administrateur de service.
`serviceurl`	URL de l'environnement utilisé pour cette activité.
`proxyserverusername`	Nom d'utilisateur permettant d'authentifier une session sécurisée avec le serveur proxy qui contrôle l'accès à Internet.
`proxyserverpassword`	Mot de passe pour l'authentification de l'utilisateur auprès du serveur proxy.
`proxyserverdomain`	Nom du domaine défini pour le serveur proxy.
`parentsnapshotdirectory`	Chemin absolu du répertoire à utiliser en tant que parent du répertoire de stockage des instantanés à traiter. Utilisez la barre oblique (/) comme séparateur de répertoires.
`emailtoaddress`	(Facultatif) Adresse électronique à laquelle les résultats de la recréation d'anciens environnements doivent être envoyés. Les résultats ne sont envoyés par courriel que si cette valeur est indiquée. **Exemple :** `john.doe@example.com`

> **Remarque**
>
> Si votre mot de passe contient des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux.

Création du script `upgradeSnapshots.ps1`

Utilisez cet exemple de script pour créer `upgradeSnapshots.ps1`.

```
# Script for recreating an old Cloud EPM environment

# read in key/value pairs from input.properties file
$inputproperties=ConvertFrom-StringData(Get-Content ./input.properties -raw)

# Global variables
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
$parentsnapshotdirectory="$($inputproperties.parentsnapshotdirectory)"
$username="$($inputproperties.username)"
$userpassword="$($inputproperties.userpassword)"
$serviceurl="$($inputproperties.serviceurl)"
$proxyserverusername="$($inputproperties.proxyserverusername)"
$proxyserverpassword="$($inputproperties.proxyserverpassword)"
$proxyserverdomain="$($inputproperties.proxyserverdomain)"
$emailtoaddress="$($inputproperties.emailtoaddress)"
$operationmessage="EPM Automate operation:"
$operationfailuremessage="EPM Automate operation failed:"
$operationsuccessmessage="EPM Automate operation completed successfully:"
$epmautomatescript="epmautomate.bat"

$workingdir="$pwd"
$logdir="$workingdir/logs/"
$logfile="$logdir/epmautomate-upgradesnapshots.log"

function LogMessage
{
    $message=$args[0]
    $_mydate=$(get-date -f dd_MM_yy_HH_mm_ss)

    echo "[$_mydate] $message" >> $logfile
}

function LogAndEchoMessage
{
    $message=$args[0]
    $_mydate=$(get-date -f dd_MM_yy_HH_mm_ss)

    echo "[$_mydate] $message" | Tee-Object -Append -FilePath $logfile
}

function LogOutput
{
    $_mydate=$(get-date -f dd_MM_yy_HH_mm_ss)
    $op=$args[0]
    $opoutput=$args[1]
    $returncode=$args[2]

    #If error
    if ($returncode -ne 0) {
        $failmessage="[$_mydate] $operationfailuremessage $op"
        LogMessage $failmessage
        LogMessage $opoutput
        LogMessage "return code: $returncode"
    } else {
        $successmessage="[$_mydate] $operationsuccessmessage $op"
        LogMessage $successmessage
        LogMessage $opoutput
        LogMessage "return code: $returncode"
    }
}

function ExecuteCommand
{
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    $op=$args[0]
    $epmautomatecall="$epmautomatescript $op"
    $date=$(get-date -f dd_MM_yy_HH_mm_ss)

    LogMessage "$operationmessage $epmautomatecall"
    $operationoutput=iex "& $epmautomatecall" >> $logfile 2>&1
    LogOutput $op $operationoutput $LastExitCode
}

function ProcessCommand
{
    $command=$args[0]
    $date=$(get-date -f dd_MM_yy_HH_mm_ss)

    if (!([string]::IsNullOrWhitespace($command))) {
        if (!($command.StartsWith("#"))) {
            ExecuteCommand $command
        }
    }
}

function Init
{
    $logdirexists=Test-Path $logdir
    if (!($logdirexists)) {
        mkdir $logdir 2>&1 | out-null
    }

    # removing existing epmautomate debug logs
    rm ./*.log

    $logfileexists=Test-Path $logfile
    # remove existing log file
    if ($logfileexists) {
        rm $logfile
    }
}

function GetNextDate
{
    $latestyearmonth=$args[0]
    LogMessage "latest year.month: $latestyearmonth"
    $latestyear,$latestmonth=$latestyearmonth.split('\.')
    LogMessage "latest year: $latestyear"
    LogMessage "latest month: $latestmonth"
    $intlatestyear=[int]$latestyear
    $intlatestmonth=[int]$latestmonth

    if ($intlatestmonth -eq 12) {
        $intnextmonth=1
        $intnextyear=$intlatestyear+1
    } else {
        $intnextmonth=$intlatestmonth+1
        $intnextyear=$intlatestyear
    }
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    $nextyear="{0:D2}" -f $intnextyear
    $nextmonth="{0:D2}" -f $intnextmonth

    echo "$nextyear.$nextmonth"
}

function ProcessSnapshot
{
    $snapshotfile=$args[0]
    LogMessage "snapshotfile: $snapshotfile"
    $nextdate=$args[1]
    LogMessage "nextdate: $nextdate"
    $snapshotfilename=$snapshotfile.split('/')[-1]
    LogMessage "snapshotfilename: $snapshotfilename"
    $snapshotname=$snapshotfilename.split('.')[0]
    LogMessage "snapshotname: $snapshotname"

    ProcessCommand
"login $username $userpassword $serviceurl $proxyserverusername $proxyserverpa
ssword $proxyserverdomain"
    ProcessCommand "recreate -f"
    ProcessCommand "uploadfile $snapshotfile"
    ProcessCommand "importsnapshot $snapshotname"
    ProcessCommand "runDailyMaintenance skipNext=true -f"
    ProcessCommand "downloadfile 'Artifact Snapshot'"
    ProcessCommand "deletefile $snapshotname"
    ProcessCommand "logout"

    $nextdatedirexists=Test-Path $parentsnapshotdirectory/$nextdate
    if (!($nextdatedirexists)) {
        mkdir $parentsnapshotdirectory/$nextdate 2>&1 | out-null
    }

    LogMessage "Renaming 'Artifact Snapshot.zip' to $snapshotname.zip and
moving to $parentsnapshotdirectory/$nextdate"
    mv $workingdir/'Artifact Snapshot.zip' $workingdir/$snapshotname.zip
>> $logfile 2>&1
    mv $workingdir/$snapshotname.zip $parentsnapshotdirectory/$nextdate
>> $logfile 2>&1
}

function callSendMail
{
    $logfile=$logfile -replace "\\", "/"
    $elements=$logfile.split('/')
    $logfilename=$elements[-1]

    if (${emailtoaddress} -match "@") {
        epmautomate.bat login ${username} ${userpassword} ${serviceurl}
        epmautomate.bat uploadFile "$logfile"
        epmautomate.bat sendMail $emailtoaddress "Recreating An Old Cloud EPM
Environment results" Body="The results of recreating an old Cloud EPM
Environment are attached." Attachments=$logfilename
        epmautomate.bat deleteFile "$logfilename"
        epmautomate.bat logout
    }
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
}

#----- main body of processing
date
Init
LogAndEchoMessage "Starting upgrade snapshots processing"
$snapshotdirs=@(Get-ChildItem -Directory "$parentsnapshotdirectory" -name)
LogMessage "snapshot directories: $snapshotdirs"
$latestreleasedate=$snapshotdirs[-1]
LogMessage "latest release date: $latestreleasedate"
$latestreleasesnapshotdir="$parentsnapshotdirectory/$latestreleasedate"
LogMessage "latest release snapshot dir: $latestreleasesnapshotdir"
$nextdate=$(GetNextDate "$latestreleasedate")
$snapshotfiles=@(Get-ChildItem -File "$latestreleasesnapshotdir")
if ($snapshotfiles.length -eq 0) {
    LogAndEchoMessage "No snapshot files found in
directory $latestreleasesnapshotdir. Exiting script."
    exit
}
foreach ($snapshotfile in $snapshotfiles) {
    LogAndEchoMessage "Processing snapshotfile: $snapshotfile"
    ProcessSnapshot $latestreleasesnapshotdir/$snapshotfile $nextdate
}
LogAndEchoMessage "Upgrade snapshots processing completed"
date
callSendMail
```

Linux/UNIX

Créez `upgradeSnapshots.sh` et `input.properties` en copiant les scripts suivants.

Création du fichier `input.properties` pour Linux/UNIX

> **Remarque**
>
> Si votre réseau n'est pas configuré de façon à utiliser un serveur proxy pour accéder à Internet, enlevez les propriétés `proxyserverusername`, `proxyserverpassword` et `proxyserverdomain` du fichier `input.properties`.

```
username=exampleAdmin
userpassword=examplePassword
serviceurl=exapleURL
proxyserverusername=
proxyserverpassword=
proxyserverdomain=
jdkdir=/home/user1/jdk_17.0.12
epmautomatescript=/home/exampleAdmin/epmautomate/bin/epmautomate.sh
parentsnapshotdirectory=/home/exampleAdmin/some_directory/snapshots
emailtoaddress=exampleAdmin@oracle.com
```

Mise à jour du fichier `input.properties`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 3-10 Paramètres du fichier input.properties

Paramètre	Description
`username`	Nom d'utilisateur d'un administrateur de service.
`userpassword`	Mot de passe de l'administrateur de service.
`serviceurl`	URL de l'environnement utilisé pour cette activité.
`proxyserverusername`	Nom d'utilisateur permettant d'authentifier une session sécurisée avec le serveur proxy qui contrôle l'accès à Internet.
`proxyserverpassword`	Mot de passe pour l'authentification de l'utilisateur auprès du serveur proxy.
`proxyserverdomain`	Nom du domaine défini pour le serveur proxy.
`jdkdir`	Emplacement de `JAVA_HOME`.
`epmautomatescript`	Chemin absolu de l'exécutable EPM Automate (`epmautomate.sh`).
`parentsnapshotdirectory`	Chemin absolu du répertoire à utiliser en tant que parent du répertoire de stockage de l'instantané à traiter.
`emailtoaddress`	(Facultatif) Adresse électronique à laquelle les résultats de la recréation d'anciens environnements doivent être envoyés.

> **Remarque**
>
> Si votre mot de passe contient des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux.

Création du script `upgradeSnapshots.sh`

Utilisez cet exemple de script pour créer `upgradeSnapshots.sh`.

```
#!/bin/sh

. ./input.properties
workingdir=$(pwd)
logdir="${workingdir}/logs"
logfile=epmautomate-upgradesnapshots.log
operationmessage="EPM Automate operation:"
operationfailuremessage="EPM Automate operation failed:"
operationsuccessmessage="EPM Automate operation completed successfully:"
logdebugmessages=true

if [ ! -d ${jdkdir} ]
then
    echo "Could not locate JDK/JRE. Please set value for "jdkdir" property in
input.properties file to a valid JDK/JRE location."
    exit
fi

if [ ! -f ${epmautomatescript} ]
then
    echo "Could not locate EPM Automate script. Please set value for
"epmautomatescript" property in the input.properties file."
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    exit
fi

export JAVA_HOME=${jdkdir}

debugmessage() {
    # logdebugmessages is defined (or not) in testbase input.properties
    if [ "${logdebugmessages}" = "true" ]
    then
        logmessage "$1"
    fi
}

logmessage()
{
    local message=$1
    local _mydate=$(date)

    echo "[$_mydate] ${message}" >> "$logdir/$logfile"
}

echoandlogmessage()
{
    local message=$1
    local _mydate=$(date)

    echo "[$_mydate] ${message}" | tee -a ${logdir}/${logfile}
}

logoutput()
{
    date=`date`
    op="$1"
    opoutput="$2"
    returncode="$3"

    #If error
    #if grep -q "EPMAT-" <<< "$2"
    if [ $returncode -ne 0 ]
    then
        failmessage="[${date}] ${operationfailuremessage} ${op}"
        logmessage "${failmessage}"
        logmessage "${opoutput}"
        logmessage "return code: ${returncode}"
    else
        successmessage="${operationsuccessmessage} ${op}"
        logmessage "${successmessage}"
        logmessage "${opoutput}"
        logmessage "return code: ${returncode}"
    fi
}

getLatestReleaseSnapshotDir()
{
    local snapshotdirs=$(find ${parentsnapshotdirectory} -type d | sort)
    debugmessage "snapshot directories: ${snapshotdirs}"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    local latestreleasesnapshotdir=$(echo ${snapshotdirs##*$\n} | rev | cut -
d' ' -f1 | rev)
    debugmessage "latest release snapshot dir: ${latestreleasesnapshotdir}"
    echo "${latestreleasesnapshotdir}"
}

getNextDate()
{
    local thisyearmonth=$1
    local thisyear=$(echo ${thisyearmonth} | cut -d'.' -f1)
    local thismonth=$(echo ${thisyearmonth} | cut -d'.' -f2)

    intthismonth=$(bc <<< ${thismonth})
    intthisyear=$(bc <<< ${thisyear})

    if [ ${intthismonth} -eq 12 ]
    then
        local intnextmonth=1
        local intnextyear=$((intthisyear+1))
    else
        local intnextmonth=$((intthismonth+1))
        local intnextyear=${intthisyear}
    fi

    nextmonth=$(printf "%02d\n" ${intnextmonth})
    nextyear=$(printf "%02d\n" ${intnextyear})

    debugmessage "next date: ${nextyear}.${nextmonth}"

    echo "${nextyear}.${nextmonth}"
}

init()
{
    if [ ! -d "$logdir" ]
    then
        mkdir $logdir
    fi

    # removing existing epmautomate debug logs
    if ls ./*.log >/dev/null 2>&1
    then
       rm ./*.log
    fi

    # remove existing log files
    if [ -f "${logdir}/${logfile}" ]
    then
        rm ${logdir}/${logfile}
    fi
}

processCommand()
{
    op="$1"
    date=`date`
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    logmessage "$operationmessage $op"
    operationoutput=`eval "$epmautomatescript $op"`
    logoutput "$op" "$operationoutput" "$?"
}

processSnapshot()
{
    local snapshotfile="$1"
    local nextdate="$2"
    local snapshotname=$(echo "${snapshotfile}" | rev | cut -d'/' -f1 | rev |
cut -d'.' -f1)

    processCommand "login ${username} ${userpassword} ${serviceurl} $
{proxyserverusername} ${proxyserverpassword}"
    processCommand "recreate -f"
    processCommand "uploadfile ${snapshotfile}"
    processCommand "importsnapshot \"${snapshotname}\""
    processCommand "runDailyMaintenance skipNext=true -f"
    processCommand "downloadfile \"Artifact Snapshot\""
    processCommand "deletefile \"${snapshotname}\""
    processCommand "logout"

    if [ ! -d ${parentsnapshotdirectory}/${nextdate} ]
    then
        mkdir ${parentsnapshotdirectory}/${nextdate}
    fi
runDailyMaintenance -f
    logmessage "Renaming \"Artifact Snapshot.zip\" to ${snapshotname}.zip and
moving to ${parentsnapshotdirectory}/${nextdate}"
    mv "${workingdir}/Artifact Snapshot.zip" "${workingdir}/$
{snapshotname}.zip" >> "$logdir/$logfile" 2>&1
    mv "${workingdir}/${snapshotname}.zip" ${parentsnapshotdirectory}/$
{nextdate} >> "$logdir/$logfile" 2>&1
}

callSendMail() {

    if [[ "${emailtoaddress}" == *"@"* ]]
    then
        ${epmautomatescript} login ${username} ${userpassword} ${serviceurl}
        ${epmautomatescript} uploadFile "$logdir/$logfile"
        ${epmautomatescript} sendMail $emailtoaddress "Recreating An Old
Cloud EPM Environment results" Body="The results of recreating an old Cloud
EPM Environment are attached" Attachments=$logfile
        ${epmautomatescript} deleteFile "$logfile"
        ${epmautomatescript} logout
    fi
}

#----- main body of processing
date
echoandlogmessage "Starting upgrade snapshots processing"
init
latestreleasesnapshotdir=$(getLatestReleaseSnapshotDir)
latestreleasedate=$(echo "${latestreleasesnapshotdir}" | rev | cut -d'/' -f1
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
| rev)
debugmessage "latest release date: ${latestreleasedate}"
nextdate=$(getNextDate ${latestreleasedate})

snapshotfiles=$(find ${latestreleasesnapshotdir} -type f -name \*.zip | tr
"\n" "|")
if [ ${#snapshotfiles} -eq 0 ]
then
    echoandlogmessage "No snapshot files found in directory $
{latestreleasesnapshotdir}"
fi

IFS="|"
for snapshotfile in $snapshotfiles
do
    echoandlogmessage "Processing snapshotfile: ${snapshotfile}"
    processSnapshot ${snapshotfile} ${nextdate}
done
unset IFS
echoandlogmessage "Upgrade snapshots processing completed."
callSendMail
```

Automatisation de l'audit et de la conformité de l'accès à la base de données

Les scripts PowerShell et d'interpréteur de commandes Bash de cette section vous permettent d'exploiter les commandes EPM Automate en vue de collecter des données d'audit et de conformité concernant l'accès manuel à la base de données.

Vous pouvez utiliser ces scripts pour effectuer les tâches suivantes :

- Télécharger le rapport d'activité du jour en cours
- Analyser le rapport en vue de déterminer si l'accès manuel à la base de données est signalé pour l'environnement
- Créer un fichier `./reports/dataAccessAuditReport.txt` relatif au répertoire à partir duquel exécuter le script. Le rapport répertorie l'heure d'accès à la base de données et la commande SQL exécutée. Il s'agit d'un fichier cumulé qui affiche les dernières informations en haut. Informations disponibles :
  - Date et heure de génération du rapport.
  - Détails de l'accès à la base de données, si disponibles. L'accès à la base de données sans demande de service et l'accès à la base de données avec demande de service sont répertoriés dans des sections distinctes.
    Si l'accès manuel à la base de données n'est pas signalé dans le rapport d'activité, le rapport indique `No SQL statements executed`.
  - Envoyez éventuellement le rapport à l'adresse électronique indiquée.

Pour automatiser l'audit et la conformité de l'accès à la base de données, procédez comme suit :

1. Copiez l'un des scripts des sections suivantes dans un fichier et enregistrez ce dernier dans votre système de fichiers. Nommez le fichier `parseActivityReport.ps1` (Windows, reportez-vous à la section Script PowerShell (parseActivityReport.ps1)) ou

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

`parseActivityReport.sh` (Linux/UNIX, reportez-vous à la section Script d'interpréteur de commandes Bash (parseActivityReport.sh)).

2. **Windows uniquement :** créez un fichier batch nommé `parseActivityReport.bat` en copiant le script suivant dans un fichier. Enregistrez le fichier dans le répertoire où `parseActivityReport.ps1` est stocké.

```
@echo off
set paramRequiredMessage=Syntax: parseActivityReport.bat USERNAME PASSWORD/
PASSWORD_FILE URL [REPORT_EMAIL_TO_ADDRESS]

if "%~1" == "" (
  echo User Name is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )
if "%~2" == "" (
  echo Password or Password_File is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )
if "%~3" == "" (
  echo URL is missing.
  echo %paramRequiredMessage%
  exit /b 1
  )

PowerShell.exe -File parseActivityReport.ps1 %*
```

3. Modifiez `parseActivityReport.bat` (Windows) ou `parseActivityReport.sh` (Linux/UNIX) pour définir les valeurs des paramètres indiqués dans le tableau suivant.

Tableau 3-11 Valeurs de variable à inclure dans les scripts

Variable	Description
`epmuser`	Nom d'utilisateur d'un administrateur de service **Exemples :** Windows : `set epmuser="jDoe"` Linux/UNIX : `epmuser="jDoe"`
`epmpassword`	Mot de passe de l'administrateur de service ou emplacement du fichier de mots de passe crypté. Pour obtenir des informations sur la création d'un fichier de mots de passe cryptés, reportez-vous à la commande encrypt. Si votre mot de passe contient des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux. **Exemples :** Windows : `set epmpassword = "Example"` Linux/UNIX : `epmpassword="Example"`
`epmurl`	URL de l'environnement Oracle Fusion Cloud Enterprise Performance Management. **Exemples :** Windows : `set epmurl="https://example.oraclecloud.com"` Linux/UNIX : `epmurl="https://example.oraclecloud.com"`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 3-11 (suite) Valeurs de variable à inclure dans les scripts

Variable	Description
`report_email_to_address`	(Facultatif) Adresse électronique à laquelle le rapport doit être envoyé. Le rapport n'est envoyé par courriel que si cette valeur est indiquée. **Exemple : `john.doe@example.com`**

4. **Pour `parseActivityReport.sh` uniquement :** assurez-vous que les valeurs suivantes sont correctement définies pour votre système :
   - *JAVA_HOME*
   - Emplacement de `epmautomatescript.sh` en mettant à jour la directive `epmautomatescript`
5. A l'aide d'un planificateur disponible sur le système d'exploitation, planifiez l'exécution de `parseActivityReport.bat` (qui exécute `parseActivityReport.ps1`) ou de `parseActivityReport.sh` une fois par jour. Reportez-vous à la section Automatisation de l'exécution de scripts.

Script PowerShell (parseActivityReport.ps1)

```
# Parse Activity Report script

$epmuser=$args[0]
$epmpassword=$args[1]
$epmurl=$args[2]
$reportemailtoaddress=$args[3]

$logdir="./logs"
$logfile="${logdir}/data_access.log"
$reportdir="./reports"
$reportfile="${reportdir}/dataAccessAuditReport.txt"
$matchfile="${reportdir}/matchfile.txt"
$nosrfile="${reportdir}/data_access_nosr.csv"
$srfile="${reportdir}/data_access_sr.csv"
$aprfilelist="${reportdir}/aprfilelist.txt"
$activityreportfilelist="${reportdir}/activityreportfiles.txt"
$activityreportregex='apr/[0-9]{4}-[0-9]{2}-[0-9]{2} [0-9]{2}_[0-9]{2}_[0-9]
{2}/[0-9]{4}-[0-9]{2}-[0-9]{2} [0-9]{2}_[0-9]{2}_[0-9]{2}.html'

$global:activityreportfile=""

$NO_SQL_EXECUTED_STATEMENT="No SQL statements executed"
$SQL_WITH_SR_EXECUTED_STATEMENT="SQL statements executed with an SR"
$SQL_WITH_NO_SR_EXECUTED_STATEMENT="SQL statements executed without an SR"

function DownloadLatestActivityReport() {
    epmautomate.bat login ${epmuser} ${epmpassword} ${epmurl} >> ${logfile}
    epmautomate.bat listfiles > ${aprfilelist}
    foreach ($line in Get-Content $aprfilelist) {
        if ($line -match $activityreportregex){
            echo "$line" >> $activityreportfilelist
        }
    }
    $global:activityreportfile=Get-Content ${activityreportfilelist} -Tail 1
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    $global:activityreportfile=$global:activityreportfile.trim()
    echo " "
    echo "Processing activity report file: $global:activityreportfile" | tee -
a ${logfile}
    epmautomate.bat downloadfile "$global:activityreportfile" >> ${logfile}
    epmautomate.bat logout >> ${logfile}
}

function deleteLine($file, $start, $end) {
    $i = 0
    $start--
    $end--
    (Get-Content $file) | where{
        ($i -lt $start -or $i -gt $end)
        $i++
    } > $file
    #(Get-Content $file)
}

function GenerateCsvs()
{
    $sqlregex='<DIV id="Database">.*?</DIV>'
    $activityreportfilename=Split-Path $global:activityreportfile -leaf

    echo "Creating CSV file: ${matchfile} from data in activityreportfile: $
{activityreportfilename}" >> ${logfile}
    # remove tab and newline characters
    $activityreportexists=Test-Path "$activityreportfilename"
    if ($activityreportexists) {
        (Get-Content "$activityreportfilename") -join ' ' | Set-Content
"$activityreportfilename"
        (Get-Content "$activityreportfilename") -replace "`t", "" | Set-
Content "$activityreportfilename"
    }

    # capture text matching regex
    $string=Get-Content $activityreportfilename
    $ans=$string -match $sqlregex

    if ($ans -eq "True") {
        $Matches.0 > $matchfile
        # remove HTML tags, etc.
        (Get-Content "$matchfile") -replace "<tr", "`n<tr" | Set-Content
"$matchfile"
        (Get-Content "$matchfile") -replace "<tr[^>]*>", "" | Set-Content
"$matchfile"
        (Get-Content "$matchfile") -replace "<th[^>]*>", "" | Set-Content
"$matchfile"
        (Get-Content "$matchfile") -replace "<td[^>]*>", "|" | Set-Content
"$matchfile"
        (Get-Content "$matchfile") -replace "<br>", "" | Set-Content
"$matchfile"
        (Get-Content "$matchfile") -replace "</td>", "" | Set-Content
"$matchfile"
        (Get-Content "$matchfile") -replace "</tr>", "" | Set-Content
"$matchfile"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        (Get-Content "$matchfile") -replace "\s*</table>\s*</DIV>", "" | Set-
Content "$matchfile"
        deleteLine $matchfile 1 2

        # create SR, NOSR CSV files
        Get-Content $matchfile | ForEach-Object {
            $elements=$_.split('|')
            $timeval=$elements[1].Trim()
            $srval=$elements[3].Trim()
            $sqlval=$elements[4].Trim()

            if (${srval} -eq "") {
                echo "${timeval}|${sqlval}" >> ${nosrfile}
            } else {
                if (${sqlval} -ne "") {
                    echo "${srval}|${timeval}|${sqlval}" >> ${srfile}
                }
            }
        }

    } else { # no SQL statements in activity report
        echo "" >> ${reportfile}
        echo $(date) >> ${reportfile}
        echo "Processing activity report file: $global:activityreportfile"
>> ${reportfile}
        echo "${NO_SQL_EXECUTED_STATEMENT}" | tee -a ${reportfile}
        CleanUp
        EmailReportResults
        exit
    }
}

function ReportResults() {
    echo $(date) >> ${reportfile}
    echo "Processing activity report file: $global:activityreportfile" >> $
{reportfile}
    $srfileexists=Test-Path $srfile
    if ($srfileexists) {
        echo "" | tee -a ${reportfile}
        echo "${SQL_WITH_SR_EXECUTED_STATEMENT}" | tee -a ${reportfile}
        echo "SR#        Time        SQL Statement" | tee -a ${reportfile}
        echo "---        ----        -------------" | tee -a ${reportfile}

        # Loop through csv file and parse
        Get-Content $srfile | ForEach-Object {
            $elements=$_.split('|')
            $srval=$elements[0]
            $timeval=$elements[1]
            $sqlval=$elements[2]
            echo "${srval}    ${timeval}    ${sqlval}" | tee -a ${reportfile}
        }
    }

    $nosrfileexists=Test-Path $nosrfile
    if ($nosrfileexists) {
        echo "" | tee -a ${reportfile}
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        echo "${SQL_WITH_NO_SR_EXECUTED_STATEMENT}" | tee -a ${reportfile}
        echo "Time        SQL Statement" | tee -a ${reportfile}
        echo "----        -------------" | tee -a ${reportfile}

        # Loop through csv file and parse
        Get-Content $nosrfile | ForEach-Object {
            $elements=$_.split('|')
            $timeval=$elements[0]
            $sqlval=$elements[1]
            echo "${timeval}    ${sqlval}" | tee -a ${reportfile}
        }
    }
    EmailReportResults
}

function EmailReportResults
{
  $elements=$reportfile.split('/')
  $reportfilename=$elements[2]

  if (${reportemailtoaddress} -match "@") {
    echo "Emailing Activity Report Results" | tee -a ${logfile}
    epmautomate.bat login ${epmuser} ${epmpassword} ${epmurl} >> ${logfile}
    epmautomate.bat uploadFile $reportfile >> ${logfile}
    epmautomate.bat sendMail $reportemailtoaddress "Database Access Audit
Report Results" Body="Database Access Audit Report Results are attached."
Attachments=$reportfilename >> ${logfile}
    epmautomate.bat deleteFile $reportfilename >> ${logfile}
    epmautomate.bat logout >> ${logfile}
  }
}

function Init
{
    $logdirexists=Test-Path $logdir
    if (!($logdirexists)) {
        mkdir $logdir 2>&1 | out-null
    }

    $reportdirexists=Test-Path $reportdir
    if (!($reportdirexists)) {
        mkdir $reportdir 2>&1 | out-null
    }

    $logfileexists=Test-Path $logfile
    if ($logfileexists) {
        rm $logfile 2>&1 | out-null
    }

    $matchfileexists=Test-Path $matchfile
    if ($matchfileexists) {
        rm $matchfile 2>&1 | out-null
    }

    $nosrfileexists=Test-Path $nosrfile
    if ($nosrfileexists) {
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        rm $nosrfile 2>&1 | out-null
    }

    $srfileexists=Test-Path $srfile
    if ($srfileexists) {
        rm $srfile 2>&1 | out-null
    }

    $aprfilelistexists=Test-Path $aprfilelist
    if ($aprfilelistexists) {
        rm $aprfilelist 2>&1 | out-null
    }

    $activityreportfilelistexists=Test-Path $activityreportfilelist
    if ($activityreportfilelistexists) {
        rm $activityreportfilelist 2>&1 | out-null
    }
}

function CleanUp
{
    $matchfileexists=Test-Path $matchfile
    if ($matchfileexists) {
        rm $matchfile 2>&1 | out-null
    }

    $aprfilelistexists=Test-Path $aprfilelist
    if ($aprfilelistexists) {
        rm $aprfilelist 2>&1 | out-null
    }

    $activityreportfilelistexists=Test-Path $activityreportfilelist
    if ($activityreportfilelistexists) {
        rm $activityreportfilelist 2>&1 | out-null
    }
}

Init
DownloadLatestActivityReport
GenerateCsvs
ReportResults
CleanUp
```

Script d'interpréteur de commandes Bash (parseActivityReport.sh)

```
#!/bin/sh

export JAVA_HOME=/scratch/dteHome/autoWork/jdk_17.0.12
epmautomatescript=/scratch/dteHome/autoWork/epmautomate/19.11.55/bin/
epmautomate.sh

epmuser="<EPM USER>"
epmpwd="<EPM PASSWORD>"
epmurl="<EPM URL>"
reportemailtoaddress="<EMAIL ADDRESS>"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
logdir=./logs
logfile="${logdir}/data_access.log"
reportdir=./reports
reportfile="${reportdir}/dataAccessAuditReport.txt"
nosrfile="${reportdir}/data_access_nosr.csv"
srfile="${reportdir}/data_access_sr.csv"
matchfile="${reportdir}/match.out"
aprfilelist="${reportdir}/aprfilelist.txt"
activityreportfile=""
activityreportregex='apr/[0-9]{4}-[0-9]{2}-[0-9]{2} [0-9]{2}_[0-9]{2}_[0-9]
{2}/[0-9]{4}-[0-9]{2}-[0-9]{2} [0-9]{2}_[0-9]{2}_[0-9]{2}.html'

NO_SQL_EXECUTED_STATEMENT="No SQL statements executed".
SQL_WITH_SR_EXECUTED_STATEMENT="SQL statements executed with an SR"
SQL_WITH_NO_SR_EXECUTED_STATEMENT="SQL statements executed without an SR"

cd "$(dirname "$0")"

generateCsvs()
{
    local sqlregex='<DIV id="Database">.*?</DIV>'
    local activityreportfilename=$(echo "${activityreportfile}" | rev | cut -
d'/' -f1 | rev)

    echo "Creating CSV file: ${matchfile} from data in activityreportfile: $
{activityreportfilename}" >> ${logfile}
    # remove tab and newline characters
    cat "${activityreportfilename}" | tr -d "\t\n\r" > ${matchfile}
    # capture text matching regex
    grep -Po "${sqlregex}" ${matchfile} > ${matchfile}.tmp

    # remove HTML tags, etc.
    sed -e 's/<tr/\n<tr/g' -e 's/<tr[^>]*>//g' -e 's/<th[^>]*>//g' -e 's/
<td[^>]*>/|/g' -e 's/<br>//g' -e 's|</td>||g' -e 's|</tr>||g' -e 's|[ ]*</
table></DIV>||g' -e 's/|[ ]*/|/g' -e 's/[ ]*|/|/g' -e 's/<DIV
id="Database">.*<!-- Print Tables -->\n//g' ${matchfile}.tmp > ${matchfile}

    # create SR, NOSR CSV files
    while read line
    do
        timeval=$(echo "${line}" | cut -d'|' -f2)
        srval=$(echo "${line}" | cut -d'|' -f4)
        sqlval=$(echo "${line}" | cut -d'|' -f5)

        if [[ "${srval}" == "" ]]
        then
            echo "${timeval}|${sqlval}" >> ${nosrfile}
        else
            if [[ "${sqlval}" != "" ]]
            then
                echo "${srval}|${timeval}|${sqlval}" >> ${srfile}
            fi
        fi
    done < ${matchfile}
}
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
reportResults() {
    echo $(date) >> ${reportfile}
    echo "Processing activity report file: $activityreportfile" >> $
{reportfile}
    if [[ -f ${srfile} ]]
    then
        echo "" | tee -a ${reportfile}
        echo "${SQL_WITH_SR_EXECUTED_STATEMENT}" | tee -a ${reportfile}
        echo "SR#        Time        SQL Statement" | tee -a ${reportfile}
        echo "---        ----        -------------" | tee -a ${reportfile}
        while read line
        do
            srval=$(echo "${line}" | cut -d'|' -f1)
            timeval=$(echo "${line}" | cut -d'|' -f2)
            sqlval=$(echo "${line}" | cut -d'|' -f3)
            echo "${srval}    ${timeval}    ${sqlval}" | tee -a ${reportfile}
        done < ${srfile}
    fi

    if [[ -f ${nosrfile} ]]
    then
        echo "" | tee -a ${reportfile}
        echo "${SQL_WITH_NO_SR_EXECUTED_STATEMENT}" | tee -a ${reportfile}
        echo "Time        SQL Statement" | tee -a ${reportfile}
        echo "----        --- ---------" | tee -a ${reportfile}
        while read line
        do
            timeval=$(echo "${line}" | cut -d'|' -f1)
            sqlval=$(echo "${line}" | cut -d'|' -f2)
            echo "${timeval}    ${sqlval}" | tee -a ${reportfile}
        done < ${nosrfile}
    fi

    if [[ ! -f ${srfile} ]] && [[ ! -f ${nosrfile} ]]
    then
        echo "" | tee -a ${reportfile}
        echo "${NO_SQL_EXECUTED_STATEMENT}" | tee -a ${reportfile}
    fi

    emailReportResults
}

downloadLatestActivityReport() {
    ${epmautomatescript} login ${epmuser} ${epmpwd} ${epmurl} >> ${logfile}
    ${epmautomatescript} listfiles > ${aprfilelist}
    activityreportfile=$(cat ${aprfilelist} | grep -P "$
{activityreportregex}" | tail -n 1 | sed -e 's/^ //')
    echo " "
    echo "Processing activity report file: ${activityreportfile}" | tee -a $
{logfile}
    ${epmautomatescript} downloadfile "${activityreportfile}" >> ${logfile}
    ${epmautomatescript} logout >> ${logfile}
}

emailReportResults() {
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    reportfilename=$(echo "${reportfile}" | cut -d'/' -f3)

    if [[ "${reportemailtoaddress}" == *"@"* ]]
    then
        echo "Emailing Activity Report Results" | tee -a ${logfile}
        ${epmautomatescript} login ${epmuser} ${epmpwd} ${epmurl} >> $
{logfile}
        ${epmautomatescript} uploadFile "$reportfile" >> ${logfile}
        ${epmautomatescript} sendMail $reportemailtoaddress "Database Access
Audit Report Results" Body="Database Access Audit Report Results are
attached." Attachments=$reportfilename >> ${logfile}
        ${epmautomatescript} deleteFile "$reportfilename" >> ${logfile}
        ${epmautomatescript} logout >> ${logfile}
    fi
}

checkParams()
{
    if [ -z "$epmuser" ]
    then
        echo "Username is missing."
        echo "Syntax: parseActivityReport.sh USERNAME PASSWORD URL"
        exit 2
    fi

    if [ -z "$epmpwd" ]
    then
        echo "Password is missing."
        echo "Syntax: parseActivityReport.sh USERNAME PASSWORD URL"
        exit 2
    fi

    if [ -z "$epmurl" ]
    then
        echo "URL is missing."
        echo "Syntax: parseActivityReport.sh USERNAME PASSWORD URL"
        exit 2
    fi
}

init()
{
    checkParams

    if [ ! -d "${logdir}" ]
    then
        mkdir ${logdir}
    fi

    if [ ! -d "${reportdir}" ]
    then
        mkdir ${reportdir}
    fi

    if [ ! -f "${epmautomatescript}" ]
    then
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        echo "Cannot locate EPMAutomate script: ${epmautomatescript}. Please
check setting and run script again. Exiting." | tee -a ${logfile}
        exit
    fi

    if [ -f "${srfile}" ]
    then
        rm ${srfile}
    fi

    if [ -f "${nosrfile}" ]
    then
        rm ${nosrfile}
    fi

    if [ -f "${matchfile}" ]
    then
        rm ${matchfile}
    fi

    if [ -f "${aprfilelist}" ]
    then
        rm ${aprfilelist}
    fi
}

cleanup()
{
    if [ -f "${matchfile}" ]
    then
        rm ${matchfile}
    fi

    if [ -f "${matchfile}.tmp" ]
    then
        rm ${matchfile}.tmp
    fi

    if [ -f "${aprfilelist}" ]
    then
        rm ${aprfilelist}
    fi
}

init
downloadLatestActivityReport
generateCsvs
reportResults
cleanup
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Réplication des utilisateurs et des affectations de rôle prédéfini

Les scripts de cette section vous aident à migrer les utilisateurs et les affectations de rôle prédéfini d'un environnement vers un autre.

A propos des scripts

Vous utilisez deux scripts distincts : l'un pour répliquer les utilisateurs dans les domaines d'identité et l'autre pour répliquer les affectations de rôle prédéfini des utilisateurs. Ils s'exécutent dans l'ordre suivant :

- Exécutez le script de réplication des utilisateurs (`replicateusers`) et vérifiez que tous les utilisateurs ont été créés dans le domaine d'identité cible. Pour exécuter ces scripts, l'utilisateur doit posséder les rôles Administrateur de domaine d'identité et Administrateur de service dans les deux environnements.
- Exécutez le script de réplication des affectations de rôle (`replicatepredefinedroles`).

> **Remarque**
>
> - Si les mots de passe contiennent des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux.
> - Les scripts de cette section fonctionnent uniquement pour les rôles prédéfinis : Administrateur de service, Super utilisateur, Utilisateur et Visualiseur.

Exécution des scripts

Pour plus d'informations sur la création des scripts et des fichiers batch requis, reportez-vous aux rubriques suivantes :

- Réplication des utilisateurs d'un domaine d'identité dans un autre
- Réplication des affectations de rôle prédéfini d'un environnement dans un autre

Etapes sous Windows

1. Créez `replicateusers.bat`, `replicateusers.ps1`, `replicatepredefinedroles.bat` et `replicatepredefinedroles.ps1`, puis enregistrez-les dans un répertoire local pour lequel vous disposez de privilèges d'écriture et d'exécution.
2. Mettez à jour les fichiers batch avec les informations relatives aux environnements source et cible, et au serveur proxy Internet, le cas échéant.
3. Exécutez `replicateusers.bat`, qui exécute à son tour `replicateusers.ps1`. Vous devez indiquer le mot de passe par défaut à affecter aux utilisateurs répliqués en tant que paramètre de ligne de commande comme suit :
   `replicateusers.bat Pwd_for_users`

   Si le mot de passe contient des caractères spéciaux, veillez à utiliser le caractère d'échappement approprié. Reportez-vous à la section Gestion des caractères spéciaux.
4. Exécutez `replicatepredefinedroles.bat` afin de créer des affectations de rôle identiques à celles présentes dans l'environnement source.

Etapes sous Linux/UNIX

1. Créez les scripts `replicateusers.sh` et `replicatepredefinedroles.sh`, et enregistrez-les dans un répertoire local pour lequel vous disposez de privilèges d'écriture et d'exécution.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

2. Mettez à jour `replicateusers.sh` et `replicatepredefinedroles.sh` avec les informations relatives aux environnements source et cible, et au serveur proxy Internet, le cas échéant.
3. Exécutez `replicateusers.sh`. Vous devez indiquer le mot de passe par défaut à affecter aux utilisateurs répliqués en tant que paramètre de ligne de commande comme suit : `./replicateusers.sh Pwd_for_users`

   Si le mot de passe contient des caractères spéciaux, veillez à utiliser le caractère d'échappement approprié. Reportez-vous à la section Gestion des caractères spéciaux.
4. Exécutez le script `replicatepredefinedroles.sh` afin de créer des affectations de rôle identiques à celles présentes dans l'environnement source.

Réplication des utilisateurs d'un domaine d'identité dans un autre

Utilisez les scripts de cette section pour cloner les utilisateurs d'un domaine d'identité à un autre domaine d'identité. Pour exécuter ces scripts, l'utilisateur doit posséder les rôles Administrateur de domaine d'identité et Administrateur de service dans les environnements source et cible.

Windows

Créez `replicateusers.bat` et `replicateusers.ps1` en copiant les scripts de cette section.

1. Créez `replicateusers.ps1` en copiant le script suivant :

```
# Replicate users script

param(
  [string]$epmusersource,
  [string]$epmpwdsource,
  [string]$epmurlsource,
  [string]$epmidentitydomainsource,
  [string]$epmusertarget,
  [string]$epmpwdtarget,
  [string]$epmurltarget,
  [string]$epmidentitydomaintarget,
  [string]$proxyserverusername,
  [string]$proxyserverpassword,
  [string]$proxyserverdomain,
  [string]$userpassword,
  [string]$resetpassword,
  [string]$emailtoaddress
)

$roleassignmentreport="roleassignmentreport.csv"
$usersreport="users.csv"

echo "Replicate users script started"


# delete existing reports
$roleassignmentreportexists=Test-Path $roleassignmentreport
if ($roleassignmentreportexists) {
    rm $roleassignmentreport 2>&1 | out-null
}

$usersreportexists=Test-Path $usersreport
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
if ($usersreportexists) {
    rm $usersreport 2>&1 | out-null
}

# epmautomate login Source App as an IDM Admin
echo "Logging into source application at ${epmurlsource}"
epmautomate login ${epmusersource} ${epmpwdsource} ${epmurlsource} $
{epmidentitydomainsource} ${proxyserverusername} ${proxyserverpassword} $
{proxyserverdomain}
echo "Creating role assignment report: ${roleassignmentreport}"
epmautomate roleAssignmentReport ${roleassignmentreport}
if (${emailtoaddress} -match "@") {
    epmautomate.bat sendMail $emailtoaddress "Role assignment report"
Body="Role assignment report is attached."
Attachments=$roleassignmentreport}
echo "Downloading role assignment report"
epmautomate downloadfile ${roleassignmentreport}
epmautomate deletefile ${roleassignmentreport}
epmautomate logout

# Create users report
Get-Content ${roleassignmentreport} | ForEach-Object {
    $user=$_.split(',')[0]
    $firstname=$_.split(',')[1]
    $lastname=$_.split(',')[2]
    $email=$_.split(',')[3]

    if ($firstname -eq "First Name") {
        return
    } else {
        echo "${firstname},${lastname},${email},${user}" >> ${usersreport}
    }

}

Get-Content -Path "${usersreport}" | Sort-Object -Unique > "$
{usersreport}.tmp"
mv -Force "${usersreport}.tmp" "${usersreport}"
$userheader="First Name,Last Name,Email,User Login"
"${userheader}`r`n" + (Get-Content $usersreport -Raw) | Set-
Content $usersreport

# epmautomate login Target App as an IDM Admin
echo "Logging into target application at ${epmurltarget}"
epmautomate login ${epmusertarget} ${epmpwdtarget} ${epmurltarget} $
{epmidentitydomaintarget} ${proxyserverusername} ${proxyserverpassword} $
{proxyserverdomain}
epmautomate deletefile ${usersreport} | Out-Null
echo "Uploading file ${usersreport}"
epmautomate uploadfile ${usersreport}
echo "Adding users"
epmautomate addUsers ${usersreport} userPassword=${userpassword}
resetPassword=${resetpassword}
epmautomate deletefile ${usersreport}
epmautomate logout
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
rm deletefile*.log | Out-Null
echo "Replicate users script completed"
```

2. Créez `replicateusers.bat` en copiant le script suivant :

```
@ECHO OFF
SET thisdir=%~dp0
SET scriptpath=%thisdir%replicateusers.ps1
SET paramRequiredMessage=Syntax: replicateusers.bat "USER_PASSWORD"

REM USER DEFINED VARIABLES
REM -----------------------
set epmusersource="<EPM USER FOR SOURCE ENVIRONMENT>"
set epmpwdsource="<EPM PASSWORD FOR SOURCE ENVIRONMENT>"
set epmurlsource="<EPM URL FOR SOURCE ENVIRONMENT>"
set epmidentitydomainsource="<EPM IDENTITY DOMAIN FOR SOURCE ENVIRONMENT>"
set epmusertarget="<EPM USER FOR TARGET ENVIRONMENT>"
set epmpwdtarget="<EPM PASSWORD FOR TARGET ENVIRONMENT>"
set epmurltarget="<EPM URL FOR TARGET ENVIRONMENT>"
set epmidentitydomaintarget="<EPM IDENTITY DOMAIN FOR TARGET ENVIRONMENT>"
set proxyserverusername="<PROXY SERVER USER NAME>"
set proxyserverpassword="<PROXY SERVER PASSWORD>"
set proxyserverdomain="<PROXY SERVER DOMAIN>"
set resetpassword=false
set emailtoaddress="<EMAIL_TO_ADDRESS>"
REM -----------------------

if "%~1" == "" (
        echo USER_PASSWORD is missing. This is used to set the default
password for the replicated users.
        echo %paramRequiredMessage%
        exit /b 1
  )

PowerShell -NoProfile -ExecutionPolicy Bypass -Command "& '%scriptpath%' -
epmusersource '%epmusersource%' -epmpwdsource '%epmpwdsource%' -
epmurlsource '%epmurlsource%' -epmidentitydomainsource
'%epmidentitydomainsource%' -epmusertarget '%epmusertarget%' -epmpwdtarget
'%epmpwdtarget%' -epmurltarget '%epmurltarget%' -epmidentitydomaintarget
'%epmidentitydomaintarget%' -proxyserverusername '%proxyserverusername%' -
proxyserverpassword '%proxyserverpassword%' -proxyserverdomain
'%proxyserverdomain%' -userpassword '%~1' -resetpassword '%resetpassword%'
-emailtoaddress '%emailtoaddress%'"
```

3. Mettez à jour `replicateusers.bat`. Reportez-vous au tableau suivant pour connaître les valeurs à indiquer.

Paramètre	Description
`epmusersource`	Nom de l'utilisateur doté des rôles Administrateur de domaine d'identité et Administrateur de service dans l'environnement source. **Exemples :** Windows : `set epmusersource="jDoe"` Linux/UNIX : `epmusersource="jDoe"`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Paramètre	Description
epmpwdsource	Mot de passe de l'utilisateur ou chemin absolu du fichier de mots de passe crypté. **Exemples :** Windows : `set epmpwdsource="Example"` Linux/UNIX : `epmpwdsource="Example"`
epmurlsource	URL de l'environnement à partir duquel les utilisateurs doivent être copiés. **Exemples :** Windows : `set epmurlsource="https://example.oraclecloud.com"` Linux/UNIX : `epmurlsource="https://example.oraclecloud.com"`
epmidentitydomainsource	Nom du domaine d'identité utilisé par l'environnement source. **Exemples :** Windows : `set epmidentitydomainsource="example_source_dom"` Linux/UNIX : `epmidentitydomainsource="example_source_dom"`
epmusertarget	Nom de l'utilisateur doté des rôles Administrateur de domaine d'identité et Administrateur de service dans l'environnement cible. **Exemples :** Windows : `set epmusertarget="John.Doe"` Linux/UNIX : `set epmusertarget="John.Doe"`
epmpwdtarget	Mot de passe de l'utilisateur ou chemin absolu du fichier de mots de passe crypté. **Exemples :** Windows : `set epmpwdtarget="Example1"` Linux/UNIX : `epmpwdtarget="Example1"`
epmurltarget	URL de l'environnement dans lequel les utilisateurs doivent être créés. **Exemples :** Windows : `set epmurltarget="https://example.oraclecloud.com"` Linux/UNIX : `epmurltarget="https://example.oraclecloud.com"`
epmidentitydomaintarget	Nom du domaine d'identité utilisé par l'environnement cible. **Exemples :** Windows : `set epmidentitydomaintarget="example_source_dom"` Linux/UNIX : `epmidentitydomaintarget="example_target_dom"`
proxyserverusername	Nom d'utilisateur permettant d'authentifier une session sécurisée avec le serveur proxy qui contrôle l'accès à Internet. Supprime toutes les occurrences de cette propriété si elle n'est pas utilisée. **Exemples :** Windows : `set proxyserverusername="Example"` Linux/UNIX : `proxyserverusername="Example"`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Paramètre	Description
`proxyserverpassword`	Mot de passe pour l'authentification de l'utilisateur auprès du serveur proxy. Supprime toutes les occurrences de cette propriété si elle n'est pas utilisée. **Exemples :** Windows : `set proxyserverpassword="examplePwd"` Linux/UNIX : `proxyserverpassword="examplePwd"`
`proxyserverdomain`	Nom du domaine défini pour le serveur proxy. Supprime toutes les occurrences de cette propriété si elle n'est pas utilisée. **Exemples :** Windows : `set proxyserverdomain="exampleDom"` Linux/UNIX : `proxyserverdomain="exampleDom"`
`emailtoaddress`	(Facultatif) Adresse électronique à laquelle le rapport sur l'affectation de rôle doit être envoyé. Le rapport n'est envoyé par courriel que si cette valeur est indiquée. **Exemple :** `emailtoaddress=john.doe@example.com`

Linux/UNIX

1. Créez `replicateusers.sh` en copiant le script suivant.

```
#!/bin/sh

userpassword="$1"

# USER DEFINED VARIABLES
#-----------------------
javahome="<JAVA HOME>"
epmautomatescript="<EPM AUTOMATE SCRIPT LOCATION>"
epmusersource="<EPM USER FOR SOURCE ENVIRONMENT>"
epmpwdsource="<EPM PASSWORD FOR SOURCE ENVIRONMENT>"
epmurlsource="<EPM URL FOR SOURCE ENVIRONMENT>"
epmidentitydomainsource="<EPM IDENTITY DOMAIN FOR SOURCE ENVIRONMENT>"
epmusertarget="<EPM USER FOR TARGET ENVIRONMENT>"
epmpwdtarget="<EPM PASSWORD FOR TARGET ENVIRONMENT>"
epmurltarget="<EPM URL FOR TARGET ENVIRONMENT>"
epmidentitydomaintarget="<EPM IDENTITY DOMAIN FOR TARGET ENVIRONMENT>"
proxyserverusername="<PROXY SERVER USER NAME>"
proxyserverpassword="<PROXY SERVER PASSWORD>"
proxyserverdomain="<PROXY SERVER DOMAIN>"
resetpassword="false"
emailtoaddress="<EMAIL TO ADDRESS>"
#-----------------------

roleassignmentreport="roleassignmentreport.csv"
usersreport="users.csv"
paramrequiredmessage='Syntax: replicateusers.sh "USER_PASSWORD"'

export JAVA_HOME=${javahome}

if [ "${userpassword}" == "" ]
then
    echo "USER_PASSWORD is missing. This is used to set the default
password for the replicated users."
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    echo "${paramrequiredmessage}"
    exit
fi

echo "Replicate users script started"

# epmautomate login Source App as an IDM Admin
echo "Logging into source application at ${epmurlsource}"
${epmautomatescript} login ${epmusersource} ${epmpwdsource} $
{epmurlsource} ${epmidentitydomainsource} ${proxyserverusername} $
{proxyserverpassword} ${proxyserverdomain}
echo "Creating role assignment report: ${roleassignmentreport}"
${epmautomatescript} roleAssignmentReport ${roleassignmentreport}
if [[ "${emailtoaddress}" == *"@"* ]]
then
    ${epmautomatescript} sendMail $emailtoaddress "Role assignment report"
Body="Role assignment report is attached."
Attachments=$roleassignmentreport
fi
echo "Downloading role assignment report"
${epmautomatescript} downloadfile ${roleassignmentreport}
${epmautomatescript} deletefile ${roleassignmentreport}
${epmautomatescript} logout

awk -F, '{print $2","$3","$4","$1}' ${roleassignmentreport} | (read -r;
printf "%s\n" "$REPLY"; sort -u) > ${usersreport}

# epmautomate login Target App as an IDM Admin
echo "Logging into target application at ${epmurltarget}"
${epmautomatescript} login ${epmusertarget} ${epmpwdtarget} $
{epmurltarget} ${epmidentitydomaintarget} ${proxyserverusername} $
{proxyserverpassword} ${proxyserverdomain}
${epmautomatescript} deletefile ${usersreport} > /dev/null 2>&1
echo "Uploading file ${usersreport}"
${epmautomatescript} uploadfile ${usersreport}
echo "Adding users"
${epmautomatescript} addUsers ${usersreport} userPassword=${userpassword}
resetPassword=${resetpassword}
${epmautomatescript} deletefile ${usersreport}
${epmautomatescript} logout
rm deletefile*.log > /dev/null 2>&1

echo "Replicate users script completed"
```

2. Mettez à jour `replicateusers.sh`. Reportez-vous au tableau précédent pour obtenir plus d'informations sur les valeurs à indiquer. Vous pouvez également indiquer les valeurs pour ces propriétés :
   - `javahome` : chemin absolu du répertoire dans lequel Java est installé.
   - `epmautomatescript` : emplacement d'`epmautomatescript.sh`. Par exemple : `epmautomatescript="/home/user1/epmautomate/bin/epmautomate.sh"`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Réplication des affectations de rôle prédéfini d'un environnement dans un autre

Utilisez les scripts de cette section pour cloner les affectations de rôle prédéfini d'un environnement dans un autre. Pour exécuter ces scripts, l'utilisateur doit posséder le rôle Administrateur de service dans les deux environnements.

> **Remarque**
>
> **Si vous utilisez la version PDF de ce document :** pour éviter les retours à la ligne et les informations de pied de page qui rendent ces scripts inutilisables, copiez-les à partir de la version HTML de cette rubrique.

Windows

1. Créez `replicatepredefineroles.ps1` en copiant le script suivant.

```
# Replicate predefined roles script

param(
  [string]$epmusersource,
  [string]$epmpwdsource,
  [string]$epmurlsource,
  [string]$epmidentitydomainsource,
  [string]$epmusertarget,
  [string]$epmpwdtarget,
  [string]$epmurltarget,
  [string]$epmidentitydomaintarget,
  [string]$proxyserverusername,
  [string]$proxyserverpassword,
  [string]$proxyserverdomain,
  [string]$emailtoaddress
)

$roleassignmentreport="roleassignmentreport.csv"

function replicateroles
{
    # epmautomate login Source App as an IDM Admin
    echo "Logging into source application at ${epmurlsource}"
    epmautomate login ${epmusersource} ${epmpwdsource} ${epmurlsource} $
{epmidentitydomainsource} ${proxyserverusername} ${proxyserverpassword} $
{proxyserverdomain}
    echo "Creating role assignment report: ${roleassignmentreport}"
    epmautomate roleAssignmentReport ${roleassignmentreport}
    if (${emailtoaddress} -match "@") {
        epmautomate.bat sendMail $emailtoaddress "Role assignment report"
Body="Role assignment report is attached."
Attachments=$roleassignmentreport
    }
    echo "Downloading role assignment report"
    epmautomate downloadfile ${roleassignmentreport}
    epmautomate deletefile ${roleassignmentreport}
    epmautomate logout
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    echo "Creating files to use with epmautomate assignRoles"

    Get-Content ${roleassignmentreport} | ForEach-Object {
        $user=$_.split(',')[0]
        $rolename=$_.split(',')[4]

        if ($rolename -like '*User' -And $rolename -notlike '*Power User')
{
            $rolenamearray=$rolename.split(" ")
            $arraysize=$rolenamearray.count
            $rolename="User"
            if ($arraysize.count -le 2) {
                echo "${user}" | Out-File -Append -Encoding "UTF8" "role-$
{rolename}.csv"
            }
        }
        elseif ($rolename -like '*Viewer') {
            $rolenamearray=$rolename.split(" ")
            $arraysize=$rolenamearray.count
            $rolename="Viewer"
            if ($arraysize -le 2) {
                echo "${user}" | Out-File -Append -Encoding "UTF8" "role-$
{rolename}.csv"
            }
        }
        elseif ($rolename -like '*Power User') {
            $rolenamearray=$rolename.split(" ")
            $arraysize=$rolenamearray.count
            $rolename="Power User"
            if ($arraysize -le 3) {
                echo "${user}" | Out-File -Append -Encoding "UTF8" "role-$
{rolename}.csv"
            }
        }
        elseif ($rolename -like '*Service Administrator') {
            $rolenamearray=$rolename.split(" ")
            $arraysize=$rolenamearray.count
            $rolename="Service Administrator"
            if ($arraysize -le 3) {
                echo "${user}" | Out-File -Append -Encoding "UTF8" "role-$
{rolename}.csv"
            }
        }
        elseif ($rolename -like 'Planner') {
            echo "${user}" | Out-File -Append -Encoding "UTF8" "role-
User.csv"
        }
    }

    # Add header and format
    $rolefiles = Get-ChildItem "role-*.csv"
    foreach ($rolefile in $rolefiles) {
        $rolefilecontent = Get-Content "$rolefile"
        $headerline='User Login'
        Set-Content $rolefile -value $headerline,$rolefilecontent
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        $txt = [io.file]::ReadAllText("$rolefile") -replace "`r`n","`n"
        [io.file]::WriteAllText("$rolefile", $txt)
    }

    # epmautomate login Target App as an IDM Admin
    echo "Logging into target application at ${epmurltarget}"
    epmautomate login ${epmusertarget} ${epmpwdtarget} ${epmurltarget} $
{epmidentitydomaintarget} ${proxyserverusername} ${proxyserverpassword} $
{proxyserverdomain}

    $rolefiles = Get-ChildItem "role-*.csv"
    foreach ($rolefile in $rolefiles) {
        $rolenamecsv=$rolefile.BaseName.split('-')[1]
        $rolename=$rolenamecsv.split('.')[0]
        epmautomate deletefile "${rolefile}" | Out-Null
        echo "Uploading file ${rolefile}"
        epmautomate uploadfile "${rolefile}"
        echo "Assigning ${rolename} roles"
        epmautomate assignRole "role-${rolename}.csv" "${rolename}"
        epmautomate deletefile "role-${rolename}.csv"
    }
    epmautomate logout
    rm deletefile*.log | Out-Null
}

function init
{
    # delete ${role}.csv files
    $rolefiles = Get-ChildItem "role-*.csv"
    foreach ($rolefile in $rolefiles) {
        $rolefileexists=Test-Path $rolefile
        if ($rolefileexists) {
            rm "${rolefile}"
        }
    }
}

echo "Replicate predefined roles script started"
init
replicateroles
echo "Replicate predefined roles script completed"
```

2. Créez `replicatepredefineroles.bat` en copiant le script suivant.

```
@ECHO OFF
SET thisdir=%~dp0
SET scriptpath=%thisdir%replicatepredefinedroles.ps1

REM USER DEFINED VARIABLES
REM -----------------------
set epmusersource="<EPM USER FOR SOURCE ENVIRONMENT>"
set epmpwdsource="<EPM PASSWORD FOR SOURCE ENVIRONMENT>"
set epmurlsource="<EPM URL FOR SOURCE ENVIRONMENT>"
set epmidentitydomainsource="<EPM IDENTITY DOMAIN FOR SOURCE ENVIRONMENT>"
set epmusertarget="<EPM USER FOR TARGET ENVIRONMENT>"
set epmpwdtarget="<EPM PASSWORD FOR TARGET ENVIRONMENT>"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
set epmurltarget="<EPM URL FOR TARGET ENVIRONMENT>"
set epmidentitydomaintarget="<EPM IDENTITY DOMAIN FOR TARGET ENVIRONMENT>"
set proxyserverusername="<PROXY SERVER USER NAME>"
set proxyserverpassword="<PROXY SERVER PASSWORD>"
set proxyserverdomain="<PROXY SERVER DOMAIN>"
set emailtoaddress="<EMAIL_TO_ADDRESS>"
REM -----------------------

PowerShell -NoProfile -ExecutionPolicy Bypass -Command "& '%scriptpath%' -
epmusersource '%epmusersource%' -epmpwdsource '%epmpwdsource%' -
epmurlsource '%epmurlsource%' -epmidentitydomainsource
'%epmidentitydomainsource%' -epmusertarget '%epmusertarget%' -epmpwdtarget
'%epmpwdtarget%' -epmurltarget '%epmurltarget%' -epmidentitydomaintarget
'%epmidentitydomaintarget%' -proxyserverusername '%proxyserverusername%' -
proxyserverpassword '%proxyserverpassword%' -proxyserverdomain
'%proxyserverdomain%' -emailtoaddress '%emailtoaddress%'"
```

3. Mettez à jour `replicatepredefineroles.bat`, si nécessaire. Reportez-vous au tableau suivant afin d'obtenir des informations sur les valeurs à définir pour les propriétés de ce fichier.

Mise à jour de `replicatepredefineroles.bat`

Paramètre	Description
`epmusersource`	Nom de l'utilisateur doté des rôles Administrateur de domaine d'identité et Administrateur de service dans l'environnement source. **Exemples :** Windows : `set epmusersource="jDoe"` Linux/UNIX : `epmusersource="jDoe"`
`epmpwdsource`	Mot de passe de l'utilisateur ou chemin absolu du fichier de mots de passe crypté. **Exemples :** Windows : `set epmpwdsource="Example"` Linux/UNIX : `epmpwdsource="Example"`
`epmurlsource`	URL de l'environnement à partir duquel les utilisateurs doivent être copiés. **Exemples :** Windows : `set epmurlsource="https://example.oraclecloud.com"` Linux/UNIX : `epmurlsource="https://example.oraclecloud.com"`
`epmidentitydomainsource`	Nom du domaine d'identité utilisé par l'environnement source. **Exemples :** Windows : `set epmidentitydomainsource="example_source_dom"` Linux/UNIX : `epmidentitydomainsource="example_source_dom"`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Paramètre	Description
epmusertarget	Nom de l'utilisateur doté des rôles Administrateur de domaine d'identité et Administrateur de service dans l'environnement cible. **Exemples :** Windows : `set epmusertarget="John.Doe"` Linux/UNIX : `set epmusertarget="John.Doe"`
epmpwdtarget	Mot de passe de l'utilisateur ou chemin absolu du fichier de mots de passe crypté. **Exemples :** Windows : `set epmpwdtarget="Example1"` Linux/UNIX : `epmpwdtarget="Example1"`
epmurltarget	URL de l'environnement dans lequel les utilisateurs doivent être créés. **Exemples :** Windows : `set epmurltarget="https://example.oraclecloud.com"` Linux/UNIX : `epmurltarget="https://example.oraclecloud.com"`
epmidentitydomaintarget	Nom du domaine d'identité utilisé par l'environnement cible. **Exemples :** Windows : `set epmidentitydomaintarget="example_target_dom"` Linux/UNIX : `epmidentitydomaintarget="example_target_dom"`
proxyserverusername	Nom d'utilisateur permettant d'authentifier une session sécurisée avec le serveur proxy qui contrôle l'accès à Internet. Supprime toutes les occurrences de cette propriété si elle n'est pas utilisée. **Exemples :** Windows : `set proxyserverusername="Example"` Linux/UNIX : `proxyserverusername="Example"`
proxyserverpassword	Mot de passe pour l'authentification de l'utilisateur auprès du serveur proxy. Supprime toutes les occurrences de cette propriété si elle n'est pas utilisée. **Exemples :** Windows : `set proxyserverpassword="examplePwd"` Linux/UNIX : `proxyserverpassword="examplePwd"`
proxyserverdomain	Nom du domaine défini pour le serveur proxy. Supprime toutes les occurrences de cette propriété si elle n'est pas utilisée. **Exemples :** Windows : `set proxyserverdomain="exampleDom"` Linux/UNIX : `proxyserverdomain="exampleDom"`
emailtoaddress	(Facultatif) Adresse électronique à laquelle le rapport sur l'affectation de rôle doit être envoyé. Le rapport n'est envoyé par courriel que si cette valeur est indiquée. **Exemple :** `emailtoaddress=john.doe@example.com`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Linux/UNIX

1. Créez `replicatepredefineroles.sh` en copiant le script suivant.

```
#!/bin/sh

# USER DEFINED VARIABLES
#-----------------------
javahome="<JAVA HOME>"
epmautomatescript="<EPM AUTOMATE SCRIPT LOCATION>"
epmusersource="<EPM USER FOR SOURCE ENVIRONMENT>"
epmpwdsource="<EPM PASSWORD FOR SOURCE ENVIRONMENT>"
epmurlsource="<EPM URL FOR SOURCE ENVIRONMENT>"
epmidentitydomainsource="<EPM IDENTITY DOMAIN FOR SOURCE ENVIRONMENT>"
epmusertarget="<EPM USER FOR TARGET ENVIRONMENT>"
epmpwdtarget="<EPM PASSWORD FOR TARGET ENVIRONMENT>"
epmurltarget="<EPM URL FOR TARGET ENVIRONMENT>"
epmidentitydomaintarget="<EPM IDENTITY DOMAIN FOR TARGET ENVIRONMENT>"
proxyserverusername="<PROXY SERVER USER NAME>"
proxyserverpassword="<PROXY SERVER PASSWORD>"
proxyserverdomain="<PROXY SERVER DOMAIN>"
emailtoaddress="<EMAIL TO ADDRESS>"
#-----------------------

roleassignmentreport="roleassignmentreport.csv"

export JAVA_HOME=${javahome}


replicateroles()
{
    # epmautomate login Source App as an DM Admin
    echo "Logging into source application at ${epmurlsource}"
    ${epmautomatescript} login ${epmusersource} ${epmpwdsource} $
{epmurlsource} ${epmidentitydomainsource} ${proxyserverusername} $
{proxyserverpassword} ${proxyserverdomain}
    echo "Creating role assignment report: ${roleassignmentreport}"
    ${epmautomatescript} roleAssignmentReport ${roleassignmentreport}
    if [[ "${emailtoaddress}" == *"@"* ]]
    then
        ${epmautomatescript} sendMail $emailtoaddress "Role assignment
report" Body="Role assignment report is attached."
Attachments=$roleassignmentreport
    fi
    echo "Downloading role assignment report"
    ${epmautomatescript} downloadfile ${roleassignmentreport}
    ${epmautomatescript} deletefile ${roleassignmentreport}
    ${epmautomatescript} logout

    echo "Creating files to use with epmautomate assignRoles"
    while read line
    do
        user=$(echo "${line}" | cut -d',' -f1)
        rolename=$(echo "${line}" | cut -d',' -f5)
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        if [[ "$rolename" == *"User" ]] && [[ "$rolename" != "*Power
User" ]]
        then
            count=$(echo "${rolename}" | wc -w);
            rolename="User"
            if [[ $count -le 2 ]]
            then
                echo "${user}" >> "role-${rolename}.csv"
            fi
        elif [[ "$rolename" == *"Viewer" ]]
        then
            count=$(echo "${rolename}" | wc -w);
            rolename="Viewer"
            if [[ $count -le 2 ]]
            then
                echo "${user}" >> "role-${rolename}.csv"
            fi
        elif [[ "$rolename" == *"Power User" ]]
        then
            count=$(echo "${rolename}" | wc -w);
            rolename="Power User"
            if [[ $count -le 3 ]]
            then
                echo "${user}" >> "role-${rolename}.csv"
            fi
        elif [[ "$rolename" == *"Service Administrator" ]]
        then
            count=$(echo "${rolename}" | wc -w);
            rolename="Service Administrator"
            if [[ $count -le 3 ]]
            then
                echo "${user}" >> "role-${rolename}.csv"
            fi
        elif [[ "$rolename" == "Planner" ]]
        then
                echo "${user}" >> "role-User.csv"
        fi
    done < ${roleassignmentreport}

    # write header line
    for f in role-*.csv
    do
        sed -i '1iUser Login' "$f"
    done

    # epmautomate login Target App as an IDM Admin
    echo "Logging into target application at ${epmurltarget}"
    ${epmautomatescript} login ${epmusertarget} ${epmpwdtarget} $
{epmurltarget} ${epmidentitydomaintarget} ${proxyserverusername} $
{proxyserverpassword} ${proxyserverdomain}

    for rolefile in role-*.csv
    do
        rolenamecsv=$(echo "$rolefile" | cut -d'-' -f2)
        rolename=$(echo "$rolenamecsv" | cut -d'.' -f1)
        ${epmautomatescript} deletefile "${rolefile}" > /dev/null 2>&1
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        echo "Uploading file ${rolefile}"
        ${epmautomatescript} uploadfile "${rolefile}"
        echo "Assigning roles"
        ${epmautomatescript} assignrole "${rolefile}" "${rolename}"
        ${epmautomatescript} deletefile "${rolefile}"
    done

    ${epmautomatescript} logout
    rm deletefile*.log > /dev/null 2>&1
}

init()
{
    # delete role-${role}.csv files
    for f in role-*.csv
    do
        rm "$f" > /dev/null 2>&1
    done
}

echo "Replicate predefined roles script started"
init
replicateroles
echo "Replicate predefined roles script completed"
```

2. Mettez à jour `replicatepredefineroles.sh`. Reportez-vous au tableau précédent pour obtenir plus d'informations sur les valeurs à indiquer. Vous pouvez également indiquer les valeurs pour ces propriétés :
   - `javahome` : chemin absolu du répertoire dans lequel Java est installé.
   - `epmautomatescript` : emplacement d'`epmautomatescript.sh`. Par exemple : `epmautomatescript="/home/user1/epmautomate/bin/epmautomate.sh"`

Création d'une cadence de mise à niveau Cloud EPM trimestrielle

Utilisez ces scripts pour créer une solution libre-service permettant d'ignorer les mises à jour de sorte que les environnements Cloud EPM soient mis à jour chaque trimestre avec un cycle de test de deux semaines. Dans ce cas, les environnements de production sont mis à jour deux semaines après les environnements de test.

Ce script peut également être utilisé pour ignorer une mise à jour mensuelle sur deux, si nécessaire. Par défaut, Cloud EPM applique une mise à jour mensuelle à vos environnements. Vous utilisez la commande skipUpdate pour ignorer l'application de mises à jour mensuelles à un environnement ou pour consulter les demandes de non-prise en compte de mise à jour en cours. Vous pouvez automatiser l'exécution manuelle des commandes `skipUpdate` à l'aide des scripts inclus dans cette section. Ces scripts automatisent le processus de non-prise en charge de mise à jour de sorte que les mises à jour soient appliquées chaque trimestre ou un mois sur deux.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Note

1. Vous ne pouvez pas ignorer les mises à jour pendant plus de trois mois consécutifs. Par exemple, le script génère une erreur si vous essayez de mettre à jour un environnement Cloud EPM uniquement en février, en juillet et en décembre.
2. Toutes les mises à jour effectuées lors de la période d'intervention seront appliquées à votre environnement lors de la prochaine mise à jour. Par exemple, supposons que vous utilisiez ce script pour planifier des mises à jour trimestrielles qui n'auront lieu qu'en février, en mai, en août et en novembre. Dans ce cas, la mise à jour de mai, par exemple, appliquera à votre environnement l'ensemble des mises à jour et des patches Cloud EPM mensuels applicables publiés après la mise à jour de février. Le processus de maintenance peut prendre plus de temps que d'habitude une fois la mise à jour appliquée.
3. Ce script configure la cadence de mise à jour pour un trimestre seulement. Exécutez ce script chaque mois pour vous assurer que la cadence de mise à jour est configurée pour toute l'année.

- Instructions et script Windows
- Instructions et script UNIX/Linux
- Script Groovy

Exécution du script

1. Pour exécuter des scripts Windows et Linux/UNIX, procédez comme suit :
   a. Créez le fichier `input.properties` et mettez-le à jour avec les informations relatives à votre environnement. Enregistrez le fichier dans un répertoire local. Le contenu de ce fichier varie en fonction du système d'exploitation.
   Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter le script.
   b. Créez un script PowerShell `skip_update.ps1` (Windows) ou Bash `skip_update.sh` (Linux/UNIX), et enregistrez-le dans le répertoire où se trouve `input.properties`.
   c. Lancez le script.
      - Linux/UNIX : exécutez `./skip_update.sh`.
      - Windows PowerShell : exécutez `skip_update.ps1`.
2. Pour exécuter le script Groovy, utilisez l'écran Groovy dans un processus métier Cloud EPM ou automatisez l'exécution du script via runBusinessRule. Pour plus d'informations sur l'exécution de scripts Groovy à l'aide d'EPM Automate, reportez-vous à la section Exécution de commandes sans installer EPM Automate.

Instructions et script Windows

Créez `input.properties` et `skip_update.ps1` en copiant les scripts de cette section.

1. Créez `input.properties` en copiant le script suivant :

```
username=exampleAdmin
password=examplePassword.epw
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
url=exampleURL
updatemonths=02,05,08,11
```

2. Mettez à jour `input.properties` en indiquant des valeurs de paramètre.

Table 3-12 Paramètres du fichier input.properties

Paramètre	Description
`username`	Nom d'utilisateur d'un administrateur de service.
`password`	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
`url`	URL de l'environnement dans lequel vous souhaitez définir la cadence de mise à jour non mensuelle.
`updatemonths`	Liste des mois, séparés par des virgules, au cours desquels des mises à jour Oracle Fusion Cloud Enterprise Performance Management doivent être appliquées à l'environnement identifié à l'aide du paramètre `url`. Par exemple, `updatemonths=02,05,08,11`. Les mois doivent être indiqués avec deux chiffres : de 01 pour janvier à 12 pour décembre. Veillez à inclure un zéro pour les numéros des mois de janvier à septembre. Le script tente d'exécuter la commande skipUpdate pour les mois non inclus dans la valeur du paramètre `updatemonths`. Par exemple, si vous indiquez `updatemonths=02,05,08,11`, le script tente de définir des indicateurs de non-prise en compte de mise à jour pour janvier, mars, avril, juin, juillet, septembre, octobre et décembre, afin que les mises à jour ne soient appliquées qu'en février, mai, août et novembre.

3. Créez `skip_updates.ps1` en copiant le script suivant :

```
# Skip Update PowerShell script

$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -
raw)
$username="$($inputproperties.username)"
$password="$($inputproperties.password)"
$url="$($inputproperties.url)"
$updatemonths="$($inputproperties.updatemonths)"

$monthsarr = ("01","02","03","04","05","06","07","08","09","10","11","12")
$global:monthsarrfromcurrent = @()
$global:yearsarrfromcurrent = @()
$updatemonthsarr = $updatemonths.Split(",")
$currentyear=Get-Date -Format yy
$currentmonth=Get-Date -Format MM
$nextyear=[int]$currentyear+1

function populateFromCurrentArrays() {
    $startposition = 0

    for ($i = 0; $i -le ($monthsarr.length - 1); $i++) {
    if (${currentmonth} -eq $monthsarr[$i]) {
            $startposition=$i
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
            break
        }
        }

        for ($i = 0; $i -le ($monthsarr.length - 1); $i++) {
        if (${i} -ge ${startposition}) {
                $global:monthsarrfromcurrent += $monthsarr[$i]
                $global:yearsarrfromcurrent += $currentyear
            }
        }

        for ($i = 0; $i -le ($monthsarr.length - 1); $i++) {
        if (${i} -lt ${startposition}) {
                $global:monthsarrfromcurrent += $monthsarr[$i]
                $global:yearsarrfromcurrent += $nextyear
            }
        }
}

function skipUpdateAdd($yearnumber, $monthnumber) {
    echo "Running: epmautomate.bat skipUpdate add version=${yearnumber}.$
{monthnumber} comment=`"adding skipUpdate`""
    epmautomate skipUpdate add version=${yearnumber}.${monthnumber}
comment="adding skipUpdate"
}

function processSkipUpdates() {
    $addcount = 0

    echo "Running: epmautomate.bat login ${username} ${password} ${url}"
    epmautomate login ${username} ${password} ${url}
    echo "Running: epmautomate.bat skipUpdate remove"
    epmautomate skipUpdate remove

    for ($i = 0; $i -le ($global:monthsarrfromcurrent.length - 1); $i++) {
        $match = 1

        if (${addcount} -eq 2) {
            echo "Two skip update add calls have been made. No more will
be attempted."
            break
    }

        for ($j = 0; $j -le ($updatemonthsarr.length - 1); $j++) {
        if ($global:monthsarrfromcurrent[$i] -eq $updatemonthsarr[$j]) {
                $match = 0
        break
        }
    }

        if (${match} -eq 1) {

skipUpdateAdd $global:yearsarrfromcurrent[$i] $global:monthsarrfromcurrent[
$i]
            $addcount += 1
    }
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    }

    echo "Running: epmautomate.bat skipUpdate list"
    epmautomate skipUpdate list
    echo "Running: epmautomate.bat logout"
    epmautomate logout
}

function compareUpdateMonths($thismonth, $nextmonth) {
    $nextmonthorig=${nextmonth}

    if (${nextmonth} -lt ${thismonth}) {
        $nextmonth+=12
    }

    $monthdiff = $nextmonth - $thismonth

    if (${monthdiff} -gt 4) {
        echo "There are more than 3 months skipped from month ${thismonth}
to month ${nextmonthorig}. Please correct updatemonths in input.properties
so that there are not more than three months skipped between each update
month. Exiting."
        exit 1
    }
}

function validateUpdateMonths() {
    for ($i = 0; $i -le ($updatemonthsarr.length - 1); $i++) {
        $nextint = $i + 1
    $thisupdatemonth = $updatemonthsarr[$i]
        $thisupdatemonthint=[int]$thisupdatemonth
    $nextupdatemonth=$updatemonthsarr[$nextint]
        $nextupdatemonthint=[int]$nextupdatemonth

        if (${nextupdatemonth} -eq "") {
            $nextupdatemonth=$updatemonthsarr[0]
            $nextupdatemonthint=[int]$nextupdatemonth
    }

        compareUpdateMonths $thisupdatemonthint $nextupdatemonthint
    }
}

validateUpdateMonths
populateFromCurrentArrays
processSkipUpdates
```

Instructions et script UNIX/Linux

Créez `input.properties` et `skip_update.sh` en copiant les scripts de cette section.

1. Créez `input.properties` en copiant le script suivant :

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
username=exampleAdmin
password=examplePassword.epw
url=exampleURL
updatemonths=02,05,08,11
```

2. Mettez à jour `input.properties` en indiquant des valeurs de paramètre.

Table 3-13 Paramètres du fichier input.properties

Paramètre	Description
`javahome`	Emplacement de `JAVA_HOME`.
`epmautomatescript`	Chemin absolu de l'exécutable EPM Automate (`epmautomate.sh`).
`username`	Nom d'utilisateur d'un administrateur de service.
`password`	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
`url`	URL de l'environnement dans lequel vous souhaitez définir la cadence de mise à jour non mensuelle.
`updatemonths`	Liste des mois, séparés par des virgules, au cours desquels des mises à jour Oracle Fusion Cloud Enterprise Performance Management doivent être appliquées à l'environnement identifié à l'aide du paramètre `url`. Par exemple, `updatemonths=02,05,08,11`. Les mois doivent être indiqués avec deux chiffres. Ajoutez un zéro pour les mois de janvier à septembre. Le script tente d'exécuter la commande skipUpdate pour les mois non inclus dans la valeur du paramètre `updatemonths`. Par exemple, si vous indiquez `updatemonths=02,05,08,11`, le script tente de définir des indicateurs de non-prise en compte de mise à jour pour janvier, mars, avril, juin, juillet, septembre, octobre et décembre, afin que les mises à jour ne soient appliquées qu'en février, mai, août et novembre.

3. Créez `skip_updates.sh` en copiant le script suivant :

```
#!/bin/sh

. ./input.properties
export JAVA_HOME=${javahome}

declare -a monthsarr=(01 02 03 04 05 06 07 08 09 10 11 12)
declare -a monthsarrfromcurrent
declare -a yearsarrfromcurrent
updatemonthsarr=( $(echo "${updatemonths}" | sed 's/,/ /g') )
currentyear=$(date +%y)
nextyear=$((currentyear+1))
currentmonth=$(date +%m)

populateFromCurrentArrays() {
    for i in ${!monthsarr[@]}
    do
        if [[ "${currentmonth}" == "${monthsarr[$i]}" ]]
        then
            startposition=$i
            break
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        fi
    done

    for i in ${!monthsarr[@]}
    do
        if [[ ${i} -ge ${startposition} ]]
        then
            monthsarrfromcurrent=("${monthsarrfromcurrent[@]}" "$
{monthsarr[$i]}")
            yearsarrfromcurrent=("${yearsarrfromcurrent[@]}" "$
{currentyear}")
        fi
    done

    for i in ${!monthsarr[@]}
    do
        if [[ ${i} -lt ${startposition} ]]
        then
            monthsarrfromcurrent=("${monthsarrfromcurrent[@]}" "$
{monthsarr[$i]}")
            yearsarrfromcurrent=("${yearsarrfromcurrent[@]}" "${nextyear}")
        fi
    done
}

skipUpdateAdd() {
    local yearnumber="$1"
    local monthnumber="$2"

    echo "Running: ${epmautomatescript} skipUpdate add version=$
{yearnumber}.${monthnumber} comment=\"adding skipUpdate\""
    ${epmautomatescript} skipUpdate add version=${yearnumber}.$
{monthnumber} comment="adding skipUpdate"
}

processSkipUpdates() {
    local addcount=0

    echo "Running: ${epmautomatescript} login ${username} ${password} $
{url}"
    ${epmautomatescript} login ${username} ${password} ${url}
    echo "Running: ${epmautomatescript} skipUpdate remove"
    ${epmautomatescript} skipUpdate remove

    for i in ${!monthsarrfromcurrent[@]}
    do
        local match=1

        if [[ ${addcount} -eq 2 ]]
        then
            echo "Two skip update add calls have been made. No more will
be attempted."
            break
        fi

        for j in ${!updatemonthsarr[@]}
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        do
            if [[ "${monthsarrfromcurrent[$i]}" == "$
{updatemonthsarr[$j]}" ]]
            then
                match=0
                break
            fi
        done

        if [[ ${match} -eq 1 ]]
        then
            skipUpdateAdd ${yearsarrfromcurrent[$i]} "$
{monthsarrfromcurrent[$i]}"
            addcount=$((addcount+1))
        fi
    done

    echo "Running: ${epmautomatescript} skipUpdate list"
    ${epmautomatescript} skipUpdate list
    echo "Running: ${epmautomatescript} logout"
    ${epmautomatescript} logout
}

compareUpdateMonths() {
    local thismonth=$1
    local nextmonth=$2
    local nextmonthorig=${nextmonth}

    if [[ ${nextmonth} -lt ${thismonth} ]]
    then
        nextmonth=$((nextmonth+12))
    fi

    monthdiff=$((nextmonth-thismonth))

    if [[ ${monthdiff} -gt 4 ]]
    then
        echo "There are more than 3 months skipped from month ${thismonth}
to month ${nextmonthorig}. Please correct updatemonths in input.properties
so that there are not more than three months skipped between each update
month. Exiting."
        exit 1
    fi
}

validateUpdateMonths() {
    for i in ${!updatemonthsarr[@]}
    do
        nextint=$((i+1))
        thisupdatemonth="${updatemonthsarr[$i]}"
        thisupdatemonthint=${thisupdatemonth#0}
        nextupdatemonth="${updatemonthsarr[$nextint]}"
        nextupdatemonthint=${nextupdatemonth#0}

        if [[ ${nextupdatemonth} == "" ]]
        then
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
            nextupdatemonth="${updatemonthsarr[0]}"
            nextupdatemonthint=${nextupdatemonth#0}
        fi

        compareUpdateMonths ${thisupdatemonthint} ${nextupdatemonthint}
    done
}

validateUpdateMonths
populateFromCurrentArrays
processSkipUpdates
```

Script Groovy

Si les mots de passe contiennent des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux. Veillez également à remplacer ces valeurs de paramètre en fonction de vos environnements :

Table 3-14 Paramètres à modifier

Paramètre	Description
user	Nom d'utilisateur d'un administrateur de service.
password	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
url	URL de l'environnement dans lequel vous souhaitez définir la cadence de mise à jour non mensuelle.
updatemonths	Liste des mois, séparés par des virgules, au cours desquels des mises à jour Oracle Fusion Cloud Enterprise Performance Management doivent être appliquées à l'environnement identifié à l'aide du paramètre `url`. Par exemple, `updatemonths=02,05,08,11`. Les mois doivent être indiqués avec deux chiffres : de 01 pour janvier à 12 pour décembre. Veillez à inclure un zéro pour les numéros des mois de janvier à septembre. Le script tente d'exécuter la commande skipUpdate pour les mois non inclus dans la valeur du paramètre `updatemonths`. Par exemple, si vous indiquez `updatemonths=02,05,08,11`, le script tente de définir des indicateurs de non-prise en compte de mise à jour pour janvier, mars, avril, juin, juillet, septembre, octobre et décembre, afin que les mises à jour ne soient appliquées qu'en février, mai, août et novembre.

```
String user = 'service_administrator'
String password = 'examplePWD'
String url = 'example_EPM_URL'
String updatemonths = '02,05,08,11'

def currentdate = new Date()
def yf = new SimpleDateFormat("yy")
def mf = new SimpleDateFormat("MM")
String[] monthsarr = ["01", "02", "03", "04", "05", "06", "07", "08", "09",
"10", "11", "12"]
List<String> monthsarrfromcurrent = new ArrayList<>()
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
List<String> yearsarrfromcurrent = new ArrayList<>()
String currentyear = yf.format(currentdate)
String nextyear = (currentyear.toInteger() + 1).toString()
String currentmonth = mf.format(currentdate)

String[] updateMonthsStringArr = updatemonths.split(',');
def updatemonthsarr = new int[updateMonthsStringArr.length];
for(int i = 0; i < updateMonthsStringArr.length; i++)
{
    updatemonthsarr[i] = Integer.parseInt(updateMonthsStringArr[i]);
}

def LogMessage(String message) {
    def date = new Date()
    def sdf = new SimpleDateFormat("MM/dd/yyyy HH:mm:ss")
    println('[' + sdf.format(date) + '][GROOVY] ' + message);
}

def LogOperationStatus(EpmAutomateStatus opstatus) {
    def returncode = opstatus.getStatus()
    if (returncode != 0){
        LogMessage(opstatus.getOutput())
    }
    LogMessage('return code: ' + returncode)
}

int CompareUpdateMonths(int thismonth, int nextmonth) {
    int nextmonthorig = nextmonth

    if (nextmonth < thismonth) {
        nextmonth = nextmonth + 12
    }

    int monthdiff = nextmonth - thismonth

    if (monthdiff > 4) {
        LogMessage('There are more than three months skipped from month ' +
thismonth + ' to month ' + nextmonthorig + '. Please correct updatemonths so
that there are not more than three months skipped between each update month.
Exiting.')
        return 1
    }

    return 0
}

int ValidateUpdateMonths(int[] updatemonthsarr) {
    for(int i = 0; i < updatemonthsarr.length; i++)
    {
        int nextint = i + 1
        String nextupdatemonth = ""
        int nextupdatemonthint = 0
        String thisupdatemonth = updatemonthsarr[i]
        int thisupdatemonthint = thisupdatemonth.toInteger()

        if (nextint < updatemonthsarr.length) {
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
            nextupdatemonth = updatemonthsarr[nextint]
        } else {
            nextupdatemonth = updatemonthsarr[0]
        }

        nextupdatemonthint = nextupdatemonth.toInteger()

        int returncode = CompareUpdateMonths(thisupdatemonthint,
nextupdatemonthint)
        if (returncode > 0) {
            return 1
        }
    }
    return 0
}

def SkipUpdateAdd(EpmAutomate automate, String yearnumber, String
monthnumber) {
    String yeardotmonth = yearnumber + '.' + monthnumber
    LogMessage('Running: epmautomate skipUpdate add version=' + yeardotmonth
+ ' comment=\"adding skipUpdate\"')
    EpmAutomateStatus status = automate.execute('skipupdate','add','version='
+ yeardotmonth,'comment=\"adding skipUpdate\"')
    LogOperationStatus(status)
}

LogMessage('Starting skip update processing')
EpmAutomate automate = getEpmAutomate()

// validate update months
int returncode = ValidateUpdateMonths(updatemonthsarr)
if (returncode != 0) {
    return 1
}

// populate arrays
int startposition = 0
for(int i = 0; i < monthsarr.length; i++)
{
    if (currentmonth == monthsarr[i]) {
        startposition = i
        break
    }
}

for(int i = 0; i < monthsarr.length; i++)
{
    if (i >= startposition) {
        monthsarrfromcurrent.add(monthsarr[i])
        yearsarrfromcurrent.add(currentyear)
    }
}

for(int i = 0; i < monthsarr.length; i++)
{
    if (i <= startposition) {
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        monthsarrfromcurrent.add(monthsarr[i])
        yearsarrfromcurrent.add(nextyear)
    }
}

// process skip updates
LogMessage("Operation: encrypt " + password + " oracleKey password.epw")
EpmAutomateStatus status =
automate.execute('encrypt',password,"oracleKey","password.epw")
LogOperationStatus(status)

LogMessage("Operation: login " + user + " password.epw " + url)
status = automate.execute('login',user,"password.epw",url)
LogOperationStatus(status)

LogMessage('Running: epmautomate skipUpdate remove')
status = automate.execute('skipupdate','remove')
LogOperationStatus(status)

int addcount = 0

for (int i = 0; i < monthsarrfromcurrent.size(); i++) {
    int match = 1

    if (addcount == 2){
        LogMessage('Two skip update add calls have been made. No more will be
attempted.')
        break
    }

    for(int j = 0; j < updatemonthsarr.length; j++) {

        if (Integer.parseInt(monthsarrfromcurrent.get(i)) ==
updatemonthsarr[j]) {
            match = 0
            break
        }
    }

    if (match == 1) {
        SkipUpdateAdd(automate, yearsarrfromcurrent.get(i),
monthsarrfromcurrent.get(i))
        addcount+=1
    }
}

LogMessage('Running: epmautomate skipUpdate list')
status = automate.execute('skipupdate','list')
LogOperationStatus(status)

LogMessage('Running: epmautomate logout')
status = automate.execute('logout')
LogOperationStatus(status)

LogMessage('Skip update processing completed')
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Création d'une cadence de mise à niveau Cloud EPM trimestrielle avec des cycles de test de six semaines

Utilisez le script de cette section pour créer une solution libre-service permettant d'ignorer les mises à jour de sorte que les environnements Oracle Fusion Cloud Enterprise Performance Management soient mis à jour chaque trimestre avec un cycle de test de six semaines. Dans ce cas, les environnements de production sont mis à jour six semaines après les environnements de test.

Par défaut, Cloud EPM applique une mise à jour mensuelle à vos environnements. Vous utilisez la commande skipUpdate pour ignorer l'application de mises à jour mensuelles à un environnement ou pour consulter les demandes de non-prise en compte de mise à jour en cours. Vous pouvez automatiser l'exécution manuelle des commandes `skipUpdate` à l'aide des scripts inclus dans cette section. Ces scripts automatisent le processus de non-prise en compte de mise à jour de sorte que les mises à jour soient appliquées chaque trimestre avec un cycle de test de six semaines.

Note

1. Vous ne pouvez pas ignorer les mises à jour pendant plus de trois mois consécutifs. Ce script génère une erreur si vous essayez de mettre à jour un environnement Cloud EPM uniquement en février et octobre.
2. Toutes les mises à jour effectuées lors de la période d'intervention seront appliquées à votre environnement lors de la prochaine mise à jour. Par exemple, supposons que vous utilisiez ce script pour planifier des mises à jour trimestrielles qui n'auront lieu que pour les mises à jour de février, mai, août et novembre. Dans ce cas, la mise à jour de mai, par exemple, appliquera à votre environnement l'ensemble des mises à jour et des patches Cloud EPM mensuels applicables publiés après la mise à jour de février. Le processus de maintenance peut prendre plus de temps que d'habitude une fois la mise à jour appliquée.
3. Ce script configure la cadence de mise à jour pour un trimestre seulement. Exemple de scénario : le cycle de mise à jour de l'environnement de test est établi comme suit : premiers vendredis de février (mise à jour 24.02), de mai (mise à jour 24.05), d'août (mise à jour 24.08) et de novembre (mise à jour 24.11). L'environnement de production est mis à jour le troisième vendredi de mars (mise à jour 24.02) avec la version utilisée pour mettre à jour l'environnement de test le premier vendredi de février (mise à jour 24.02). Une mise à jour similaire de l'environnement de production est effectuée la troisième semaine de juin (mise à jour 24.05), de septembre (mise à jour 24.08) et de décembre (mise à jour 24.11). Dans ce scénario, l'environnement de production n'est pas mis à jour vers la mise à jour en cours mais vers celle actuellement exécutée dans l'environnement de test.

Exemple de script Windows

Créez `skip_update.ps1` en copiant le script suivant. Stockez-le dans un répertoire local. Reportez-vous à la section Exécution du script pour plus d'informations sur l'exécution du script :

```
# Skip Update PowerShell script
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -raw)
$username="$($inputproperties.username)"
$password="$($inputproperties.password)"
$url="$($inputproperties.url)"
$updateversions="$($inputproperties.updateversions)"
$podtype="$($inputproperties.podtype)"
$proxyserverusername="$($inputproperties.proxyserverusername)"
$proxyserverpassword="$($inputproperties.proxyserverpassword)"
$proxyserverdomain="$($inputproperties.proxyserverdomain)"

echo "Starting skip_update.ps1 script."

$monthsarr = ("01","02","03","04","05","06","07","08","09","10","11","12")
$global:monthsarrfromcurrent = @()
$global:yearsarrfromcurrent = @()
$updateversionsarr = $updateversions.Split(",")
$currentyear=Get-Date -Format yy
$currentmonth=Get-Date -Format MM
$nextyear=[int]$currentyear+1

function populateFromCurrentArrays() {
    $startposition = 0

    for ($i = 0; $i -le ($monthsarr.length - 1); $i++) {
    if (${currentmonth} -eq $monthsarr[$i]) {
            if (${podtype} -eq "prod") {
                if (${updateversionsarr} -contains ${currentmonth}) {
                    $startposition=$i-2
        } else {
                    $startposition=$i-1
                }
                } else {
                if (${updateversionsarr} -contains ${currentmonth}) {
                    $startposition=$i
        } else {
                    $startposition=$i-1
        }
                }
        break
    }
    }

    if (${startposition} -lt 0) {
        $startposition=$startposition+12
    }

    for ($i = 0; $i -le ($monthsarr.length - 1); $i++) {
    if (${i} -ge ${startposition}) {
            $global:monthsarrfromcurrent += $monthsarr[$i]
            $global:yearsarrfromcurrent += $currentyear
        }
    }

    for ($i = 0; $i -le ($monthsarr.length - 1); $i++) {
    if (${i} -lt ${startposition}) {
            $global:monthsarrfromcurrent += $monthsarr[$i]
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
            $global:yearsarrfromcurrent += $nextyear
        }
    }
}

function skipUpdateAdd($yearnumber, $monthnumber) {
    echo "Running: epmautomate.bat skipUpdate add version=${yearnumber}.$
{monthnumber} comment=`"adding skipUpdate`""
    epmautomate skipUpdate add version=${yearnumber}.${monthnumber}
comment="adding skipUpdate"
}

function processSkipUpdates() {
    $addcount = 0
    $countlimit = 0

    if (${podtype} -eq "prod") {
        $countlimit = 3
    } else {
        $countlimit = 2
    }

    if ((${proxyserverusername} -eq "") -And (${proxyserverpassword} -eq "") -
And (${proxyserverdomain} -eq "")) {
        echo "Running: epmautomate.bat login ${username} ${password} ${url}"
        epmautomate login ${username} ${password} ${url}
    } else {
        echo "Running: epmautomate.bat login ${username} ${password} ${url}
ProxyServerUserName=${proxyserverusername} ProxyServerPassword=$
{proxyserverpassword} ProxyServerDomain=${proxyserverdomain}"
        epmautomate login ${username} ${password} ${url} ProxyServerUserName=$
{proxyserverusername} ProxyServerPassword=${proxyserverpassword}
ProxyServerDomain=${proxyserverdomain}
    }

    echo "Running: epmautomate.bat skipUpdate remove"
    epmautomate skipUpdate remove

    for ($i = 0; $i -le ($global:monthsarrfromcurrent.length - 1); $i++) {
        $match = 1

        if (${addcount} -eq ${countlimit}) {
            echo "Update calls are completed. No more will be attempted."
            break
    }

        for ($j = 0; $j -le ($updateversionsarr.length - 1); $j++) {
        if ((${currentmonth} -eq $updateversionsarr[$j]) -And (${addcount} -
gt 0)) {
                $match = 1
        break
        }

        if (($global:monthsarrfromcurrent[$i] -eq $updateversionsarr[$j]) -
And (${addcount} -eq 0)){
                $match = 0
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        break
        }
    }

        if (${match} -eq 1) {

skipUpdateAdd $global:yearsarrfromcurrent[$i] $global:monthsarrfromcurrent[$i]
            $addcount += 1
    }
    }

    echo "Running: epmautomate.bat skipUpdate list"
    epmautomate skipUpdate list
    echo "Running: epmautomate.bat logout"
    epmautomate logout
}

function compareUpdateMonths($thismonth, $nextmonth) {
    $nextmonthorig=${nextmonth}

    if (${nextmonth} -lt ${thismonth}) {
        $nextmonth+=12
    }

    $monthdiff = $nextmonth - $thismonth

    if (${monthdiff} -gt 4) {
        echo "There are more than 3 versions skipped from version $
{thismonth} to version ${nextmonthorig}. Please correct updateversions in
input.properties so that there are not more than three versions skipped
between each update version. Exiting."
        exit 1
    }
}

function validateUpdateVersions() {
    for ($i = 0; $i -le ($updateversionsarr.length - 1); $i++) {
        $nextint = $i + 1
    $thisupdatemonth = $updateversionsarr[$i]
        $thisupdatemonthint=[int]$thisupdatemonth
    $nextupdatemonth=$updateversionsarr[$nextint]
        $nextupdatemonthint=[int]$nextupdatemonth

        if (${nextupdatemonth} -eq "") {
            $nextupdatemonth=$updateversionsarr[0]
            $nextupdatemonthint=[int]$nextupdatemonth
    }

        compareUpdateMonths $thisupdatemonthint $nextupdatemonthint
    }
}

validateUpdateVersions
populateFromCurrentArrays
processSkipUpdates
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exemple de script Linux/UNIX

Créez `skip_update.sh` en copiant le script suivant. Stockez-le dans un répertoire local. Reportez-vous à la section Exécution du script pour plus d'informations sur l'exécution du script :

```
#!/bin/sh

. ./input.properties

echo "Starting skip_update.sh script."

export JAVA_HOME=${javahome}

updateversionsarr=( $(echo "${updateversions}" | sed 's/,/ /g') )
declare -a updatemonthsarr

year=$(date +%Y)
month=$(date +%m)
day=$(date +%d)
yearcurrent="${year:2:4}"
yearnext=$((yearcurrent+1))
yearprevious=$((yearcurrent-1))
dayofyear=""

populateUpdateMonthsArray() {
    if [[ "${podtype}" == "test" ]]
    then
        updatemonthsarr=("${updateversionsarr[@]}")
    else
        for i in ${updateversionsarr[@]}
        do
            version=$((10#$i))
            if [[ ${version} == "12" ]]
            then
                nextversion="01"
            else
                nextversion=$((version+1))
            fi
            printf -v nextversion "%02d" $nextversion
            updatemonthsarr+=(${nextversion})
        done
    fi
}

getVersionPushDay() {
    local podtype="$1"
    local month="$2"
    local count=0

    for s in `seq 1 28`
    do
        fr=$(date +%a --date="${year}-${month}-$s" )
        if [ "$fr" == "Fri" ];then
            friday[$count]=$s
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
            count=$((count+1))
        fi
    done

    if [[ "${podtype}" == "prod" ]]
    then
        echo ${friday[2]}
    else
        echo ${friday[0]}
    fi
}

getVersionPushDayOfYear() {
    local year="$1"
    local month="$2"
    local day="$3"
    local pushday=""

    pushday=$(date -d "${year}-${month}-${day}" +%j)
    pushday=$((10#$pushday))

    echo "${pushday}"
}

skipUpdateAdd() {
    local version="$1"

    echo "Running: epmautomate skipUpdate add version=${version}
comment=\"adding skipUpdate\""
    ${epmautomatescript} skipUpdate add version=${version} comment="adding
skipUpdate"
}

getVersionInstalled() {
    local version=""
    local response=$(curl -s -u "${username}:${password}" -H 'Content-Type:
application/json' "${url}/interop/rest/v2/maintenance/
getdailymaintenancestarttime")
    version=$(echo "${response}" | jq '.items[0].buildVersion' | cut -d'"' -
f2 | rev | cut -d'.' -f2- | rev)

    if [[ ${version} == "" ]]
    then
        echo "Unable to determine installed EPM Cloud version. Exiting."
        exit 1
    fi

    echo "${version}"
}

getVersionNext() {
    local version="$1"
    local versionyear=$(echo "${version}" | cut -d '.' -f1)
    local versionmonth=$(echo "${version}" | cut -d '.' -f2)
    local versionnext=""
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    if [[ "${versionmonth}" == "12" ]]
    then
        versionyearnext=$(($versionyear+1))
        versionnext="${versionyearnext}.01"
    else
        versionmonthnext=$((10#$versionmonth+1))
        versionmonthnext=$(printf '%02d\n' "$versionmonthnext")
        versionnext="${versionyear}.${versionmonthnext}"
    fi

    echo "${versionnext}"
}

processSkipUpdates() {
    local day=$1
    local versionpushday=$2
    local version=""
    local daybeforeversionpushday=$((versionpushday-1))

    if [[ "${proxyserverusername}" == "" ]] && [[ "${proxyserverpassword}" ==
"" ]] && [[ "${proxyserverdomain}" == "" ]]
    then
        echo "Running: ${epmautomatescript} login ${username} ${password} $
{url}"
        ${epmautomatescript} login ${username} ${password} ${url}
    else
        echo "Running: ${epmautomatescript} login ${username} ${password} $
{url} ProxyServerUserName=${proxyserverusername} ProxyServerPassword=$
{proxyserverpassword} ProxyServerDomain=${proxyserverdomain}"
        ${epmautomatescript} login ${username} ${password} ${url}
ProxyServerUserName=${proxyserverusername} ProxyServerPassword=$
{proxyserverpassword} ProxyServerDomain=${proxyserverdomain}
    fi

    echo "Running: epmautomate skipUpdate remove"
    ${epmautomatescript} skipUpdate remove

    if [[ "${podtype}" == "test" ]]
    then
        if [[ ${updateversionsarr[@]} =~ ${month} ]] && [[ ${day} -eq $
{versionpushday} ]]
        then
            return
        else
            version=$(getVersionInstalled)
            echo "Installed version: $version"
            versionnext=$(getVersionNext "${version}")
            versionafternext=$(getVersionNext "${versionnext}")
            skipUpdateAdd "${versionnext}"
            skipUpdateAdd "${versionafternext}"
        fi
    else # prod
        if [[ ${updatemonthsarr[@]} =~ ${month} ]] && [[ ${day} -eq $
{daybeforeversionpushday} ]]
        then
            return
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        else
            version=$(getVersionInstalled)
            echo "Installed version: $version"
            versionnext=$(getVersionNext "${version}")
            versionafternext=$(getVersionNext "${versionnext}")
            versionafternextnext=$(getVersionNext "${versionafternext}")
            skipUpdateAdd "${versionnext}"
            skipUpdateAdd "${versionafternext}"
            skipUpdateAdd "${versionafternextnext}"
        fi
    fi
    echo "Running: ${epmautomatescript} skipUpdate list"
    ${epmautomatescript} skipUpdate list
    echo "Running: ${epmautomatescript} logout"
    ${epmautomatescript} logout
}

compareUpdateMonths() {
    local thismonth=$1
    local nextmonth=$2
    local nextmonthorig=${nextmonth}

    if [[ ${nextmonth} -lt ${thismonth} ]]
    then
        nextmonth=$((nextmonth+12))
    fi

    monthdiff=$((nextmonth-thismonth))

    if [[ ${monthdiff} -gt 4 ]]
    then
        echo "There are more than 3 versions skipped from version $
{thismonth} to version ${nextmonthorig}. Please correct updateversions in
input.properties so that there are not more than three versions skipped
between each update version. Exiting."
        exit 1
    fi
}

validateUpdateVersions() {
    for i in ${!updateversionsarr[@]}
    do
        nextint=$((i+1))
        thisupdatemonth="${updateversionsarr[$i]}"
        thisupdatemonthint=${thisupdatemonth#0}
        nextupdatemonth="${updateversionsarr[$nextint]}"
        nextupdatemonthint=${nextupdatemonth#0}

        if [[ ${nextupdatemonth} == "" ]]
        then
            nextupdatemonth="${updateversionsarr[0]}"
            nextupdatemonthint=${nextupdatemonth#0}
        fi

        compareUpdateMonths ${thisupdatemonthint} ${nextupdatemonthint}
    done
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
}

validateUpdateVersions
populateUpdateMonthsArray
versionpushday=$(getVersionPushDay ${podtype} ${month})
versionpushdayofyear=$(getVersionPushDayOfYear ${year} ${month} $
{versionpushday})
dayofyear=$(date -d "${year}-${month}-${day}" +%j)
dayofyear=$((10#$dayofyear))
processSkipUpdates "${dayofyear}" "${versionpushdayofyear}"
```

Script Groovy

Créez le script Groovy `skip_update.groovy` en copiant le script suivant et en le mettant à jour. Reportez-vous à la section Exécution du script pour plus d'informations sur l'exécution du script :
Mettez à jour les variables suivantes du script Groovy :

- `username` : nom d'utilisateur d'un administrateur de service sur l'environnement dans lequel définir la cadence de mise à jour non mensuelle.
- `password` : mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
- `url` : URL de l'environnement dans lequel vous souhaitez définir la cadence de mise à jour non mensuelle.
- `updateversions` : liste des mises à jour Cloud EPM, séparées par des virgules, qui doivent être appliquées à l'environnement identifié à l'aide du paramètre `url`. Par exemple, `updateversions=02,05,08,11`.
  Les versions doivent être indiquées avec deux chiffres. Ajoutez un zéro pour les mises à jour 01 à 09. Le script tente d'exécuter la commande skipUpdate pour les mises à jour non incluses dans la valeur du paramètre `updateversions`. Par exemple, si vous indiquez `updateversions=02,05,08,11`, le script tente de définir des indicateurs de non-prise en compte des mises à jour 01 (janvier), 03 (mars), 04 (avril), 06 (juin), 07 (juillet), 09 (septembre), 10 (octobre) et 12 (décembre). Dans ce cas, les mises à jour Cloud EPM 02 (février), 05 (mai), 08 (août) et 11 (novembre) sont appliquées à l'environnement.
- `podtype` : type d'environnement Cloud EPM. Les valeurs valides sont `test` et `prod`.
- `proxyserverusername` : nom d'utilisateur qui permet d'authentifier une session sécurisée auprès du serveur proxy qui contrôle l'accès à Internet.
- `proxyserverpassword` : mot de passe qui permet d'authentifier l'utilisateur auprès du serveur proxy.
- `proxyserverdomain` : nom du domaine défini pour le serveur proxy.

> **Note**
>
> Si vous n'utilisez pas de serveur proxy, n'indiquez pas de valeur pour les paramètres `proxyserverusername`, `proxyserverpassword` et `proxyserverdomain`.

```
String username = 'service_administrator'
String password = 'examplePWD'
String url = 'example_EPM_URL'
String updateversions = '01,04,07,10'
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
String podtype = 'test'
String proxyserverusername = ''
String proxyserverpassword = ''
String proxyserverdomain = ''

def currentdate = new Date()
def yf = new SimpleDateFormat("yy")
def mf = new SimpleDateFormat("MM")
String[] monthsarr = ["01", "02", "03", "04", "05", "06", "07", "08", "09",
"10", "11", "12"]
List<String> monthsarrfromcurrent = new ArrayList<>()
List<String> yearsarrfromcurrent = new ArrayList<>()
String currentyear = yf.format(currentdate)
String nextyear = (currentyear.toInteger() + 1).toString()
String currentmonth = mf.format(currentdate)

String[] updateVersionsStringArr = updateversions.split(',');
def updateversionsarr = new int[updateVersionsStringArr.length];
for(int i = 0; i < updateVersionsStringArr.length; i++)
{
    updateversionsarr[i] = Integer.parseInt(updateVersionsStringArr[i]);
}

def LogMessage(String message) {
    def date = new Date()
    def sdf = new SimpleDateFormat("MM/dd/yyyy HH:mm:ss")
    println('[' + sdf.format(date) + '][GROOVY] ' + message);
}

def LogOperationStatus(EpmAutomateStatus opstatus) {
    def returncode = opstatus.getStatus()
    LogMessage(opstatus.getOutput())
    LogMessage('return code: ' + returncode)
}

int CompareUpdateMonths(int thismonth, int nextmonth) {
    int nextmonthorig = nextmonth

    if (nextmonth < thismonth) {
        nextmonth = nextmonth + 12
    }

    int monthdiff = nextmonth - thismonth

    if (monthdiff > 4) {
        LogMessage('There are more than 3 versions skipped from version ' +
thismonth + ' to version ' + nextmonthorig + '. Please correct updateversions
so that there are not more than three versions skipped between each update
version. Exiting.')
        return 1
    }

    return 0
}

int ValidateUpdateMonths(int[] updateversionsarr) {
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    for(int i = 0; i < updateversionsarr.length; i++)
    {
        int nextint = i + 1
        String nextupdatemonth = ""
        int nextupdatemonthint = 0
        String thisupdatemonth = updateversionsarr[i]
        int thisupdatemonthint = thisupdatemonth.toInteger()

        if (nextint < updateversionsarr.length) {
            nextupdatemonth = updateversionsarr[nextint]
        } else {
            nextupdatemonth = updateversionsarr[0]
        }

        nextupdatemonthint = nextupdatemonth.toInteger()

        int returncode = CompareUpdateMonths(thisupdatemonthint,
nextupdatemonthint)
        if (returncode > 0) {
            return 1
        }
    }
    return 0
}

def SkipUpdateAdd(EpmAutomate automate, String yearnumber, String
monthnumber) {
    String yeardotmonth = yearnumber + '.' + monthnumber
    LogMessage('Running: epmautomate skipUpdate add version=' + yeardotmonth
+ ' comment=\"adding skipUpdate\"')
    EpmAutomateStatus status = automate.execute('skipupdate','add','version='
+ yeardotmonth,'comment=\"adding skipUpdate\"')
    LogOperationStatus(status)
}

LogMessage('Starting skip update processing')
EpmAutomate automate = getEpmAutomate()

// validate update months
int returncode = ValidateUpdateMonths(updateversionsarr)
if (returncode != 0) {
    return 1
}

// populate arrays
int startposition = 0
for(int i = 0; i < monthsarr.length; i++)
{
    if (currentmonth == monthsarr[i]) {
        if (podtype.equals("prod")) {
            if (updateVersionsStringArr.contains(currentmonth)) {
                startposition = (i-2)
            } else {
                startposition = (i-1)
            }
        } else {
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
            if (updateVersionsStringArr.contains(currentmonth)) {
                startposition = i
            } else {
                startposition = (i-1)
            }
        }

        break
    }
}

if (startposition < 0) {
    startposition = startposition + 12
}

for(int i = 0; i < monthsarr.length; i++)
{
    if (i >= startposition) {
        monthsarrfromcurrent.add(monthsarr[i])
        yearsarrfromcurrent.add(currentyear)
    }
}

for(int i = 0; i < monthsarr.length; i++)
{
    if (i <= startposition) {
        monthsarrfromcurrent.add(monthsarr[i])
        yearsarrfromcurrent.add(nextyear)
    }
}

// process skip updates
LogMessage("Operation: encrypt " + password + " oracleKey password.epw")
EpmAutomateStatus status =
automate.execute('encrypt',password,"oracleKey","password.epw")
LogOperationStatus(status)

if ((proxyserverusername != null && proxyserverusername != '') &&
(proxyserverpassword != null && proxyserverpassword != '') &&
(proxyserverdomain != null && proxyserverdomain != '')) {
    LogMessage("Operation: login " + username + " password.epw " + url + "
ProxyServerUserName=" + proxyserverusername + " ProxyServerPassword=" +
proxyserverpassword + " ProxyServerDomain=" + proxyserverdomain)
    status =
automate.execute('login',username,"password.epw",url,"ProxyServerUserName=" +
proxyserverusername,"ProxyServerPassword=" +
proxyserverpassword,"ProxyServerDomain=" + proxyserverdomain)
    LogOperationStatus(status)
} else {
    LogMessage("Operation: login " + username + " password.epw " + url)
    status = automate.execute('login',username,"password.epw",url)
    LogOperationStatus(status)
}
LogMessage('Running: epmautomate skipUpdate remove')
status = automate.execute('skipupdate','remove')
LogOperationStatus(status)
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
int addcount = 0
int countlimit = 0

if (podtype.equals("prod")) {
    countlimit = 3
} else {
    countlimit = 2
}

for (int i = 0; i < monthsarrfromcurrent.size(); i++) {
    int match = 1

    if (addcount == countlimit){
        LogMessage('Update add calls are completed. No more will be
attempted.')
        break
    }

    for(int j = 0; j < updateversionsarr.length; j++) {

        if ((Integer.parseInt(currentmonth) == updateversionsarr[j]) &&
(addcount > 0)) {
            match = 1
            break
        }

        if ((Integer.parseInt(monthsarrfromcurrent.get(i)) ==
updateversionsarr[j]) && (addcount == 0)) {
            match = 0
            break
        }
    }

    if (match == 1) {
        SkipUpdateAdd(automate, yearsarrfromcurrent.get(i),
monthsarrfromcurrent.get(i))
        addcount+=1
    }
}

LogMessage('Running: epmautomate skipUpdate list')
status = automate.execute('skipupdate','list')
LogOperationStatus(status)
println(status.getItemsList())

LogMessage('Running: epmautomate logout')
status = automate.execute('logout')
LogOperationStatus(status)

LogMessage('Skip update processing completed')
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Création du fichier input.properties pour exécuter des scripts skip_update Windows et Linux/UNIX

Pour exécuter skip_update.ps1 ou skip_update.sh, créez le fichier input.properties et mettez-le à jour avec les informations de votre environnement. Enregistrez le fichier dans un répertoire local. Le contenu de ce fichier varie en fonction du système d'exploitation.
Windows

```
username=exampleAdmin
password=examplePassword.epw
url=exampleURL
updateversions=01,04,07,10
podtype=test
```

Linux/UNIX

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
username=exampleAdmin
password=examplePassword.epw
url=exampleURL
updateversions=01,04,07,10
podtype=test
```

Table 3-15 Paramètres du fichier input.properties

Paramètre	Description
javahome	Emplacement de JAVA_HOME. Pour Linux/UNIX uniquement.
epmautomatescript	Chemin absolu de l'exécutable EPM Automate (epmautomate.sh). Pour Linux/UNIX uniquement.
username	Nom d'utilisateur d'un administrateur de service.
password	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
url	URL de l'environnement dans lequel vous souhaitez définir la cadence de mise à jour non mensuelle.
updateversions	updateversions : liste des mises à jour Cloud EPM, séparées par des virgules, qui doivent être appliquées à l'environnement identifié à l'aide du paramètre url. Par exemple, updateversions=02,05,08,11. Les versions doivent être indiquées avec deux chiffres. Ajoutez un zéro pour les mises à jour 01 à 09. Le script tente d'exécuter la commande skipUpdate pour les mises à jour non incluses dans la valeur du paramètre updateversions. Par exemple, si vous indiquez updateversions=02,05,08,11, le script tente de définir des indicateurs de non-prise en compte des mises à jour 01 (janvier), 03 (mars), 04 (avril), 06 (juin), 07 (juillet), 09 (septembre), 10 (octobre) et 12 (décembre). Dans ce cas, les mises à jour Cloud EPM 02 (février), 05 (mai), 08 (août) et 11 (novembre) sont appliquées à l'environnement.
podtype	Type d'environnement Cloud EPM. Les valeurs valides sont test et prod.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exécution du script

1. **Pour Windows et Linux/UNIX uniquement :**
   - Créez `skip_update.ps1` ou `skip_update.sh` en copiant le script d'une section précédente.
   - Créez le fichier `input.properties` et enregistrez-le dans le répertoire où se trouve le script `skip_update`. Le contenu de ce fichier varie en fonction du système d'exploitation. Reportez-vous à la section Création du fichier input.properties pour exécuter des scripts skip_update Windows et Linux/UNIX.
     Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter le script.
   - Lancez le script.
     - **Windows PowerShell :** exécutez `skip_update.ps1`.
     - **Linux/UNIX :** exécutez `./skip_update.sh`.
2. **Groovy :**
   - Créez le script Groovy `skip_update.groovy` et mettez-le à jour comme nécessaire.
   - Utilisez l'écran Groovy dans un processus métier Cloud EPM ou automatisez l'exécution du script via runBusinessRule. Pour plus d'informations sur l'exécution de scripts Groovy à l'aide d'EPM Automate, reportez-vous à la section Exécution de commandes sans installer EPM Automate.

Exemples de scénarios pour Planning, Consolidation, Tax Reporting et Enterprise Profitability and Cost Management

Les scripts disponibles dans cette section vous aident à automatiser des tâches dans les environnements Planning (y compris les modules Planning), Financial Consolidation and Close, Tax Reporting et Enterprise Profitability and Cost Management.

Voir aussi :

- Automatisation de l'export d'un grand nombre de cellules à partir d'un cube en mode "aggregate storage"
  Utilisez le script PowerShell ou Bash de cette section pour exporter un grand nombre de cellules à partir d'un cube en mode "aggregate storage" (ASO).
- Import de métadonnées dans une application
  Utilisez ces scripts pour importer manuellement les métadonnées d'application d'un fichier.
- Import de données, exécution d'un script de calcul et copie des données d'une base de données en mode "block storage" vers une base de données en mode "aggregate storage"
  Utilisez ces scripts afin d'importer les données d'un fichier, d'actualiser le cube, d'exécuter une règle métier pour calculer le cube et de transmettre les données vers un cube ASO.
- Export et téléchargement de données et de métadonnées
  Utilisez ces scripts pour exporter les données et métadonnées d'application, puis télécharger les fichiers exportés dans un répertoire local.
- Export et téléchargement de données d'application
  Utilisez ces scripts pour exporter les données d'application et les télécharger dans un répertoire local.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Automatisation de l'archivage des enregistrements d'audit d'application
  Utilisez les scripts Windows et Linux de cette section pour automatiser le processus d'export et d'archivage des données d'audit d'application vers un ordinateur local.
- Téléchargement d'un fichier de données dans un environnement et exécution d'une règle de chargement de données
  Utilisez ces scripts pour charger un fichier dans un environnement et exécuter une règle de données afin d'importer les données du fichier dans une application.
- Automatisation de l'intégration quotidienne des données
  Ce scénario porte sur l'utilisation d'un exemple de script pour automatiser l'intégration régulière des données.

Automatisation de l'export d'un grand nombre de cellules à partir d'un cube en mode "aggregate storage"

Utilisez le script PowerShell ou Bash de cette section pour exporter un grand nombre de cellules à partir d'un cube en mode "aggregate storage" (ASO).

Les limites imposées par le paramètre `QUERYRESULTLIMIT` d'Oracle Essbase rendent impossible l'export de grandes quantités de données à partir de l'interface utilisateur. Le script PowerShell disponible dans cette section fractionne l'opération d'export en un nombre défini de jobs, exécute chaque job, télécharge les données exportées, puis concatène les fichiers d'export en un seul fichier pour garantir qu'un seul en-tête est utilisé.

> **Remarque**
>
> Ces scripts exécutent un job existant de type Exporter les données. Pour obtenir des instructions détaillées sur la création de jobs, reportez-vous à la section "Gestion des jobs" du guide *Administration de Planning*.

Script PowerShell

```
$user = '<USERNAME>'
$pass = '<PASSWORD>'
$serverURL = '<URL>'
$applicationName = '<APPLICATIONNAME>'
$cubeName = '<CUBENAME>'
$splitDimension = '<DIMENSION_TO_SPLIT_THE_EXPORT>'
$topLevelMemberForExport = '<TOP_MEMBER_FOR_EXPORT>'
$exportJobName = '<EXPORT_JOB_NAME>'
$exportFilePrefix = '<PREFIX_FOR_EXPORT_FILE>'
$columnMembers = '<MEMBERS_ON_COLUMNS>'
$povMembers = '<POV_MEMBERS>'
$numberOfExportFiles = <NUMBER_OF_FILES_TO_SPLIT_THE_EXPORT>

$memberArray = @()
$exportFileArray = @()

function getLevel0 ($parent) {
    $parent.children.ForEach({
        if ( $_.children.count -eq 0 ) {
            $script:memberArray += $_.name
        }
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        getLevel0($_)
    })
}

function findMember ($tree, $memberName) {
    $subtree = ""
    if ($tree.name -eq $memberName){
        return $tree
    } else {
        $tree.children.ForEach({
            #Write-Host $_.name
            if ($subtree -eq ""){ $subtree = findMember $_ $memberName}
        })
        return $subtree
    }
}

#putting together base64 encoded authentication header based un user and
password
$encodedCredentials =
[Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($($user) +
":" + $($pass)))
$headers = @{ Authorization = "Basic $encodedCredentials" }

#test login
$testRequest = $serverURL + '/HyperionPlanning/rest/v3/applications'

try {
    $response = Invoke-RestMethod -Uri $testRequest -Method Get -
Headers $headers -UseBasicParsing
}
catch {
    Write-Host $_
    return
}

#retrieve dimension hierarchy from application
Write-Host "Retrieving member list for split dimension " $splitDimension
$request = $serverURL + '/HyperionPlanning/rest/v3/internal/applications/'
+ $applicationName + '/plantypes/' + $cubeName + '/dimensions/'
+ $splitDimension
try {
    $response = Invoke-RestMethod -Uri $request -Method Get -Headers $headers
-UseBasicParsing
}
catch {
    Write-Host $_
    return
}
Write-Host $splitDimension " member list retrieved"

#search for the top of the export hierarchy
Write-Host "Searching for member " $topLevelMemberForExport " in hierarchy"
$member = findMember $response $topLevelMemberForExport
if ( $member.name -ne $topLevelMemberForExport ) {
    Write-Host $topLevelMemberForExport " not found in hierarchy, exiting ..."
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    return 128
}
Write-Host "Found member " $topLevelMemberForExport " in hierarchy"

#retrieve level 0 memebers in export hierarchy
Write-Host "Retrieving Level 0 members for hierarchy"
getLevel0($member)
if ( $memberArray.Length -eq 0 ) {
    Write-Host "no level 0 members found in hierarchy, exiting ..."
    return 128
}
Write-Host $memberArray.Length " Level 0 members for export hierarchy
retrieved"


$request = $serverURL + '/HyperionPlanning/rest/v3/applications/'
+ $applicationName + '/jobs'

#splitting member list into the number of export files
$numberOfEntitiesPerFile =
[math]::truncate($memberArray.Length / $numberOfExportFiles)
for ($i = 1; $i -le $numberOfExportFiles; $i++) {
    $memberList = ""
    $firstMember = ($i - 1) * $numberOfEntitiesPerFile
    if ($i -lt $numberOfExportFiles) {
        $lastMember = $i * $numberOfEntitiesPerFile
    } else {
        $lastMember = $i * $numberOfEntitiesPerFile + $memberArray.Length
% $numberOfExportFiles
    }
    for ($j = $firstMember; $j -lt $lastMember; $j++) {
        $memberList += $memberArray[$j]
        if ($j -lt $lastMember - 1) {$memberList += ","} #avoid adding a
comma (,) after the last member of each set
    }

    $jobDetails='
    {
    "jobType":"EXPORT_DATA","jobName":"' + $exportJobName + '",
    "parameters":{
                  "exportFileName":"Export-' + $i + '.zip",
                  "rowMembers":"' +  $memberList + '",
                  "columnMembers":"' + $columnMembers + '",
                "povMembers":"' + $povMembers + '"
                 }
    }'

    #start export job
    try{
        $response = Invoke-RestMethod -Uri $request -Method Post -
Headers $headers -Body $jobDetails -ContentType "application/json"}
    catch {
        Write-Host $_
        return
    }
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    Write-Host "Started export job " $i " out of " $numberOfExportFiles

    #checking job status, continue once jos is completed
    $statusRequest = $serverURL + '/HyperionPlanning/rest/v3/applications/'
+ $applicationName + '/jobs/' + $response.jobId
    $statusResponse = Invoke-RestMethod -Uri $statusRequest -Method Get -
Headers $headers -UseBasicParsing

    while ( $statusResponse.descriptiveStatus -eq "Processing" ) {
        Write-Host $statusResponse.descriptiveStatus
        Start-Sleep -s 10
        $statusResponse = Invoke-RestMethod -Uri $statusRequest -Method Get -
Headers $headers -UseBasicParsing
    }
    Write-Host $statusResponse.descriptiveStatus

    Write-Host "Downloading export file ..."
    $downloadRequest = $serverURL + '/interop/rest/11.1.2.3.600/
applicationsnapshots/Export-' + $i + '.zip/contents'
    $statusResponse = Invoke-RestMethod -Uri $downloadRequest -Method Get -
Headers $headers -OutFile "$exportFilePrefix-$i.zip"

    Write-Host "Expanding archive ..."
    Expand-Archive -Force -LiteralPath "$exportFilePrefix-$i.zip" -
DestinationPath "$exportFilePrefix-$i"
    Remove-Item "$exportFilePrefix-$i.zip"

    Get-ChildItem -Path "$exportFilePrefix-$i" -File -Name | ForEach-Object
{ $exportFileArray += "$exportFilePrefix-$i\" +  $_ }
}

Write-Host "creating outputfile ..."
#write header to outputfile
Get-Content $exportFileArray[0] | Select-Object -First 1 | Out-File
"$exportFilePrefix.csv"

#write content to outputfile skipping header
ForEach ($exportFile in $exportFileArray) {
    Get-Content $exportFile | Select-Object -Skip 1 | Out-File -Append
"$exportFilePrefix.csv"
}

Compress-Archive -LiteralPath "$exportFilePrefix.csv"  -DestinationPath
"$exportFilePrefix.zip"

Write-Host "cleaning up ..."
Remove-Item "$exportFilePrefix-*" -Recurse
Remove-Item "$exportFilePrefix.csv"
```

Script Bash

```
#!/bin/bash

user='<USERNAME>'
pass='<PASSWORD>'
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
serverURL='<URL>'
applicationName='<APPLICATIONNAME>'
cubeName='<CUBENAME>'
splitDimension='<DIMENSION_TO_SPLIT_THE_EXPORT>'
topLevelMemberForExport='<TOP_MEMBER_FOR_EXPORT>'
exportJobName='<EXPORT_JOB_NAME>'
exportFilePrefix='<PREFIX_FOR_EXPORT_FILE>'
columnMembers='<MEMBERS_ON_COLUMNS>'
povMembers='<POV_MEMBERS>'
numberOfExportFiles=<NUMBER_OF_FILES_TO_SPLIT_THE_EXPORT>

getRowMembers() {
    local memberList="$1"
    local firstMember=$2
    local lastMember=$3
    local nameCount=0
    local rowMember=""
    local rowMembers=""

    while IFS= read -r line
    do
        if [[ "${line}" == *"name"* ]]
        then
            if [[ ${nameCount} -ge ${firstMember} ]] && [[ ${nameCount} -lt $
{lastMember} ]]
            then
              rowMember=$(echo "${line}" | cut -d':' -f2- | sed s'/[",]//g')
              rowMembers="${rowMembers}${rowMember},"
            fi
            ((nameCount+=1))
        fi
    done <<< "${memberList}"
    rowMembers=$(echo "${rowMembers}" | rev | cut -d',' -f2- | rev)
    echo "${rowMembers}"
}

getLevel0()
{
    local memberList="$1"
    local names=$(echo "${memberList}" | jq 'recurse (try .children[])
| .name' | sed -e 's/"//g')
    local elements=""

    formerIFS=$IFS
    IFS=$'\n'
    namesarr=($names)
    IFS=$formerIFS

    for i in ${!namesarr[@]}
    do
        testelement=$(echo "${memberList}" | jq --arg currentName "$
{namesarr[i]}" 'recurse (try .children[]) | select(.name==$currentName)')
        if [[ "${testelement}" != *"children"* ]]
        then
            elements="${elements}${testelement}"
        fi
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    done

    echo "${elements}"
}

#test login
header="Content-Type: application/x-www-form-urlencoded"
applicationsRequest="${serverURL}/HyperionPlanning/rest/v3/applications"
response=$(curl -X "GET" -s -w "%{http_code}" -u "${user}:${pass}" -H "$
{header}" "${applicationsRequest}")
http_response_code=$(echo "${response}" | rev | cut -d'}' -f1 | rev)

if [ ${http_response_code} -ne 200 ]
then
    echo "${response}"
    exit
fi

#retrieve dimension hierarchy from application
echo "Retrieving member list for split dimension ${splitDimension}"
splitDimensionRequest="${serverURL}/HyperionPlanning/rest/v3/internal/
applications/${applicationName}/plantypes/${cubeName}/dimensions/$
{splitDimension}"
response=$(curl -X GET -s -w "%{http_code}" -u "${user}:${pass}" -o "response-
memberlist.txt" -D "respHeader-memberlist.txt" -H "${header}" "$
{splitDimensionRequest}")
http_response_code=$(echo "${response}" | rev | cut -d'}' -f1 | rev)

if [ ${http_response_code} -ne 200 ]
then
    echo "${response}"
    exit
fi

echo "${splitDimension} member list retrieved"

#search for the top of the export hierarchy
echo "Searching for member ${topLevelMemberForExport} in hierarchy"
memberList=$(cat response-memberlist.txt | jq --arg topLevelMember "$
{topLevelMemberForExport}" 'recurse(try .children[]) | select (.name
== $topLevelMember)')
if [[ "${memberList}" == "" ]]
then
    echo "${topLevelMemberForExport} not found in hierarchy, exiting ..."
    exit 128
fi

echo "Found member ${topLevelMemberForExport} in hierarchy"

#retrieve level 0 members in export hierarchy
echo "Retrieving Level 0 members for hierarchy"
totalCount=$(echo "${memberList}" | grep "name" | wc -l)
grepChildrenCount=$(echo "${memberList}" | grep "children" | wc -l)
levelZeroCount=$((totalCount-grepChildrenCount))

if [[ "${levelZeroCount}" -eq 0 ]]
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
then
    echo "no level 0 members found in hierarchy, exiting ..."
    exit 128
fi

echo "${levelZeroCount} Level 0 members for export hierarchy retrieved"

#splitting member list into the number of export files
numberOfEntitiesPerFile=$((levelZeroCount/numberOfExportFiles))
jobsRequest="${serverURL}/HyperionPlanning/rest/v3/applications/$
{applicationName}/jobs"
header="Content-Type: application/json"

for ((i = 1 ; i <= ${numberOfExportFiles}; i++))
do
    firstMember=$((($i-1)*numberOfEntitiesPerFile))
    if [[ ${i} -lt ${numberOfExportFiles} ]]
    then
        lastMember=$((i*numberOfEntitiesPerFile))
    else
        lastMember=$
((i*numberOfEntitiesPerFile+levelZeroCount%numberOfExportFiles))
    fi

    elements=$(getLevel0 "${memberList}")
    rowMembers=$(getRowMembers "${elements}" ${firstMember} ${lastMember})

    response=$(curl -X POST -s -w "%{http_code}" -u "${user}:${pass}" -o
"response-job.txt" -D "respHeader-job.txt" -H "${header}" "${jobsRequest}" -d
'{"jobType":"EXPORT_DATA","jobName":"'"${exportJobName}"'","parameters":
{"exportFileName":"Export-'"${i}"'.zip","rowMembers":"'"$
{rowMembers}"'","columnMembers":"'"${columnMembers}"'","povMembers":"'"$
{povMembers}"'"}}')

    echo "Started export job " $i " out of " $numberOfExportFiles
    jobId=$(cat response-job.txt | grep -o '"jobId":[^, }]*' | cut -d':' -f2)
    descriptiveStatus=$(cat response-job.txt | grep -o '"descriptiveStatus":
[^, }]*' | cut -d':' -f2 | sed -e 's/"//g')
    jobIdRequest="${serverURL}/HyperionPlanning/rest/v3/applications/$
{applicationName}/jobs/${jobId}"
    response=$(curl -X GET -s -w "%{http_code}" -u "${user}:${pass}" -o
"response-jobstatus.txt" -D "respHeader-jobstatus.txt" -H "${header}" "$
{jobIdRequest}")

    jobId=$(cat response-jobstatus.txt | grep -o '"jobId":[^, }]*' | cut -
d':' -f2)
    descriptiveStatus=$(cat response-jobstatus.txt | grep -o
'"descriptiveStatus":[^, }]*' | cut -d':' -f2 | sed -e 's/"//g')

    while [[ "${descriptiveStatus}" == "Processing" ]]
    do
        echo "${descriptiveStatus}"
        sleep 10
        response=$(curl -X GET -s -w "%{http_code}" -u "${user}:${pass}" -o
"response-jobstatus.txt" -D "respHeader-jobstatus.txt" -H "${header}" "$
{jobIdRequest}")
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        descriptiveStatus=$(cat response-jobstatus.txt | grep -o
'"descriptiveStatus":[^, }]*' | cut -d':' -f2 | sed -e 's/"//g')
    done

    echo "${descriptiveStatus}"

    echo "Downloading export file ..."
    contentsRequest="${serverURL}/interop/rest/11.1.2.3.600/
applicationsnapshots/Export-${i}.zip/contents"
    curl -X GET -s -w "%{http_code}" -u "${user}:${pass}" -D "respHeader-
download.txt" "${contentsRequest}" > "${exportFilePrefix}-${i}.zip"

    echo "Expanding archive ..."
    unzip "${exportFilePrefix}-${i}.zip" -d "${exportFilePrefix}-${i}"
    rm "${exportFilePrefix}-${i}.zip"

    echo "Writing to outputfile ..."
    if [[ -d "${exportFilePrefix}-${i}" ]]
    then
        find "${exportFilePrefix}-${i}" -name \*.csv | xargs cat | tail -n +2
>> "${exportFilePrefix}.csv"
    fi
done
zip "${exportFilePrefix}.zip" "${exportFilePrefix}.csv"

echo "cleaning up ..."
find . -name "${exportFilePrefix}-*" | xargs rm -r
rm "${exportFilePrefix}.csv"
```

Pour exporter un grand nombre de cellules à partir d'un cube ASO (en mode "agregate storage"), procédez comme suit :

1. Copiez le script Bash ou PowerShell, et enregistrez-le dans votre système de fichiers, par exemple sous la forme `ASOCellExport.ps1` ou `ASOCellExport.sh`.
2. Modifiez le fichier de script et définissez les valeurs de paramètre. Pour plus de détails, reportez-vous au tableau suivant.

Tableau 3-16 Valeurs de variable à inclure dans les scripts PowerShell et Bash

Variable	Description
`user`	Domaine et nom d'utilisateur d'un administrateur de service au format *`DOMAIN.USER`*. **Exemples :** **Windows :** `$user = 'exampleDomain.jDoe'` **Linux/UNIX :** `user = 'exampleDomain.jDoe'`
`pass`	Mot de passe de l'administrateur de service ou emplacement du fichier de mots de passe crypté. Pour obtenir des informations sur la création d'un fichier de mots de passe cryptés, reportez-vous à la commande encrypt. **Exemples :** **Windows :** `$pass = 'Example'` **Linux/UNIX :** `pass = 'Example'`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 3-16 (suite) Valeurs de variable à inclure dans les scripts PowerShell et Bash

Variable	Description
`serverURL`	URL de l'environnement Oracle Fusion Cloud Enterprise Performance Management. **Exemples :** **Windows :** `$serverURL = 'https://example .oraclecloud.com'` **Linux/UNIX :** `serverURL = 'https://example.oraclecloud.com'`
`applicationName`	Nom d'une application Planning, Financial Consolidation and Close, Tax Reporting ou Enterprise Profitability and Cost Management. **Exemples :** **Windows :** `$applicationName = 'Vision'` **Linux/UNIX :** `applicationName = 'Vision'`
`cubeName`	Nom d'un cube dans l'application. **Exemples :** **Windows :** `$cubeName = 'VisASO'` **Linux/UNIX :** `cubeName = 'VisASO'`
`splitDimension`	Nom d'une dimension dont les membres sont utilisés pour fractionner l'export en groupes. **Exemples :** **Windows :** `$splitDimension = 'Account'` **Linux/UNIX :** `splitDimension = 'Account'`
`topLevelMemberForExport`	Nom d'un membre de la sous-hiérarchie de dimension dans laquelle la liste des membres de niveau 0 est créée. **Exemples :** **Windows :** `$topLevelMemberForExport = 'Total Cash Flow'` **Linux/UNIX :** `topLevelMemberForExport = 'Total Cash Flow'`
`exportJobName`	Nom d'un job existant de type Export de données. Les paramètres indiqués dans ce job seront remplacés par ceux définis dans le script. **Exemples :** **Windows :** `$exportJobName = 'ASO Cell Export'` **Linux/UNIX :** `exportJobName = 'ASO Cell Export'`
`exportFilePrefix`	Préfixe de nom de fichier utilisé pour identifier les fichiers générés par le job d'export. **Exemples :** **Windows :** `$exportFilePrefix = 'cashflow'` **Linux/UNIX :** `exportFilePrefix = 'cashflow'`
`columnMembers`	Colonnes de membres à inclure dans l'export. **Exemples :** **Windows :** `$columnMembers = 'Period'` **Linux/UNIX :** `columnMembers = 'Period'`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 3-16 (suite) Valeurs de variable à inclure dans les scripts PowerShell et Bash

Variable	Description
`povMembers`	Points de vue à inclure dans l'export. Les membres de PDV doivent inclure toutes les autres dimensions et peuvent comprendre les fonctions indiquées ci-dessous : `ILvl0Descendants(YearTotal), ILvl0Descendants(Year), ILvl0Descendants(Scenario), ILvl0Descendants(Version), ILvl0Descendants(P_TP), ILvl0Descendants(AltYear)` **Exemples :** **Windows :** `$povMembers = 'YTD'` **Linux/UNIX :** `povMembers = 'YTD'`
`numberOfExportFiles`	Nombre de jobs à exécuter pour cette opération d'export. Si l'export échoue en raison de la limite de requête, augmentez le nombre. **Exemples :** **Windows :** `$numberOfExportFiles = 3` **Linux/UNIX :** `numberOfExportFiles = 3`

3. A l'aide du planificateur Windows ou d'un job Cron, planifiez l'exécution du script à un moment opportun. Reportez-vous à la section Automatisation de l'exécution de scripts pour obtenir les étapes détaillées.

Import de métadonnées dans une application

Utilisez ces scripts pour importer manuellement les métadonnées d'application d'un fichier.

Les scripts effectuent les activités suivantes :

- Connexion à un environnement.
- Chargement d'un fichier de métadonnées.
- Import des métadonnées du fichier chargé vers l'application à l'aide d'un job.
- Actualisation du cube.
- Déconnexion.

Exemple de script Windows

Créez `importMetadata.ps1` en copiant le script suivant. Stockez-le dans un répertoire local.

```
$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -raw)
$username="$($inputproperties.username)"
$passwordfile="$($inputproperties.passwordfile)"
$serviceURL="$($inputproperties.serviceURL)"
$file1="$($inputproperties.file1)"
$jobName="$($inputproperties.jobName)"

epmautomate login ${username} ${passwordfile} ${serviceURL}
epmautomate uploadfile ${file1}
epmautomate importmetadata ${jobName} ${file1}
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
epmautomate refreshcube
epmautomate logout
```

Exemple de script Linux/UNIX

Créez `importMetadata.sh` en copiant le script suivant. Stockez-le dans un répertoire local.

```
#!/bin/bash
. ./input.properties
export JAVA_HOME=${javahome}
${epmautomatescript} login "${username}" "${passwordfile}" "${serviceURL}"
${epmautomatescript} uploadfile "${file1}"
${epmautomatescript} importmetadata "${jobName}" "${file1}"
${epmautomatescript} refreshcube
${epmautomatescript} logout
```

Création du fichier input.properties

Créez le fichier `input.properties` en copiant l'un des exemples suivants et en le mettant à jour avec les informations de votre environnement. Enregistrez le fichier dans le répertoire dans lequel `importMetadata.ps1` ou `importMetadata.sh` est stocké.

Windows

```
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
File1=FILE_NAME.zip
jobName=JOB_NAME
```

Linux/UNIX

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
File1=FILE_NAME.zip
jobName=JOB_NAME
```

Tableau 3-17 Paramètres du fichier input.properties

Paramètre	Description
`javahome`	Emplacement de `JAVA_HOME`. Pour Linux/UNIX uniquement.
`epmautomatescript`	Chemin absolu de l'exécutable EPM Automate (`epmautomate.sh`). Pour Linux/UNIX uniquement.
`username`	Nom d'utilisateur d'un administrateur de service disposant également du rôle Administrateur de domaine d'identité.
`password`	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
`serviceURL`	URL de l'environnement à partir duquel générer l'instantané.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 3-17 (suite) Paramètres du fichier input.properties

Paramètre	Description
`File1`	Nom du fichier ZIP contenant les métadonnées à importer.
`JobName`	Job à utiliser pour l'import des métadonnées.

Exécution des scripts

1. Créez `importMetadata.ps1` ou `importMetadata.sh` en copiant le script d'une section précédente.
2. Créez le fichier `input.properties` et enregistrez-le dans le répertoire où se trouve le script `importMetadata`. Le contenu de ce fichier varie en fonction du système d'exploitation. Reportez-vous à la section Création du fichier input.properties.
   Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter le script.
3. Lancez le script.
   - **Windows PowerShell :** exécutez `importMetadata.ps1`.
   - **Linux/UNIX :** exécutez `./importMetadata.sh`.

Import de données, exécution d'un script de calcul et copie des données d'une base de données en mode "block storage" vers une base de données en mode "aggregate storage"

Utilisez ces scripts afin d'importer les données d'un fichier, d'actualiser le cube, d'exécuter une règle métier pour calculer le cube et de transmettre les données vers un cube ASO.

Les scripts effectuent les actions suivantes :

- Connexion à un environnement.
- Chargement d'un fichier `data.csv`.
- Import des données du fichier `data.csv` dans l'application à l'aide du job `loadingq1data`.
- Actualisation du cube.
- Exécution de règles métier pour transformer les données.
- Transmission de données vers une base de données en mode "aggregate storage" à l'aide d'un job.
- Déconnexion.

Exemple de script Windows

Créez `importDataPlus.ps1` en copiant le script suivant. Enregistrez-le dans un répertoire local.

```
$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -raw)
$username="$($inputproperties.username)"
$passwordfile="$($inputproperties.passwordfile)"
$serviceURL="$($inputproperties.serviceURL)"
$importDataJobName="$($inputproperties.importDataJobName)"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
$businessRuleName="$($inputproperties.businessRuleName)"
$planTypeMapName="$($inputproperties.planTypeMapName)"
$param1Key="$($inputproperties.param1Key)"
$param1Value="$($inputproperties.param1Value)"
$param2Key="$($inputproperties.param2Key)"
$param2Value="$($inputproperties.param2Value)"
$clearData="$($inputproperties.clearData)"

epmautomate login ${username} ${passwordfile} ${serviceURL}
epmautomate uploadfile ${file1}
epmautomate importdata ${importDataJobName} ${file1}
epmautomate refreshcube
epmautomate runbusinessrule ${businessRuleName} ${param1Key}=${param1Value} $
{param2Key}=${param2Value}
epmautomate runplantypemap ${planTypeMapName} clearData=${clearData}
epmautomate logout
```

Exemple de script Linux/UNIX

Créez `importDataPlus.ps1` en copiant le script suivant. Enregistrez-le dans un répertoire local.

```
#!/bin/bash
. ./input.properties
export JAVA_HOME=${javahome}
${epmautomatescript} login "${username}" "${passwordfile}" "${serviceURL}"
${epmautomatescript} uploadfile "${file1}"
${epmautomatescript} importdata "${importDataJobName}" "${file1}"
${epmautomatescript} refreshcube
${epmautomatescript} runbusinessrule "${businessRuleName}" "${param1Key}=$
{param1Value}" "${param2Key}=${param2Value}"
${epmautomatescript} runplantypemap "${planTypeMapName}" clearData=$
{clearData}
${epmautomatescript} logout
```

Création du fichier input.properties

Windows

```
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
File1=FILE_NAME.csv
importDataJobName=FILE_NAME
businessRuleName=RULE_NAME
planTypeMapName=PLAN_TYPE_MAP_NAME
param1Key=RUN-TIME PARAMETER_1
param1Value=RUN-TIME PARAMETER_1_VALUE
param2Key=RUN-TIME PARAMETER_2
param2Value=RUN-TIME PARAMETER_2_VALUE
clearData=true
```

Linux/UNIX

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
File1=FILE_NAME.csv
importDataJobName=FILE_NAME
businessRuleName=RULE_NAME
planTypeMapName=PLAN_TYPE_MAP_NAME
param1Key=RUN-TIME PARAMETER_1
param1Value=RUN-TIME PARAMETER_1_VALUE
param2Key=RUN-TIME PARAMETER_2
param2Value=RUN-TIME PARAMETER_2_VALUE
clearData=true
```

Tableau 3-18 Paramètres du fichier input.properties

Paramètre	Description
javahome	Emplacement de JAVA_HOME. Pour Linux/UNIX uniquement.
epmautomatescript	Chemin absolu de l'exécutable EPM Automate (epmautomate.sh). Pour Linux/UNIX uniquement.
username	Nom d'utilisateur d'un administrateur de service disposant également du rôle Administrateur de domaine d'identité.
password	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
serviceURL	URL de l'environnement à partir duquel générer l'instantané.
File1	Fichier d'import à partir duquel charger les données dans l'application.
importDataJobName	Nom du job à utiliser pour l'import des données.
businessRuleName	Règle métier à exécuter sur les données importées.
planTypeMapName	Mappe de type de plan à utiliser pour la copie de données d'une base de données BSO vers une base de données ASO, ou d'une base de données BSO vers une autre base de données BSO.
param1Key	Invite d'exécution 1 pour exécuter la règle métier.
param1Value	Valeur de l'invite d'exécution 1.
param2Key	Invite d'exécution 2 pour exécuter la règle métier.
param2Value	Valeur de l'invite d'exécution 2.
clearData	Indique si les données dans la base de données cible doivent être supprimées. Indiquez false pour conserver les données.

Exécution des scripts

1. Créez importDataPlus.ps1 ou importDataPlus.sh en copiant le script d'une section précédente.
2. Créez le fichier input.properties et enregistrez-le dans le répertoire où se trouve le script importDataPlus. Le contenu de ce fichier varie en fonction du système d'exploitation. Reportez-vous à la section Création du fichier input.properties.
   Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter le script.
3. Lancez le script.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- **Windows PowerShell :** exécutez `importDataPlus.ps1`.
- **Linux/UNIX :** exécutez `./importDataPlus.sh`.

Export et téléchargement de données et de métadonnées

Utilisez ces scripts pour exporter les données et métadonnées d'application, puis télécharger les fichiers exportés dans un répertoire local.

Les scripts effectuent les activités suivantes :

- Connexion à un environnement.
- Export des métadonnées dans un fichier ZIP à l'aide du job indiqué.
- Export des données d'application dans un fichier ZIP à l'aide du job indiqué.
- Spécification du contenu de la boîte de réception/d'envoi.
- Téléchargement des fichiers de données exportés sur l'ordinateur local.
- Déconnexion.

Exemple de script Windows

Créez `exportDownloadMetadataAndData.ps1` en copiant le script suivant. Stockez-le dans un répertoire local.

```
$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -raw)
$username="$($inputproperties.username)"
$passwordfile="$($inputproperties.passwordfile)"
$serviceURL="$($inputproperties.serviceURL)"
$exportFile1="$($inputproperties.exportFile1)"
$exportFile2="$($inputproperties.exportFile2)"
$exportMetaDataJobName="$($inputproperties.exportMetaDataJobName)"
$exportDataJobName="$($inputproperties.exportDataJobName)"

epmautomate login ${username} ${passwordfile} ${serviceURL}
epmautomate exportmetadata ${exportMetaDataJobName} ${exportFile1}
epmautomate exportdata ${exportDataJobName} ${exportFile2}
epmautomate listfiles
epmautomate downloadfile ${exportFile1}
epmautomate downloadfile f${exportFile2}
epmautomate logout
```

Exemple de script Linux/UNIX

Créez `exportDownloadMetadataAndData.sh` en copiant le script suivant. Stockez-le dans un répertoire local.

```
#!/bin/bash
. ./input.properties
export JAVA_HOME=${javahome}
${epmautomatescript} login "${username}" "${passwordfile}" "${serviceURL}"
${epmautomatescript} exportmetadata "${exportMetaDataJobName}" "$
{exportFile1}"
${epmautomatescript} exportdata "${exportDataJobName}" "${exportFile2}"
${epmautomatescript} listfiles
${epmautomatescript} downloadfile "${exportFile1}"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
${epmautomatescript} downloadfile "${exportFile2}"
${epmautomatescript} logout
```

Création du fichier de propriétés

Créez le fichier `input.properties` en copiant l'un des exemples suivants et en le mettant à jour avec les informations de votre environnement. Enregistrez le fichier dans le répertoire dans lequel `exportDownloadMetadataAndData.ps1` ou `exportDownloadMetadataAndData.sh` est stocké.

Windows

```
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
exportFile1=FILE_NAME1.zip
exportFile2=FILE_NAME2.zip
exportMetaDataJobName=METADATA_EXPORT_JOB_NAME
exportDataJobName=DATA_EXPORT_JOB_NAME
```

Linux/UNIX

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
exportFile1=FILE_NAME1.zip
exportFile2=FILE_NAME2.zip
exportMetaDataJobName=METADATA_EXPORT_JOB_NAME
exportDataJobName=DATA_EXPORT_JOB_NAME
```

Tableau 3-19 Paramètres du fichier input.properties

Paramètre	Description
`javahome`	Emplacement de `JAVA_HOME`. Pour Linux/UNIX uniquement.
`epmautomatescript`	Chemin absolu de l'exécutable EPM Automate (`epmautomate.sh`). Pour Linux/UNIX uniquement.
`username`	Nom d'utilisateur d'un administrateur de service disposant également du rôle Administrateur de domaine d'identité.
`password`	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
`serviceURL`	URL de l'environnement à partir duquel générer l'instantané.
`exportFile1`	Nom du fichier vers lequel exporter les métadonnées.
`exportFile2`	Nom du fichier vers lequel exporter les données.
`exportDataJobName1`	Job à utiliser pour l'export des métadonnées.
`exportDataJobName2`	Job à utiliser pour l'export des données.

Exécution des scripts

1. Créez `exportDownloadMetadataAndData.ps1` ou `exportDownloadMetadataAndData.sh` en copiant le script d'une section précédente.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

2. Créez le fichier `input.properties` et enregistrez-le dans le répertoire où se trouve le script `exportDownloadMetadataAndData`. Le contenu de ce fichier varie en fonction du système d'exploitation. Reportez-vous à la section Création du fichier de propriétés. Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter le script.
3. Lancez le script.
   - **Windows PowerShell :** exécutez `exportDownloadMetadataAndData.ps1`.
   - **Linux/UNIX :** exécutez `./exportDownloadMetadataAndData.sh`.

Export et téléchargement de données d'application

Utilisez ces scripts pour exporter les données d'application et les télécharger dans un répertoire local.

Les scripts effectuent les opérations suivantes :

- Connexion à un environnement.
- Sauvegarde de deux ensembles de données à l'aide des jobs indiqués.
- Téléchargement des fichiers de données exportés.
- Déconnexion.

Exemple de script Windows

Créez `exportDownloadData.ps1` en copiant le script suivant. Enregistrez-le dans un répertoire local.

```
$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -raw)
$username="$($inputproperties.username)"
$passwordfile="$($inputproperties.passwordfile)"
$serviceURL="$($inputproperties.serviceURL)"
$exportFile1="$($inputproperties.exportFile1)"
$exportFile2="$($inputproperties.exportFile2)"
$exportDataJobName1="$($inputproperties.exportDataJobName1)"
$exportDataJobName2="$($inputproperties.exportDataJobName2)"

epmautomate login ${username} ${passwordfile} ${serviceURL}
epmautomate exportdata ${exportDataJobName1} ${exportFile1}
epmautomate exportdata ${exportDataJobName2} ${exportFile2}
epmautomate listfiles
epmautomate downloadfile ${exportFile1}
epmautomate downloadfile ${exportFile2}
epmautomate logout
```

Exemple de script Linux/UNIX

Créez `exportDownloadData.sh` en copiant le script suivant. Enregistrez-le dans un répertoire local.

```
#!/bin/bash
. ./input.properties
export JAVA_HOME=${javahome}
${epmautomatescript} login "${username}" "${passwordfile}" "${serviceURL}"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
${epmautomatescript} exportdata "${exportDataJobName1}" "${exportFile1}"
${epmautomatescript} exportdata "${exportDataJobName2}" "${exportFile2}"
${epmautomatescript} listfiles
${epmautomatescript} downloadfile "${exportFile1}"
${epmautomatescript} downloadfile "${exportFile2}"
${epmautomatescript} logout
```

Création du fichier input.properties

Créez le fichier `input.properties` en copiant l'un des exemples suivants et en le mettant à jour avec les informations de votre environnement. Enregistrez le fichier dans le répertoire dans lequel `exportDownloadData.ps1` ou `exportDownloadData.sh` est stocké.

Windows

```
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
exportFile1=FILE_NAME.zip
exportFile2=FILE_NAME.zip
exportDataJobName1=JOB_NAME
exportDataJobName2=FILE_NAME
```

Linux/UNIX

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
exportFile1=FILE_NAME.zip
exportFile2=FILE_NAME.zip
exportDataJobName1=FILE_NAME
exportDataJobName2=FILE_NAME
```

Tableau 3-20 Paramètres du fichier input.properties

Paramètre	Description
`javahome`	Emplacement de `JAVA_HOME`. Pour Linux/UNIX uniquement.
`epmautomatescript`	Chemin absolu de l'exécutable EPM Automate (`epmautomate.sh`). Pour Linux/UNIX uniquement.
`username`	Nom d'utilisateur d'un administrateur de service disposant également du rôle Administrateur de domaine d'identité.
`password`	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
`serviceURL`	URL de l'environnement à partir duquel générer l'instantané.
`exportFile1` et `exportFile2`	Nom du fichier vers lequel exporter les données.
`exportDataJobName1` et `exportDataJobName2`	Job à utiliser pour l'export des données.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exécution des scripts

1. Créez `exportDownloadData.ps1` ou `exportDownloadData.sh` en copiant le script d'une section précédente.
2. Créez le fichier `input.properties` et enregistrez-le dans le répertoire où se trouve le script `exportDownloadData`. Le contenu de ce fichier varie en fonction du système d'exploitation. Reportez-vous à la section Tableau 1.
   Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter le script.
3. Lancez le script.
   - **Windows PowerShell :** exécutez `exportDownloadData.ps1`.
   - **Linux/UNIX :** exécutez `./exportDownloadData.sh`.

Automatisation de l'archivage des enregistrements d'audit d'application

Utilisez les scripts Windows et Linux de cette section pour automatiser le processus d'export et d'archivage des données d'audit d'application vers un ordinateur local.

Les données d'audit d'application sont conservées pendant 365 jours uniquement. Personnalisez ces scripts et exécutez-les tous les 180 jours, ou selon la fréquence requise par vos stratégies de conservation de données, afin d'éviter toute perte de données d'audit historiques datant de plus de 365 jours.

> **Note**
>
> Ces scripts sont adaptés à l'archivage de données dans un stockage local. Vous pouvez les modifier afin qu'ils archivent les fichiers de données d'audit exportés sur un stockage réseau ou un cloud de stockage (Oracle Object Storage par exemple).

Table 3-21 Paramètres et leurs valeurs

Paramètre	Valeur
`url`	URL de l'environnement. **Exemples :** • **Windows :** `set url=https://example-epmidm.epm.usphoenix-1.ocs.oraclecloud.com/epmcloud` • **Linux :** `url=https://example-epmidm.epm.usphoenix-1.ocs.oraclecloud.com/epmcloud`
`user`	Nom d'utilisateur d'un administrateur de service pour la connexion à l'environnement afin de télécharger des données d'audit. **Exemples :** • **Windows :** `set user=ExampleAdmin` • **Linux :** `user=ExampleAdmin`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Table 3-21 (Cont.) Paramètres et leurs valeurs

Paramètre	Valeur
`password`	Mot de passe de l'administrateur de service (non recommandé), ou nom et emplacement du fichier de mots de passe cryptés. Pour plus d'informations sur la création d'un fichier de mot de passe crypté, reportez-vous à la commande encrypt. **Exemples :** • **Windows :** `set password="C:\mySecuredir\password.epw"` • **Linux :** `password="/home/user1/mySecuredir/password.epw"`
`AuditFileName`	Nom du fichier de données d'audit. Le script ajoute l'horodatage de l'export des données d'audit à la fin nom du fichier afin de rendre ce dernier unique. **Exemple :** • **Windows :** `set AuditFileName=AuditData` • **Linux :** `AuditFileName=AuditData`
`NumberOfBackups`	Nombre de fichiers de sauvegarde à conserver dans le stockage. La valeur par défaut est 10. Au-delà, la sauvegarde la plus ancienne est remplacée. **Exemple :** • **Windows :** `set NumberOfBackups=20` • **Linux :** `NumberOfBackups=20`
Pour les scripts Linux uniquement	
`epmautomatescript`	Emplacement dans lequel EPM Automate est installé. **Exemple :** `/home/user1/epmautomate/bin/epmautomate.sh`
`javahome`	Emplacement de *JAVA_HOME*. **Exemple :** `/home/user1/jdk_17.0.12/`

Script Windows

Créez un fichier batch (par exemple, `AuditExport.bat`) contenant un script semblable au script suivant afin d'automatiser l'export des données d'audit et leur téléchargement sur un ordinateur local.

```
@echo off
rem Sample script to download and maintain 10 audit data backups
rem Update the following  parameters

SET url=https://example.oraclecloud.com
SET user=ServiceAdmin
SET password=Example.epw
SET AuditFileName="AuditBackup"
SET NumberOfBackups=10

rem EPM Automate commands
call epmautomate login %user% %password% %url%
     IF %ERRORLEVEL% NEQ 0 goto :ERROR
       call epmautomate exportAppAudit %AuditFileName% nDays=180
     IF %ERRORLEVEL% NEQ 0 goto :ERROR
       call epmautomate downloadfile %AuditFileName%.zip
     IF %ERRORLEVEL% NEQ 0 goto :ERROR
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
       call epmautomate logout
     IF %ERRORLEVEL% NEQ 0 goto :ERROR

rem Rename downloaded audit data backup, keep the last 10 backups
Set Timestamp=%date:~4,2%_%date:~7,2%_%date:~10,2%%
Set Second=%time:~0,2%%time:~3,2%
ren %AuditFileName%.zip %AuditFileName%_%Timestamp%_%Second%.zip
SET Count=0
FOR %%A IN (%AuditFileName%*.*) DO SET /A Count += 1
IF %Count% gtr %NumberOfBackups% FOR %%A IN (%AuditFileName%*.*) DO del "%%A"
&& GOTO EOF
:EOF

echo Scheduled Task Completed successfully
exit /b %errorlevel%
:ERROR
echo Failed with error #%errorlevel%.
exit /b %errorlevel%
```

Script Linux

Créez un script d'interpréteur de commandes (par exemple, `AuditExport.sh`) contenant un script semblable au script suivant afin d'automatiser l'export des données d'audit et leur téléchargement sur un ordinateur local.

```
#!/bin/sh
# Sample script to export, download and maintain 10 audit data backups
# Update the following seven parameters
url=https://example.oraclecloud.com
user=serviceAdmin
password=/home/user1/epmautomate/bin/example.epw
auditfilename="AuditBackup"
numberofbackups=10
epmautomatescript=/home/user1/epmautomate/bin/epmautomate.sh
javahome=/home/user1/jdk_17.0.12/

export JAVA_HOME=${javahome}

printResult()
    {
    op="$1"
    opoutput="$2"
    returncode="$3"

    if [ "${returncode}" -ne 0 ]
    then
        echo "Command failed. Error code: ${returncode}. ${opoutput}"
    else
        echo "${opoutput}"
    fi
}

processCommand()
{
    op="$1"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
    date=`date`

    echo "Running ${epmautomatescript} ${op}"
    operationoutput=`eval "$epmautomatescript $op"`
    printResult "$op" "$operationoutput" "$?"
}
op="login ${user} ${password} ${url}"
processCommand "${op}"
op="exportAppAudit \"${auditfilename}\" -nDays=180"
processCommand "${op}"
op="downloadfile \"${auditfilename}.zip\""
processCommand "${op}"
op="logout"
processCommand "${op}"
# Rename the downloaded audit data backup, keep the last 10 backups
timestamp=`date +%m_%d_%Y_%I%M`
mv "${auditfilename}.zip" "${auditfilename}_${timestamp}.zip"

((numberofbackups+=1))
ls -tp ${auditfilename}*.zip | grep -v '/$' | tail -n +${numberofbackups} |
xargs -d '\n' -r rm --
```

Téléchargement d'un fichier de données dans un environnement et exécution d'une règle de chargement de données

Utilisez ces scripts pour charger un fichier dans un environnement et exécuter une règle de données afin d'importer les données du fichier dans une application.

Prérequis

- Les définitions suivantes dans la gestion des données :
  - une définition de règle de chargement de données nommée `VisionActual` ;. Nous partons du principe que la règle de données n'indique pas de chemin de fichier pour le fichier d'entrée.
  - les définitions des périodes du `Mar-15` au `Jun-15`.
- Un fichier de données correctement formaté (`GLActual.dat`) qui contient des données.

Pour importer des données et exécuter une règle de chargement des données, exécutez les commandes permettant de réaliser les étapes suivantes :

- Vous connecter à l'environnement.
- Télécharger un fichier `GLActual.dat` qui contient des données des périodes du `Mar-15` au `Jun-15` dans le dossier de gestion des données `inbox/Vision`.
- Importer les données du fichier `GLActual.dat` dans la gestion des données à l'aide de la règle de chargement des données `VisionActual`, de la période de début `Mar-15`, de la période de fin `Jun-15` et du mode d'import `REPLACE`.
- Exporter les données avec l'option `STORE_DATA` pour fusionner les données de la table intermédiaire de gestion des données avec les données d'application existantes.
- Vous déconnecter.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Exemple de script Windows

Créez `runDataLoadRule.ps1` en copiant le script suivant. Stockez-le dans un répertoire local.

```
$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -raw)
$username="$($inputproperties.username)"
$passwordfile="$($inputproperties.passwordfile)"
$serviceURL="$($inputproperties.serviceURL)"
$dataFile="$($inputproperties.dataFile)"
$dataRuleName="$($inputproperties.dataRuleName)"
$startPeriod="$($inputproperties.startPeriod)"
$endPeriod="$($inputproperties.endPeriod)"
$importMode="$($inputproperties.importMode)"
$exportMode="$($inputproperties.exportMode)"

epmautomate login ${username} ${passwordfile} ${serviceURL}
epmautomate uploadfile ${datafile} ${dataFileUploadLocation}
epmautomate rundatarule ${dataRuleName} ${startPeriod} ${endPeriod} $
{importMode} ${exportMode} ${dataFileUploadLocation}/${dataFile}
epmautomate logout
```

Exemple de script Linux/UNIX

Créez `runDataLoadRule.sh` en copiant le script suivant. Stockez-le dans un répertoire local.

```
#!/bin/bash
. ./input.properties
export JAVA_HOME=${javahome}
${epmautomatescript} login "${username}" "${passwordfile}" "${serviceURL}"
${epmautomatescript} uploadfile "${datafile}" "${dataFileUploadLocation}"
${epmautomatescript} rundatarule "${dataRuleName}" "${startPeriod}" "$
{endPeriod}" "${importMode}" "${exportMode}" "${dataFileUploadLocation}/$
{dataFile}"
${epmautomatescript} logout
```

Création du fichier input.properties

Créez le fichier `input.properties` en copiant l'un des exemples suivants et en le mettant à jour avec les informations de votre environnement. Enregistrez le fichier dans le répertoire dans lequel `runDataLoadRule.ps1` ou `runDataLoadRule.sh` est stocké.

Windows

```
username=serviceAdmin
passwordfile=./password.epw
serviceURL=https://example.oraclecloud.com
dataFile=GLActual.dat
dataFileUploadLocation=UPLOAD_LOCATION
dataRuleName=RULE_NAME
startPeriod=START_PERIOD
endPeriod=END_PERIOD
importMode=IMPORT_MODE
exportMode=EXPORT_MODE
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Linux/UNIX

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURLdataFile=GLActual.dat
dataFileUploadLocation=UPLOAD_LOCATION
dataRuleName=RULE_NAME
startPeriod=START_PERIOD
endPeriod=END_PERIOD
importMode=IMPORT_MODE
exportMode=EXPORT_MODE
```

Tableau 3-22 Paramètres du fichier input.properties

Paramètre	Description
`javahome`	Emplacement de `JAVA_HOME`. Pour Linux/UNIX uniquement.
`epmautomatescript`	Chemin absolu de l'exécutable EPM Automate (`epmautomate.sh`). Pour Linux/UNIX uniquement.
`username`	Nom d'utilisateur d'un administrateur de service disposant également du rôle Administrateur de domaine d'identité.
`password`	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
`serviceURL`	URL de l'environnement à partir duquel générer l'instantané.
`dataFile`	Fichier qui contient les données à importer à l'aide de la règle de données.
`dataFileUploadLocation`	Emplacement dans lequel charger le fichier de données.
`dataRuleName`	Nom d'une règle de chargement de données définie dans l'intégration des données.
`startPeriod`	Première période pour laquelle les données doivent être chargées. Ce nom de période doit être défini dans le mapping de période de l'intégration des données.
`endPeriod`	Pour un chargement de données multipériode, dernière période pour laquelle des données doivent être chargées. Pour un chargement de période unique, utilisez la même période que la période de début. Ce nom de période doit être défini dans le mapping de période de l'intégration des données.
`importMode`	Mode d'import des données dans l'intégration des données. Utilisez `APPEND`, `REPLACE` ou `RECALCULATE`. Utilisez `NONE` pour ignorer l'import de données dans les tables intermédiaires.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 3-22 (suite) Paramètres du fichier input.properties

Paramètre	Description
exportMode	Mode d'export des données vers l'application. Utilisez l'intégration des données. Utilisez STORE_DATA, ADD_DATA, SUBTRACT_DATA ou REPLACE_DATA. Utilisez NONE pour ignorer l'export de données de l'intégration des données vers l'application. **Remarque** Financial Consolidation and Close prend uniquement en charge les modes MERGE et NONE.

Exécution du script

1. Créez `runDataLoadRule.ps1` ou `runDataLoadRule.sh` en copiant le script d'une section précédente.
2. Créez le fichier `input.properties` et enregistrez-le dans le répertoire où se trouve le script `runDataLoadRule`. Le contenu de ce fichier varie en fonction du système d'exploitation. Reportez-vous à la section Création du fichier input.properties.
   Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter le script.
3. Lancez le script.
   - **Windows PowerShell :** exécutez `runDataLoadRule.ps1`.
   - **Linux/UNIX :** exécutez `./runDataLoadRule.sh`.

Automatisation de l'intégration quotidienne des données

Ce scénario porte sur l'utilisation d'un exemple de script pour automatiser l'intégration régulière des données.

Créez un fichier de lot (`.bat`) ou d'interpréteur de commandes (`.sh`) qui contient un script semblable aux scripts suivants pour automatiser les activités relatives à l'intégration des données. Le script d'exemple suivant pour Windows automatise l'intégration quotidienne des données d'application en réalisant ces activités :

- Connectez-vous à un environnement.
- Supprimez `DailyPlanData` s'il est présent.
- Téléchargez `DailyPlanData` dans le service.
- Exécutez la règle métier `Clear Plan Targets` sur le type de plan `Plan1`.
- Importez les données à l'aide du nom de job `LoadDailyPlan`.
- Exécutez la règle métier `Balance Sheet - Plan`.
- Exécutez la règle métier `Allocate Plan Targets`.
- Supprimez `DailyTarget.zip` s'il est présent.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Exportez les données dans `DailyTarget.zip` à l'aide du nom de job `ExportDailyTarget`.
- Téléchargez `DailyTarget.zip` vers votre serveur et ajoutez l'horodatage
- Déconnectez-vous de l'environnement.

> **Remarque**
>
> Si vous réadaptez ce script à votre utilisation personnelle, veillez à modifier les valeurs des paramètres `SET url` et `SET user`. En outre, vous pouvez modifier les valeurs des paramètres `dataimportfilename`, `dataexportfilename`, `importdatajobname`, `exportdatajobname`, `br_clear`, `br_calculatebalancesheet` et `br_allocatetarget` afin qu'ils répondent à vos besoins.

Reportez-vous à la section Automatisation de l'exécution de scripts pour plus d'informations sur la planification du script à l'aide du planificateur de tâches Windows.

```
@echo off

rem Sample Script to demonstrate daily data integration with
rem Cloud EPM application.
rem This script uploads Plan data, clears target numbers,
rem runs a business rule to calculate balance sheet data, and
rem recalculates target numbers on the Vision demo application

rem Please update these parameters
SET url=https://example.oraclecloud.com
SET user=serviceAdmin
SET dataimportfilename=DailyPlanData.csv
SET dataexportfilename=DailyTarget
SET importdatajobname=LoadDailyPlan
SET exportdatajobname=ExportDailyTarget
SET br_clear=Clear Plan Targets
SET br_calculatebalancesheet=Balance Sheet - Plan
SET br_allocatetarget=Allocate Plan Targets

SET password=%1

rem Executing EPM Automate commands

CD /D %~dp0
call epmautomate login %user% %password% %url%
IF %ERRORLEVEL% NEQ 0 goto :ERROR

for /f %%i in ('call epmautomate listfiles') do if %%i==%dataimportfilename%
(call epmautomate deletefile %%i)
IF %ERRORLEVEL% NEQ 0 goto :ERROR

call epmautomate uploadfile %dataimportfilename%
IF %ERRORLEVEL% NEQ 0 goto :ERROR

call epmautomate runbusinessrule "%br_clear%"
IF %ERRORLEVEL% NEQ 0 goto :ERROR
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
call epmautomate importdata "%importdatajobname%"
IF %ERRORLEVEL% NEQ 0 goto :ERROR

call epmautomate runbusinessrule "%br_calculatebalancesheet%"
IF %ERRORLEVEL% NEQ 0 goto :ERROR

call epmautomate runbusinessrule "%br_allocatetarget%"
"TargetVersion=Baseline"
IF %ERRORLEVEL% NEQ 0 goto :ERROR

for /f %%i in ('call epmautomate listfiles') do if %
%i=="%dataexportfilename%.zip" (call epmautomate deletefile %%i)
IF %ERRORLEVEL% NEQ 0 goto :ERROR

call epmautomate exportdata %exportdatajobname% "%dataexportfilename%.zip"
IF %ERRORLEVEL% NEQ 0 goto :ERROR

call epmautomate downloadfile "%dataexportfilename%.zip"
IF %ERRORLEVEL% NEQ 0 goto :ERROR

rem Section to rename the file

Set Timestamp=%date:~4,2%_%date:~7,2%_%date:~10,4%_%time:~1,1%%time:~3,2%%
ren "%dataexportfilename%.zip" "%dataexportfilename%_%Timestamp%.zip"

call epmautomate logout
IF %ERRORLEVEL% NEQ 0 goto :ERROR

:EOF
echo Scheduled Task Completed successfully
exit /b %errorlevel%

:ERROR
echo Failed with error #%errorlevel%.
exit /b %errorlevel%
```

Exemples de scénarios pour Account Reconciliation

Voir aussi :

- Chargement de soldes préformatés dans une période
  Utilisez ces scripts pour importer les données mappées d'un fichier chargé dans un environnement Account Reconciliation.
- Chargement et import d'un instantané de sauvegarde
  Utilisez ces scripts pour charger et importer un instantané de sauvegarde dans un environnement Account Reconciliation.
- Archivage des anciennes transactions mises en correspondance et purge des transactions archivées
  Utilisez les scripts de cette section pour archiver les transactions mises en correspondance, y compris les détails annexes et les détails d'ajustement, dont l'ancienneté est égale ou supérieure à une valeur donnée, puis purger les transactions archivées d'Account Reconciliation. Les transactions mises en correspondance archivées sont enregistrées dans un fichier ZIP.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Chargement de soldes préformatés dans une période

Utilisez ces scripts pour importer les données mappées d'un fichier chargé dans un environnement Account Reconciliation.

Exemple de script Windows

Créez un fichier nommé `runPreformattedBalances.ps1` en copiant le script suivant. Stockez-le dans un répertoire local.

```
$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -raw)
$username="$($inputproperties.username)"
$passwordfile="$($inputproperties.passwordfile)"
$serviceURL="$($inputproperties.serviceURL)"
$dataFile="$($inputproperties.dataFile)"
$period="$($inputproperties.period)"
$balanceType="$($inputproperties.balanceType)"
$currencyBucket="$($inputproperties.currencyBucket)"

$elements=$dataFile.split('/')
$dataFileName=$elements[-1]

epmautomate login ${username} ${passwordfile} ${serviceURL}
epmautomate uploadfile ${dataFile}
epmautomate importpremappedbalances ${period} ${dataFileName} ${balanceType} $
{currencyBucket}
epmautomate deletefile ${dataFileName}
epmautomate logout
```

Exemple de script Linux/UNIX

Créez un fichier nommé `runPreformattedBalances.sh` en copiant le script suivant. Stockez-le dans un répertoire local.

```
#!/bin/bash

. ./input.properties

export JAVA_HOME=${javahome}

dataFileName=$(echo "${dataFile}" | rev | cut -d'/' -f1 | rev)

${epmautomatescript} login "${username}" "${passwordfile}" "${serviceURL}"
${epmautomatescript} uploadfile "${dataFile}"
${epmautomatescript} importpremappedbalances "${period}" "${dataFileName}" "$
{balanceType}" "${currencyBucket}"
${epmautomatescript} deletefile "${dataFileName}"
${epmautomatescript} logout
```

Exemple de fichier input.properties

Pour exécuter les scripts `runPreformattedBalances`, créez le fichier `input.properties` et mettez-le à jour avec les informations de votre environnement. Enregistrez le fichier dans le

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

répertoire dans lequel `runPreformattedBalances.sh` ou `runPreformattedBalances.ps1` est stocké.

Windows

```
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
dataFile=DATA_FILE_NAME.csv
period=PERIOD_NAME
balanceType=BALANCE_TYPE
currencyBucket=CURRENCY_BUCKET
```

Linux/UNIX

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
dataFile=DATA_FILE_NAME.csv
period=PERIOD_NAME
balanceType=BALANCE_TYPE
currencyBucket=CURRENCY_BUCKET
```

Tableau 3-23 Paramètres du fichier input.properties

Paramètre	Description
`javahome`	Emplacement de `JAVA_HOME`. Pour Linux/UNIX uniquement.
`epmautomatescript`	Chemin absolu de l'exécutable EPM Automate (`epmautomate.sh`). Pour Linux/UNIX uniquement.
`username`	Nom d'utilisateur d'un administrateur de service.
`password`	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
`serviceURL`	URL de l'environnement qui héberge l'application dans laquelle charger des soldes préformatés.
`dataFile`	Fichier CSV qui contient les soldes préformatés (généralement créés à partir d'une comptabilité) à charger dans l'application. Ce fichier doit déjà avoir été chargé dans l'environnement à l'aide de la commande uploadFile.
`period`	Période de rapprochement pour laquelle charger les soldes préformatés.
`balanceType`	Type des soldes préformatés contenus dans `dataFile`.
`currencyBucket`	Catégorie de devise des soldes préformatés.

Exécution du script

1. Créez `runPreformattedBalances.ps1` ou `runPreformattedBalances.sh` en copiant le script d'une section précédente.
2. **Pour Windows et Linux/UNIX uniquement :**

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Créez le fichier `input.properties` et enregistrez-le dans le répertoire où se trouve le script `runPreformattedBalances`. Le contenu de ce fichier varie en fonction du système d'exploitation. Reportez-vous à la section Tableau 1.
  Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter le script.
- Lancez le script.
  - **Windows PowerShell :** exécutez `runPreformattedBalances.ps1`.
  - **Linux/UNIX :** exécutez `./runPreformattedBalances.sh`.

Chargement et import d'un instantané de sauvegarde

Utilisez ces scripts pour charger et importer un instantané de sauvegarde dans un environnement Account Reconciliation.

Exemple de script Windows

Créez un fichier nommé `importBackupSnapshot.ps1` en copiant le script suivant. Stockez-le dans un répertoire local.

```
$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -raw)
$username="$($inputproperties.username)"
$passwordfile="$($inputproperties.passwordfile)"
$serviceURL="$($inputproperties.serviceURL)"
$snapshotName="$($inputproperties.snapshotName)"
$userPassword="$($inputproperties.userPassword)"

epmautomate login ${username} ${passwordfile} ${serviceURL}
epmautomate uploadfile ${snapshotName}.zip
epmautomate importsnapshot ${snapshotName} "userPassword=${userPassword}"
epmautomate deletefile ${snapshotName}.zip
epmautomate logout
```

Exemple de script Linux/UNIX

Créez un fichier nommé `importBackupSnapshot.sh` en copiant le script suivant. Stockez-le dans un répertoire local.

```
#!/bin/bash

. ./input.properties
export JAVA_HOME=${javahome}
${epmautomatescript} login "${username}" "${passwordfile}" "${serviceURL}"
${epmautomatescript} uploadfile "${snapshotName}.zip"
${epmautomatescript} importsnapshot "${snapshotName}" "userPassword=$
{userPassword}"
${epmautomatescript} deletefile "${snapshotName}.zip"
${epmautomatescript} logout
```

Exemple de fichier input.properties

Pour exécuter les scripts `importBackupSnapshot`, créez le fichier `input.properties` et mettez-le à jour avec les informations de votre environnement. Enregistrez le fichier dans le répertoire dans lequel `importBackupSnapshot.sh` ou `importBackupSnapshot.ps1` est stocké.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Windows

```
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
snapshotName=SNAPSHOT_NAME
userPassword=IDM_NEW_USER_PWD
```

Linux/UNIX

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
snapshotName=SNAPSHOT_NAME
userPassword=IDM_NEW_USER_PWD
```

Tableau 3-24 Paramètres du fichier input.properties

Paramètre	Description
`javahome`	Emplacement de `JAVA_HOME`. Pour Linux/UNIX uniquement.
`epmautomatescript`	Chemin absolu de l'exécutable EPM Automate (`epmautomate.sh`). Pour Linux/UNIX uniquement.
`username`	Nom d'utilisateur d'un administrateur de service.
`password`	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
`serviceURL`	URL de l'environnement dans lequel importer l'instantané.
`snapshotName`	Nom de l'instantané à partir duquel les artefacts et les données doivent être importés. Cet instantané doit déjà avoir été chargé dans l'environnement à l'aide de la commande uploadFile.
`userPassword`	Mot de passe par défaut à affecter à tous les utilisateurs créés dans le domaine d'identité consécutivement à l'import de l'instantané.

Exécution du script

1. Créez `importBackupSnapshot.ps1` ou `importBackupSnapshot.sh` en copiant le script d'une section précédente.
2. Créez le fichier `input.properties` et enregistrez-le dans le répertoire où se trouve le script `runPreformattedBalances`. Le contenu de ce fichier varie en fonction du système d'exploitation. Reportez-vous à la section Exemple de fichier input.properties. Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter le script.
3. Lancez le script.
   - **Windows PowerShell :** exécutez `importBackupSnapshot.ps1`.
   - **Linux/UNIX :** exécutez `./importBackupSnapshot.sh`.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Archivage des anciennes transactions mises en correspondance et purge des transactions archivées

Utilisez les scripts de cette section pour archiver les transactions mises en correspondance, y compris les détails annexes et les détails d'ajustement, dont l'ancienneté est égale ou supérieure à une valeur donnée, puis purger les transactions archivées d'Account Reconciliation. Les transactions mises en correspondance archivées sont enregistrées dans un fichier ZIP.

Fonctionnement du script

1. Connexion à l'environnement à l'aide des informations du fichier `input.properties`.
2. Exécution de la commande `archiveTmTransactions` suivante pour créer une archive. Le fichier ZIP et le fichier journal générés utilisent les noms par défaut `Archive_Transactions_INTERCO_`*`JOB_ID`*`.zip` et `Archive_Transactions_INTERCO_`*`JOB_ID`*`.log`.
   ```
   epmautomate archiveTmTransactions INTERCO 365 filterOperator=contains
   filterValue=14001
   ```
   Vous pouvez modifier le fichier `input.properties` pour modifier les paramètres de commande.
3. Téléchargement du fichier journal et du fichier ZIP contenant les transactions archivées sur l'ordinateur local. Le script affiche un message d'erreur si aucune transaction correspondante n'est trouvée.
4. Copie du fichier .zip contenant les transactions archivées vers la banque d'objets Oracle.
5. Exécution de la commande `purgeArchivedTmTransactions` (avec l'ID du job archiveTmTransactions) pour supprimer les transactions mises en correspondance archivées de l'application.

Exécution du script

1. Créez le fichier `input.properties` et mettez-le à jour avec les informations relatives à votre environnement. Enregistrez le fichier dans un répertoire local. Ce répertoire est nommé `parentsnapshotdirectory` dans cette discussion. Le contenu de ce fichier varie en fonction du système d'exploitation.
   Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter les scripts.
2. Créez le script `transaction_match.ps1` (Windows PowerShell) ou `transaction_match.sh` (Linux/UNIX), puis enregistrez-le dans le répertoire `parentsnapshotdirectory`, où se trouve `input.properties`.
3. Lancez le script.
   - Linux/UNIX : exécutez `./transaction_match.sh`.
   - Windows PowerShell : exécutez `transaction_match.ps1`.

Création du script input.properties

Créez `input.properties` en copiant et en mettant à jour le script suivant.

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
epmusername=exampleAdmin1
epmpassword=examplePassword1.epw
epmurl=exampleURL1
objectstorageusername=exampleAdmin2
objectstoragepassword=examplePassword2
objectstorageurl=exampleURL2
matchtype=INTERCO
age=365
filteroperator=contains
filtervalues=FilterValue=14001
proxyserverusername=myProxyserver
proxyserverpassword=myProxyserver_pwd
proxyserverdomain=myProxyDomain
```

Remarque

Windows uniquement : enlevez les propriétés suivantes du fichier `input.properties` :

- `javahome=`*`JAVA_HOME`*
- `epmautomatescript=`*`EPM_AUTOMATE_LOCATION`*

Si l'`authentification au niveau du serveur proxy` n'est pas activée pour votre environnement réseau Windows, enlevez les propriétés suivantes du fichier `input.properties`.

- `proxyserverusername`
- `proxyserverpassword`
- `proxyserverdomain`

Tableau 3-25 Paramètres du fichier input.properties

Paramètre	Description
`javahome`	Répertoire dans lequel le JDK utilisé par EPM Automate est installé. Supprimez cette entrée de la version Windows du fichier `input.properties`. **Exemple :** `javahome=./home/JDK/bin`
`epmautomatescript`	Répertoire dans lequel EPM Automate est installé. Supprimez cette entrée de la version Windows du fichier `input.properties`. **Exemple :**`epmautomatescript=./home/utils/EPMAutomate/bin`
`epmusername`	Nom d'utilisateur d'un administrateur de service, d'un super utilisateur, d'un utilisateur ou d'un visualiseur autorisé à archiver des transactions mises en correspondance. **Exemple :** `epmusername=ServiceAdmin`
`epmuserpassword`	Fichier de mots de passe cryptés de l'utilisateur identifié comme `epmusername`. **Exemple :** `epmpassword=myPwd.epw`
`epmurl`	URL de l'environnement dans lequel les transactions mises en correspondance doivent être archivées. **Exemple :** `epmurl=https://test-cloudpln.pbcs.us1.oraclecloud.com`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 3-25 (suite) Paramètres du fichier input.properties

Paramètre	Description
`objectstorageusername`	ID d'un utilisateur qui dispose des droits d'accès en écriture requis dans Oracle Object Storage Cloud. Pour les utilisateurs créés dans un fournisseur d'identités fédérées, indiquez le nom qualifié complet de l'utilisateur (par exemple, `exampleIdP/jdoe` ou `exampleIdP/john.doe@example.com`, où `exampleIdP` est le nom du fournisseur d'identités fédérées). Pour les autres utilisateurs, indiquez l'ID utilisateur. **Exemple :** `epmusername=myIdP/jdoe`
`objectstoragepassword`	Mot de passe Swift ou jeton d'authentification associé à l'utilisateur identifié dans `objectstorageusername`. Ce mot de passe est différent de celui dont l'utilisateur se sert pour se connecter à la console Object Storage. Le jeton d'authentification est un jeton généré par Oracle que vous utilisez pour l'authentification auprès d'API tierces, par exemple auprès d'un client Swift. Afin d'obtenir des instructions pour créer ce jeton, reportez-vous à la section Procédure de création d'un jeton d'authentification du guide *Documentation Oracle Cloud Infrastructure*. **Exemple :** `objectstoragepassword=jDoe_PWD`
`objectstorageurl`	URL du bucket Oracle Object Storage Cloud à laquelle est ajouté un nom d'objet facultatif. **Exemple :** `objectstorageurl=https://swiftobjectstorage.region_identifier.oraclecloud.com/v1/namespace/MT_Archives/2023_archives`
`matchtype`	Identificateur (TextID) du type de correspondance dont les transactions mises en correspondance doivent être archivées. **Exemple :** `matchtype=cashrecon`
`age`	Nombre de jours écoulés depuis la mise en correspondance de la transaction. Toute transaction mise en correspondance antérieure ou égale à cette valeur sera archivée. **Exemple :** `age=180`
`filteroperator`	Conditions de filtre permettant d'identifier les comptes qui contiennent des transactions mises en correspondance pour archivage. Valeurs admises : `equals`, `not_equals`, `starts_with`, `ends_with`, `contains` ou `not_contains`. Cette valeur est associée à `filterValue` pour identifier les comptes dont les transactions mises en correspondance doivent être archivées. **Exemple :** `filteroperator=not_equals`
`filtervalues`	Valeurs de filtre permettant d'identifier les transactions à archiver. Si `equals` ou `not_equals` est indiqué en tant que `filterOperator`, vous pouvez utiliser une liste de valeurs séparées par des espaces pour indiquer plusieurs valeurs. Si plusieurs valeurs sont indiquées, les transactions provenant des comptes correspondant à toute combinaison d'opérateur de filtre et de valeur de filtre sont sélectionnées afin d'être archivées. **Exemple :** `filterValue=101-120 filterValue=140-202`
`proxyserverusername`	Nom d'utilisateur permettant d'authentifier une session sécurisée avec le serveur proxy qui contrôle l'accès à Internet. **Exemple :** `proxyserverusername=myProxyserver`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 3-25 (suite) Paramètres du fichier input.properties

Paramètre	Description
proxyserverpassword	Mot de passe pour l'authentification de l'utilisateur auprès du serveur proxy. **Exemple :** proxyserverpassword=myProxyserver_pwd
proxyserverdomain	Nom du domaine défini pour le serveur proxy. **Exemple :** proxyserverdomain=myProxyDomain

Script Windows

Créez transaction_match.ps1 en copiant le script suivant. Stockez-le dans le même dossier que le fichier input.properties.

```
# Archive and Purge Transaction Matching PowerShell script

$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -raw)
$epmusername="$($inputproperties.epmusername)"
$epmpassword="$($inputproperties.epmpassword)"
$epmurl="$($inputproperties.epmurl)"
$objectstorageusername="$($inputproperties.objectstorageusername)"
$objectstoragepassword="$($inputproperties.objectstoragepassword)"
$objectstorageurl="$($inputproperties.objectstorageurl)"
$matchtype="$($inputproperties.matchtype)"
$age="$($inputproperties.age)"
$filteroperator="$($inputproperties.filteroperator)"
$filtervalues="$($inputproperties.filtervalues)"
$proxyserverusername="$($inputproperties.proxyserverusername)"
$proxyserverpassword="$($inputproperties.proxyserverpassword)"
$proxyserverdomain="$($inputproperties.proxyserverdomain)"

echo "Running processes to archive and purge transaction matching
transactions ..."
if (${proxyserverusername}) {
    echo "Running epmautomate login ${epmusername} ${epmpassword} ${epmurl} $
{proxyserverusername} ${proxyserverpassword} ${proxyserverdomain}"
    epmautomate login ${epmusername} ${epmpassword} ${epmurl} $
{proxyserverusername} ${proxyserverpassword} ${proxyserverdomain}
} else {
    echo "Running epmautomate login ${epmusername} ${epmpassword} ${epmurl}"
    epmautomate login ${epmusername} ${epmpassword} ${epmurl}
}
echo "Running epmautomate archiveTmTransactions \"${matchtype}\" ${age}
filterOperator=${filteroperator} ${filtervalues}"
epmautomate archiveTmTransactions "${matchtype}" "${age}" "filterOperator=$
{filteroperator}" "${filtervalues}" > ./outfile.txt.tmp

$jobIdLine=Select-String -Path "outfile.txt.tmp" -Pattern "Job
ID"; $jobIdLine=($jobIdLine -split ":")[-2]; $jobid=($jobIdLine -split " ")
[1];
$logfile="Archive_Transactions_${matchtype}_${jobid}.log"
$transactionsfile="Archive_Transactions_${matchtype}_${jobid}.zip"
epmautomate listfiles > ./files.txt.tmp
$transactionslogfound=Select-String -Path "./files.txt.tmp" -Pattern "$
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
{logfile}"
$transactionsfilefound=Select-String -Path "./files.txt.tmp" -Pattern "$
{transactionsfile}"
rm ./outfile.txt.tmp
rm ./files.txt.tmp

if (${transactionslogfound}) {
    echo "Running epmautomate downloadfile \"${logfile}\""
    epmautomate downloadfile "${logfile}"
    if (${transactionsfilefound}) {
        echo "Running epmautomate downloadfile ${transactionsfile}"
        epmautomate downloadfile "${transactionsfile}"
        if ($?) {
            echo "Running epmautomate copyToObjectStorage $
{transactionsfile} ${objectstorageusername} ${objectstoragepassword} $
{objectstorageurl}"
            epmautomate copyToObjectStorage "${transactionsfile}" "$
{objectstorageusername}" "${objectstoragepassword}" "${objectstorageurl}"
            if ($?) {
                echo "Running epmautomate purgeArchivedTMTransactions JOBID=$
{jobid}"
                epmautomate purgeArchivedTMTransactions "JobID=${jobid}"
            }
        else {
                echo "EPMAutomate copyToObjectStorage failed. Purging of
archived matched transactions will not be attempted."
            }
    }
    else {
            echo "Download of transactions file ${transactionsfile} failed.
Copy to object storage, and purging of archived matched transactions will not
be attempted."
    }
    }
    else {
        echo "No matched transactions found. Archive file download, copy to
object storage, and purging of archived matched transactions will not be
attempted."
    }
}
else {
    echo "Matchtype ID \"${matchtype}\" not found. Archive file download,
copy to object storage, and purging of archived matched transactions will not
be attempted."
}

echo "Running epmautomate logout"
epmautomate logout
echo "Script processing completed."
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Script Linux/UNIX

Créez `transaction_match.sh` en copiant le script suivant. Stockez-le dans le même dossier que le fichier `input.properties`.

```
#!/bin/sh

. ./input.properties
export JAVA_HOME=${javahome}

echo "Running processes to archive and purge transaction matching
transactions..."
if [[ "${proxyserverusername}" != "" ]]
then
    echo "Running epmautomate login ${epmusername} ${epmpassword} ${epmurl}
ProxyServerUserName=${proxyserverusername} ProxyServerPassword=$
{proxyserverpassword} ProxyServerDomain=${proxyserverdomain}"
    ${epmautomatescript} login "${epmusername}" "${epmpassword}" "${epmurl}"
"ProxyServerUserName=${proxyserverusername}" "ProxyServerPassword=$
{proxyserverpassword}" "ProxyServerDomain=${proxyserverdomain}"
else
    echo "Running epmautomate login ${epmusername} ${epmpassword} ${epmurl}"
    ${epmautomatescript} login "${epmusername}" "${epmpassword}" "${epmurl}"
fi
echo "Running epmautomate archiveTmTransactions \"${matchtype}\" ${age}
filterOperator=${filteroperator} ${filtervalues}"
${epmautomatescript} archiveTmTransactions "${matchtype}" "${age}"
"filterOperator=${filteroperator}" "${filtervalues}" > ./output.txt.tmp

jobid=$(grep "Job ID" ./output.txt.tmp | cut -d':' -f3 | cut -d' ' -f2)
logfile="Archive_Transactions_${matchtype}_${jobid}.log"
transactionsfile="Archive_Transactions_${matchtype}_${jobid}.zip"
${epmautomatescript} listfiles > ./files.txt.tmp
transactionslogfound=$(grep "${logfile}" ./files.txt.tmp | wc -l)
transactionsfilefound=$(grep "${transactionsfile}" ./files.txt.tmp | wc -l)
rm ./files.txt.tmp
rm ./output.txt.tmp

if [ ${transactionslogfound} -eq 0 ]
then
    echo "Matchtype ID \"${matchtype}\" not found. Archive file download,
copy to object storage, and purging of archived matched transactions will not
be attempted."
else
    echo "Running epmautomate downloadfile \"${logfile}\""
    ${epmautomatescript} downloadfile "${logfile}"
    if [ ${transactionsfilefound} -eq 0 ]
    then
        echo "No matched transactions found. Archive file download, copy to
object storage, and purging of archived matched transactions will not be
attempted."
    else
        echo "Running epmautomate downloadfile ${transactionsfile}"
        ${epmautomatescript} downloadfile "${transactionsfile}"
        if [ $? -eq 0 ]
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
        then
            echo "Running epmautomate copyToObjectStorage $
{transactionsfile} ${objectstorageusername} ${objectstoragepassword} $
{objectstorageurl}"
            ${epmautomatescript} copyToObjectStorage "${transactionsfile}" "$
{objectstorageusername}" "${objectstoragepassword}" "${objectstorageurl}"
            if [ $? -eq 0 ]
            then
                echo "Running epmautomate purgeArchivedTMTransactions JOBID=$
{jobid}"
                ${epmautomatescript} purgeArchivedTMTransactions "JobID=$
{jobid}"
            else
                echo "EPMAutomate copyToObjectStorage failed. Purging of
archived matched transactions will not be attempted."
            fi
        else
            echo "Download of transactions file ${transactionsfile} failed.
Copy to object storage, and purging of archived matched transactions will not
be attempted."
        fi
    fi
fi

echo "Running epmautomate logout"
${epmautomatescript} logout
echo "Script processing completed."
```

Exemples de scénarios pour Profitability and Cost Management

Ces scénarios portent sur l'utilisation des commandes EPM Automate permettant d'effectuer certaines tâches Profitability and Cost Management courantes.

Voir aussi :

- Import de métadonnées dans une application
  Utilisez ces scripts pour charger un fichier de métadonnées et importer les métadonnées de dimension de ce dernier dans une application Profitability and Cost Management.
- Import de données et exécution de règles de programme
  Utilisez ces scripts pour charger des fichiers de données et importer les données des fichiers chargés dans un processus métier Profitability and Cost Management.

Import de métadonnées dans une application

Utilisez ces scripts pour charger un fichier de métadonnées et importer les métadonnées de dimension de ce dernier dans une application Profitability and Cost Management.

Les scripts effectuent les opérations suivantes :

- Connexion à l'environnement.
- Chargement d'un fichier de métadonnées.
- Import des métadonnées du fichier chargé vers une application.
- Activation de l'application.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Déconnexion.

Script Windows

Créez `importMetadata.ps1` en copiant le script suivant.

```
$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -raw)
$username="$($inputproperties.username)"
$passwordfile="$($inputproperties.passwordfile)"
$serviceURL="$($inputproperties.serviceURL)"
$applicationName="$($inputproperties.applicationName)"
$dataFileName="$($inputproperties.dataFileName)"
$dataFileNameDestination="$($inputproperties.dataFileNameDestination)"

epmautomate login ${username} ${passwordfile} ${serviceURL}
epmautomate uploadfile "${dataFileName}" ${dataFileNameDestination}
epmautomate loaddimdata ${applicationName} dataFileName=${dataFileName}
epmautomate enableapp ${applicationName}
epmautomate logout
```

Script Linux/UNIX

Créez `importMetadata.sh` en copiant le script suivant.

```
#!/bin/bash
. ./input.properties
export JAVA_HOME=${javahome}
${epmautomatescript} login "${username}" "${passwordfile}" "${serviceURL}"
${epmautomatescript} uploadfile "${dataFileName}" "${dataFileNameDestination}"
${epmautomatescript} loaddimdata "${applicationName}" "dataFileName=$
{dataFileName}"
${epmautomatescript} enableapp "${applicationName}"
${epmautomatescript} logout
```

Création du fichier input.properties

Pour exécuter les scripts `importMetadata`, créez le fichier `input.properties` et mettez-le à jour avec les informations de votre environnement. Enregistrez le fichier dans le répertoire dans lequel `importMetadata.ps1` ou `importMetadata.sh` est stocké.
Windows

```
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
applicationName=APPLICATION_NAME
dataFileName=DATA_FILE.txt
dataFileNameDestination=profitinbox
```

Linux/UNIX

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
username=exampleAdmin
passwordfile=examplePassword.epw
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
serviceURL=exampleURL
applicationName=APPLICATION_NAME
dataFileName=DATA_FILE.txt
dataFileNameDestination=profitinbox
```

Tableau 3-26 Paramètres du fichier input.properties

Paramètre	Description
javahome	Emplacement de JAVA_HOME. Pour Linux/UNIX uniquement.
epmautomatescript	Chemin absolu de l'exécutable EPM Automate (epmautomate.sh). Pour Linux/UNIX uniquement.
username	Nom d'utilisateur d'un administrateur de service disposant également du rôle Administrateur de domaine d'identité.
password	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
serviceURL	URL de l'environnement à partir duquel générer l'instantané.
applicationName	Nom de l'application Profitability and Cost Management dans laquelle charger les données.
dataFileName	Nom du fichier contenant les métadonnées à importer.
dataFileNameDestination	Emplacement de chargement du fichier de métadonnées.

Exécution des scripts

1. Créez `importMetadata.ps1` ou `importMetadata.sh` en copiant le script d'une section précédente.
2. Créez le fichier `input.properties` et enregistrez-le dans le répertoire où se trouve le script `importMetadata`. Le contenu de ce fichier varie en fonction du système d'exploitation. Reportez-vous à la section Création du fichier input.properties.
   Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter le script.
3. Lancez le script.
   - **Windows PowerShell :** exécutez `importMetadata.ps1`.
   - **Linux/UNIX :** exécutez `./importMetadata.sh`.

Import de données et exécution de règles de programme

Utilisez ces scripts pour charger des fichiers de données et importer les données des fichiers chargés dans un processus métier Profitability and Cost Management.

Les scripts effectuent les étapes suivantes :

- Connexion à l'environnement.
- Chargement d'un fichier contenant les données à charger.
- Chargement d'un fichier contenant des règles de données.
- Chargement des données du fichier de données dans l'application pour remplacer les valeurs existantes.
- Exécution de toutes les règles du fichier de règles.
- Déconnexion.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Script Windows

Créez un fichier nommé `importData.ps1` en copiant le script suivant. Stockez-le dans un répertoire local.

```
$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -raw)
$username="$($inputproperties.username)"
$passwordfile="$($inputproperties.passwordfile)"
$serviceURL="$($inputproperties.serviceURL)"
$applicationName="$($inputproperties.applicationName)"
$dataFileName="$($inputproperties.dataFileName)"
$rulesFileName="$($inputproperties.rulesFileName)"
$fileDestination="$($inputproperties.fileDestination)"
$clearAllDataFlag="$($inputproperties.clearAllDataFlag)"
$dataLoadValue="$($inputproperties.dataLoadValue)"

epmautomate login ${username} ${passwordfile} ${serviceURL}
epmautomate uploadfile "${dataFileName}" ${fileDestination}
epmautomate uploadfile "${rulesFileName}" ${fileDestination}
epmautomate loaddata ${applicationName} clearAllDataFlag=${clearAllDataFlag}
dataLoadValue=${dataLoadValue} rulesFileName="${rulesFileName}"
dataFileName="${dataFileName}"
epmautomate logout
```

Script Linux/UNIX

Créez un fichier nommé `importData.sh` en copiant le script suivant. Stockez-le dans un répertoire local.

```
#!/bin/bash
. ./input.properties
export JAVA_HOME=${javahome}
${epmautomatescript} login "${username}" "${passwordfile}" "${serviceURL}"
${epmautomatescript} uploadfile "${dataFileName}" "${fileDestination}"
${epmautomatescript} uploadfile "${rulesFileName}" "${fileDestination}"
${epmautomatescript} loaddata "${applicationName}" "clearAllDataFlag=$
{clearAllDataFlag}" "dataLoadValue=${dataLoadValue}" rulesFileName="$
{rulesFileName}" dataFileName="${dataFileName}"
${epmautomatescript} logout
```

Création du fichier input.properties

Pour exécuter les scripts `importData`, créez le fichier `input.properties` et mettez-le à jour avec les informations de votre environnement. Enregistrez le fichier dans le répertoire dans lequel `importData.ps1` ou `importData.sh` est stocké.
Windows

```
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
applicationName=APPLICATION_NAME
dataFileName=DATA_FILE.txt
rulesFileName=RULE_FILE.txt
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
fileDestination=profitinbox
clearAllDataFlag=true
dataLoadValue=OVERWRITE_EXISTING_VALUES
```

Linux/UNIX

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
applicationName=APPLICATION_NAME
dataFileName=DATA_FILE.txt
rulesFileName=RULE_FILE.txt
fileDestination=profitinbox
clearAllDataFlag=true
dataLoadValue=OVERWRITE_EXISTING_VALUES
```

Tableau 3-27 Paramètres du fichier input.properties

Paramètre	Description
`javahome`	Emplacement de `JAVA_HOME`. Pour Linux/UNIX uniquement.
`epmautomatescript`	Chemin absolu de l'exécutable EPM Automate (`epmautomate.sh`). Pour Linux/UNIX uniquement.
`username`	Nom d'utilisateur d'un administrateur de service disposant également du rôle Administrateur de domaine d'identité.
`password`	Mot de passe de l'administrateur de service, ou nom et emplacement du fichier de mots de passe cryptés.
`serviceURL`	URL de l'environnement à partir duquel générer l'instantané.
`applicationName`	Nom de l'application Profitability and Cost Management dans laquelle charger les données.
`dataFileName`	Nom du fichier contenant les données à importer.
`rulesFileName`	Nom du fichier contenant les règles à exécuter sur les données importées.
`fileDestination`	Emplacement dans lequel charger les fichiers de données et de règles.
`clearAllDataFlag`	Indique si les données existantes doivent être effacées dans le cube d'application. Définissez ce paramètre sur `false` si vous ne voulez pas effacer les données existantes.
`dataLoadValue`	Indique comment gérer les données existantes. Indiquez `ADD_TO_EXISTING` pour conserver les données existantes du cube.

Exécution des scripts

1. Créez `importData.ps1` ou `importData.sh` en copiant le script d'une section précédente.
2. Créez le fichier `input.properties` et enregistrez-le dans le répertoire où se trouve le script `importData`. Le contenu de ce fichier varie en fonction du système d'exploitation. Reportez-vous à la section Création du fichier input.properties.
   Assurez-vous que vous disposez de privilèges d'écriture sur ce répertoire. Pour Windows, vous devrez peut-être démarrer PowerShell à l'aide de l'option **Exécuter en tant qu'administrateur** afin de pouvoir exécuter le script.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

3. Lancez le script.
   - **Windows PowerShell :** exécutez `importData.ps1`.
   - **Linux/UNIX :** exécutez `./importData.sh`.

Exemples de scénarios pour Cloud EDM

Ces exemples de scénario portent sur l'utilisation des commandes EPM Automate permettant de synchroniser les dimensions d'application entre Oracle Fusion Cloud Enterprise Data Management et Oracle Fusion Cloud Enterprise Performance Management.

Voir aussi :

- Synchronisation de dimensions et de mappings Cloud EDM avec des applications Cloud EPM
  Cet exemple de scénario porte sur la synchronisation d'une dimension entre une application Oracle Fusion Cloud Enterprise Data Management et une application Oracle Fusion Cloud Enterprise Performance Management.
- Synchronisation de dimensions Cloud EPM avec des applications Cloud EDM
  Cet exemple de scénario porte sur la synchronisation d'une dimension entre une application Oracle Fusion Cloud Enterprise Performance Management et une application Oracle Fusion Cloud Enterprise Data Management.

Synchronisation de dimensions et de mappings Cloud EDM avec des applications Cloud EPM

Cet exemple de scénario porte sur la synchronisation d'une dimension entre une application Oracle Fusion Cloud Enterprise Data Management et une application Oracle Fusion Cloud Enterprise Performance Management.

Les scripts de cette section vous permettent d'effectuer les tâches suivantes :

- Exporter une dimension à partir d'une application Cloud EDM
- Exporter des mappings à partir d'une dimension d'application Cloud EDM
- Copier les fichiers d'export vers un environnement Cloud EPM
- Importer des mappings et des métadonnées de dimension dans l'application Cloud EPM

Pour synchroniser une dimension et des mappings entre une application Cloud EDM et une application Cloud EPM, procédez comme suit :

1. Créez un fichier de script en copiant le script suivant :

```
rem Integration example to sync application dimensions between Cloud EDM
and Cloud EPM
rem Windows script for demonstration purposes only; do not use in
production environments

set EDMUSER=userid
set EDMSVR=https://hostname
set EDMPWDFILE=example_EDM
set EDMAPP=appname
set EDMDIM=dimname
set EDMLOC=location
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
set EPMUSER=userid
set EPMSVR=https://hostname
set EPMIMPJOB=importjobname
set PWDFILE=C:\Oracle\EPM.epw
set DIMFILE=dimension.csv
set MAPFILE=mapping.csv

rem Synchronizing EDM ---> EPM
rem Export Dimension and Mappings from EDM

call epmautomate login %EDMUSER% %EDMPWDFILE% %EDMSVR%
call epmautomate exportdimension %EDMAPP% %EDMDIM% %DIMFILE%
call epmautomate exportdimensionmapping %EDMAPP% %EDMDIM% %EDMLOC%
%MAPFILE%
call epmautomate logout

rem Log into the Cloud EPM environment
call epmautomate login %EPMUSER% %PWDFILE% %EPMSVR%

rem Copy exported files from EDM environment to EPM and import metadata
and mappings
call epmautomate copyfilefrominstance %DIMFILE% %EDMUSER% %EDMPWDFILE%
%EDMSVR% inbox/%DIMFILE%
call epmautomate importmetadata %EPMIMPJOB%

call epmautomate copyfilefrominstance %MAPFILE% %EDMUSER% %EDMPWDFILE%
%EDMSVR% inbox/%MAPFILE%
call epmautomate importmapping %EDMDIM% %MAPFILE% REPLACE FALSE %EDMLOC%

call epmautomate logout
```

2. Modifiez le fichier de script et définissez les valeurs de paramètre requises. Pour obtenir des explications concernant les paramètres ainsi que des exemples, reportez-vous à la section Paramètres d'exécution de script.
3. Exécutez le script manuellement ou planifiez l'exécution selon vos besoins. Reportez-vous à la section Automatisation de l'exécution de scripts.

Paramètres d'exécution de script

Pour les fichiers de script de cette section, vous devez indiquer certaines valeurs de paramètre expliquées dans le tableau suivant. Tous ces paramètres ne sont pas utilisés dans tous les scripts.

Tableau 3-28 Valeurs de paramètre pour les fichiers de script

Paramètre	Description
EDMUSER	ID de connexion utilisateur d'un administrateur de service Cloud EDM. **Exemple :** EDMUSER=jdoe@example.com
EDMSVR	URL de l'environnement Cloud EDM. **Exemple :** EDMSVR=https:// example.oraclecloud.com
EDMPWDFILE	Nom et emplacement du fichier de mots de passe cryptés (EPW) de l'administrateur de service Cloud EDM. **Exemple :** EDMPWDFILE=edm_jdoe.epw

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Tableau 3-28 (suite) Valeurs de paramètre pour les fichiers de script

Paramètre	Description
EDMAPP	Nom d'une dimension d'application Cloud EDM. **Exemple :** EDMAPP=USOperations
EDMDIM	Nom de la dimension à exporter ou à importer. **Exemple :** EDMDIM=entity
EDMLOC	Nom de l'emplacement à exporter. **Exemple :** EDMLOC=Loc1
EPMUSER	Nom de connexion d'un administrateur de service Cloud EPM. **Exemple :** EPMUSER=john.doe@example.com
EPMSVR	URL de l'environnement Cloud EPM. **Exemple :** EPMSVR=https://example.oraclecloud.com
EPMIMPJOB	Nom d'un job d'import existant de type Importer les métadonnées dans l'environnement Cloud EPM. **Exemple :** EPMIMPJOB=imp_DIMMetadata
EPMEXPJOB	Nom d'un job existant de type Exporter les métadonnées dans l'environnement Cloud EPM. **Exemple :** EPMEXPJOB=Exp_DIMMetadata
PWDFILE	Nom et emplacement du fichier de mots de passe cryptés (EPW) de l'administrateur de service Cloud EPM. Reportez-vous à la commande encrypt. **Exemple :** PWDFILE=pwd_jdoe.epw
DIMFILE	Nom du fichier destiné à contenir les données de dimension exportées. **Exemple :** DIMFILE=entity_file.CSV
MAPFILE	Nom du fichier destiné à contenir les données de mapping exportées. **Exemple :** MAPFILE=map_file.CSV

Synchronisation de dimensions Cloud EPM avec des applications Cloud EDM

Cet exemple de scénario porte sur la synchronisation d'une dimension entre une application Oracle Fusion Cloud Enterprise Performance Management et une application Oracle Fusion Cloud Enterprise Data Management.

Les scripts de cette section vous permettent d'effectuer les tâches suivantes :

- Exporter des métadonnées (dimensions) à partir d'une application Cloud EPM
- Copier les fichiers d'export contenant les données de dimension dans un environnement Cloud EDM
- Importer des métadonnées de dimension dans l'application Cloud EDM

Pour synchroniser une dimension entre une application Cloud EPM et une application Cloud EDM, procédez comme suit :

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

1. Créez un fichier de script Windows en copiant le script suivant :

```
rem Integration example to sync an application dimension between Cloud EPM
and Cloud EDM
rem Windows script for demonstration purposes only; do not use in
production environments

set EDMUSER=userid
set EDMSVR=https://hostname
set EDMPWDFILE=example_EDM.epw
set EDMAPP=appname
set EDMDIM=dimname

set EPMUSER=userid
set EPMSVR=https://hostname
set PWDFILE=example_epm.epw
set EPMEXPJOB=exportjobname

rem Synchronizing EPM ---> EDM

rem Export Metadata from EPM
call epmautomate login %EPMUSER% %PWDFILE% %EPMSVR%
call epmautomate exportmetadata %EPMEXPJOB%
call epmautomate logout

rem Import Dimension to EDM
rem Log into the EDM environment
call epmautomate login %EDMUSER% %EDMPWDFILE% %EDMSVR%
rem Copy exported metadata file into the EDM environment
call epmautomate copyfilefrominstance %EPMEXPJOB%.zip %EPMUSER% %PWDFILE%
%EPMSVR% %EPMEXPJOB%.zip
call epmautomate importdimension %EDMAPP% %EDMDIM% ReplaceNodes
%EPMEXPJOB%.zip
call epmautomate logout
```

   Modifiez le fichier de script et définissez les valeurs de paramètre requises. Pour obtenir des explications concernant les paramètres ainsi que des exemples, reportez-vous à la section Paramètres d'exécution de script.

2. Exécutez le script manuellement ou planifiez l'exécution selon vos besoins. Reportez-vous à la section Automatisation de l'exécution de scripts.

Automatisation de l'exécution de scripts

Un administrateur de service planifie des scripts dans le planificateur de tâches Windows ou utilise un job Cron pour automatiser des activités à l'aide d'EPM Automate.

Pour planifier l'exécution de script EPM Automate à l'aide du planificateur de tâches Windows, procédez comme suit :

1. Cliquez sur **Démarrer**, puis **Panneau de configuration** et **Outils d'administration**.
2. Ouvrez le **Planificateur de tâches**.
3. Sélectionnez **Action** et **Créer une tâche de base**.
4. Saisissez un nom de tâche et une description facultative, puis cliquez sur **Suivant**.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

5. Dans **Déclencheur de tâche**, sélectionnez une planification pour exécuter le script, puis cliquez sur **Suivant**.
6. Dans l'écran suivant, indiquez d'autres paramètres de planification, puis cliquez sur **Suivant**.
7. Dans Action, assurez-vous que l'option **Démarrer un programme** est sélectionnée.
8. Dans **Démarrer un programme**, procédez comme suit :
   a. Dans **Programme/script**, parcourez la liste et sélectionnez le script que vous souhaitez planifier.
   b. Dans **Ajouter des arguments (facultatif)**, saisissez le mot de passe de l'administrateur de service identifié par le paramètre de script `SET user`.
   c. Dans **Commencer dans (facultatif)**, saisissez l'emplacement où EPM Automate est installé. En règle générale, il s'agit de `C:/Oracle/EPMAutomate/bin`.
   d. Cliquez sur **Suivant**.
9. Dans **Résumé**, sélectionnez **Ouvrir les propriétés de cette tâche quand j'aurai cliqué sur Terminer**, puis cliquez sur **Terminer**.
10. Dans **Général**, sélectionnez ces options de sécurité, puis cliquez sur **OK**.
    - **Exécuter même si l'utilisateur n'est pas connecté**
    - **Exécuter avec les autorisations maximales**

Surveillance des activités EPM Automate

Pour vous aider à identifier le statut de l'opération que vous avez initialisée, EPM Automate affiche des codes de statut dans la console à partir de laquelle vous l'exécutez.

Reportez-vous à la section Codes de sortie.

Utilisez la console des jobs pour surveiller les jobs que vous exécutez à l'aide d'EPM Automate. Reportez-vous à la section Gestion des jobs du guide *Administration de Planning* pour plus de détails.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

4

Exécution de commandes sans installer EPM Automate

A l'aide de Groovy, vous pouvez exécuter certaines commandes directement dans Oracle Fusion Cloud Enterprise Performance Management. Il n'est pas nécessaire d'installer EPM Automate pour exécuter des commandes côté serveur.

> **Note**
>
> Dans ce scénario, le script Groovy est écrit pour être exécuté directement dans Cloud EPM, et non sur n'importe quel ordinateur client.

Ce chapitre contient les sections suivantes :

- Environnements prenant en charge l'exécution de commandes côté serveur
- Sources d'information
- Commandes prises en charge
- Méthodes à utiliser pour l'exécution d'EPM Automate à l'aide de Groovy côté serveur
- Clonage d'un environnement à l'aide d'un script Groovy côté serveur
- Envoi par courriel du rapport d'activité à l'aide d'un script Groovy côté serveur

Environnements prenant en charge l'exécution de commandes côté serveur

La prise en charge des scripts Groovy pour l'exécution côté serveur des commandes EPM Automate n'est disponible que dans les environnements suivants :

- Processus métier Planning et de modules Planning déployés dans des environnements EPM Enterprise Cloud Service
- Enterprise Planning and Budgeting Cloud
- Planning and Budgeting Cloud avec option Plus One
- FreeForm
- Enterprise Profitability and Cost Management
- Financial Consolidation and Close dans des environnements EPM Enterprise Cloud Service
- Tax Reporting dans des environnements EPM Enterprise Cloud Service
- Strategic Workforce Planning
- Sales Planning

Les scripts Groovy incorporant des commandes EPM Automate peuvent être écrits et exécutés uniquement dans les environnements Oracle Fusion Cloud Enterprise Performance

Copyright © 2016, 2025, Oracle and/or its affiliates.

Management précédents. Les scripts écrits dans de tels environnements peuvent cependant émettre des commandes EPM Automate sur n'importe quel environnement Cloud EPM. Vous pouvez par exemple créer le script dans un environnement Planning EPM Enterprise Cloud Service, puis lui faire émettre des commandes sur un environnement Narrative Reporting qui ne prend pas en charge les scripts Groovy.

Sources d'information

Pour obtenir des informations détaillées, reportez-vous aux sections suivantes du guide *Conception avec Calculation Manager* :

- A propos des règles métier Groovy
- Création d'une règle métier Groovy

Commandes prises en charge

Toutes les commandes EPM Automate peuvent être exécutées via Groovy, à l'exception des suivantes :

- downloadFile
- upgrade
- uploadFile

Les commandes suivantes ne peuvent pas être exécutées sur l'environnement qui exécute le script Groovy :

- recreate
- replay
- resetService
- restructureCube

> **Note**
>
> - Lors de l'exécution de la commande encrypt, le fichier de mots de passe crypté est créé sur le serveur et conservé pour une utilisation à long terme. Le fichier de mots de passe crypté n'apparaît pas si vous exécutez la commande listFiles. Vous ne pouvez pas le télécharger à partir du serveur à l'aide de la commande downloadFile.
> - Pour que la commande feedback fonctionne, toutes les captures d'écran et tous les fichiers joints doivent être disponibles dans l'emplacement de chargement par défaut. Reportez-vous à la section Emplacement de chargement par défaut. Il s'agit de l'emplacement où les fichiers sont stockés si vous n'indiquez pas d'emplacement dans la commande uploadFile.

Méthodes à utiliser pour l'exécution d'EPM Automate à l'aide de Groovy côté serveur

- `getEPMAutomate ()` : cette méthode statique fournit une instance de classe `EpmAutomate`, qui peut ensuite être utilisée pour appeler des commandes EPM Automate.

Copyright © 2016, 2025, Oracle and/or its affiliates.

- `execute ()` : cette méthode de classe `EpmAutomate` est utilisée pour exécuter une commande EPM Automate. Transmettez le nom de la commande EPM Automate en tant que premier paramètre et les options de commande en tant que paramètres suivants. Cette méthode renvoie une instance de classe `EpmAutomateStatus`.
- `getStatus ()` : cette méthode de classe `EPMAutomateStatus` renvoie le statut d'exécution renvoyé par la commande. Une valeur renvoyée de 0 indique que la commande a réussi, tandis qu'une valeur autre que 0 indique l'échec de la commande.
- `getOutput ()` : cette méthode de classe `EPMAutomateStatus` renvoie la sortie de chaîne de la commande. Vous pouvez par exemple utiliser cette méthode pour renvoyer la sortie des commandes `getApplicationAdminMode` et `getDailyMaintenanceStartTime`. Si le statut renvoyé de la commande n'est pas 0, cette méthode renvoie le message d'erreur.
- `getItemsList ()` : cette méthode de classe `EPMAutomateStatus` renvoie la sortie de liste de la commande. Vous pouvez par exemple utiliser cette méthode pour renvoyer la sortie des commandes `getSubstVar`, `listBackups` et `listFiles`.

Clonage d'un environnement à l'aide d'un script Groovy côté serveur

Vous pouvez inclure des commandes EPM Automate dans des scripts Groovy côté serveur pour cloner des environnements à des fins de récupération après sinistre. Cela permet la configuration d'une récupération après sinistre sans aucune empreinte sur site.

Si les mots de passe contiennent des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux. Veillez également à remplacer ces valeurs de paramètre en fonction de vos environnements :

Table 4-1 Paramètres à modifier

Paramètre	Description
`password`	Mot de passe de l'administrateur de service qui effectue l'opération de clonage dans l'environnement source.
`targetpassword`	Mot de passe de l'administrateur de service qui effectue l'opération de clonage dans l'environnement cible.
`username`	ID utilisateur de l'administrateur de service dans l'environnement source.
`targetusername`	ID utilisateur de l'administrateur de service dans l'environnement cible. Cet utilisateur doit également être affecté au rôle Administrateur de domaine d'identité dans l'environnement cible.
`email_id`	Adresse électronique à laquelle vous prévoyez d'envoyer des informations à propos du processus de clonage.

Script pour le cryptage d'un mot de passe

```
EpmAutomate automate = getEpmAutomate()

//Encrypt the password of a Service Administrator in the source environment
EpmAutomateStatus encryptstatus1 = automate.execute('encrypt',
'password','encryptionKey','sourcePassword.epw')
if(encryptstatus1.getStatus() != 0)
throwVetoException(encryptstatus1.getOutput())
```


Copyright © 2016, 2025, Oracle and/or its affiliates.

```
println(encryptstatus1.getOutput())

//Encrypt the password of a Service Administrator in the target environment
//This user must also have the Identity Domain Administrator
//role in the target environment

EpmAutomateStatus encryptstatus2= automate.execute('encrypt',
'targetpassword',
'encryptionKey', 'targetPassword.epw')
if(encryptstatus2.getStatus() != 0)
throwVetoException(encryptstatus2.getOutput())
println(encryptstatus2.getOutput())
```

Script pour le clonage d'un environnement

Ce script utilise les fichiers de mots de passe cryptés créés à l'aide du script précédent.

```
EpmAutomate automate = getEpmAutomate()

//Log into the target environment
EpmAutomateStatus loginstatus = automate.execute('login',
'username','targetPassword.epw' , 'targeturl')
if(loginstatus.getStatus() != 0) throwVetoException(loginstatus.getOutput())
println(loginstatus.getOutput())

//Recreate the target environment
EpmAutomateStatus recreatestatus = automate.execute('recreate' , '-f' )
if(recreatestatus.getStatus() != 0)
throwVetoException(recreatestatus.getOutput())
println(recreatestatus.getOutput())

//Copy Artifact Snapshot from the source environment
//to the target environment
EpmAutomateStatus copystatus = automate.execute('copysnapshotfrominstance',
'Artifact Snapshot', 'sourceusername', 'sourcePassword.epw','source url')
if(copystatus.getStatus() != 0) throwVetoException(copystatus.getOutput())
println(copystatus.getOutput())

//import Artifact Snapshot into the target environment
EpmAutomateStatus importstatus = automate.execute('importsnapshot', 'Artifact
Snapshot')
println(importstatus.getOutput())

//Send an email to a designated user with the status of the
//snapshot import process
EpmAutomateStatus emailstatus = automate.execute('sendmail',
'email_id' ,'Status of DR' , 'BODY='+ importstatus.getOutput())
println(emailstatus.getOutput())

//Sign out of the target environment
EpmAutomateStatus logoutstatus = automate.execute('logout')
println(logoutstatus.getOutput())
```


Copyright © 2016, 2025, Oracle and/or its affiliates.

Envoi par courriel du rapport d'activité à l'aide d'un script Groovy côté serveur

Ce script peut être utilisé pour envoyer le rapport d'activité par courriel à une liste de destinataires. Vous pouvez alors planifier l'exécution quotidienne du script pour obtenir un rapport d'activité journalier. Le script effectue les opérations suivantes :

- Il crypte le mot de passe de l'administrateur de service qui exécute le script Groovy.
- Il se connecte à l'environnement Oracle Fusion Cloud Enterprise Performance Management à l'aide du mot de passe crypté.
- Il envoie le rapport d'activité disponible dans l'environnement par courriel à une liste de destinataires, généralement les administrateurs de service.
- Il se déconnecte de l'environnement.

Si les mots de passe contiennent des caractères spéciaux, reportez-vous à la section Gestion des caractères spéciaux. Veillez également à remplacer ces valeurs de paramètre en fonction de vos environnements :

Table 4-2 Paramètres à modifier

Paramètre	Description
user	ID utilisateur d'un administrateur de service pour la connexion à l'environnement.
password	Mot de passe de l'administrateur de service.
url	URL de l'environnement Cloud EPM à partir duquel le rapport d'activité doit être envoyé par courriel.
emailaddresses	Liste des adresses électroniques auxquelles le rapport d'activité doit être envoyé, séparées par des points-virgules.

Pour obtenir des informations détaillées sur l'utilisation des règles Groovy, reportez-vous à la section Utilisation de règles Groovy du guide *Administration de Planning*.

```
/*RTPS: {user} {password} {url} {emailaddresses}*/

String user = 'service_administrator'
String password = 'examplePWD'
String url = 'example_EPM_URL'
String emailaddresses = 'service_administrator@oracle.com'

EpmAutomate automate = getEpmAutomate()

def LogMessage(String message) {
    def date = new Date()
    def sdf = new SimpleDateFormat("MM/dd/yyyy HH:mm:ss")
    println('[' + sdf.format(date) + '][GROOVY] ' + message);
}

def LogOperationStatus(EpmAutomateStatus opstatus) {
    def returncode = opstatus.getStatus()
    LogMessage(opstatus.getOutput())
    LogMessage('return code: ' + returncode)
```


Copyright © 2016, 2025, Oracle and/or its affiliates.

```
}

LogMessage('Starting mail activity report processing')

// encrypt
LogMessage("Operation: encrypt " + password + " oracleKey password.epw")
EpmAutomateStatus status =
automate.execute('encrypt',password,"oracleKey","password.epw")
LogOperationStatus(status)

// login
LogMessage("Operation: login " + user + " password.epw " + url)
status = automate.execute('login',user,"password.epw",url)
LogOperationStatus(status)

// listfiles
LogMessage('Operation: listfiles')
status = automate.execute('listfiles')
LogOperationStatus(status)

String filelist = status.getItemsList()
String[] str = filelist.split(',');
String reportfile = ''

for( String svalues : str ) {
    String[] ftr = svalues.split('/')
    for( String fvalues : ftr ) {
        if (fvalues.startsWith('2') && fvalues.endsWith('html')) {
            reportfile = fvalues
        }
    }
}

def reportdir = reportfile.tokenize(".")[0]
String reportpath = 'apr/' + reportdir + '/' + reportfile

// sendMail
LogMessage('Operation: sendMail ' + emailaddresses + ' Daily Activity Report
Body=Daily Activity Report Attachments=' + reportpath)
status = automate.execute('sendmail',emailaddresses,'Daily Activity
Report','Body=Daily Activity Report',"Attachments=${reportpath}")
LogOperationStatus(status)

// logout
LogMessage('Operation: logout')
status = automate.execute('logout')
LogOperationStatus(status)
```


Copyright © 2016, 2025, Oracle and/or its affiliates.

5

Réplication d'environnements Cloud EPM

Ces étapes interviennent dans la configuration d'un environnement Oracle Fusion Cloud Enterprise Performance Management secondaire pour assurer la disponibilité du service si la région principale n'est plus disponible à la suite de circonstances imprévues.

> **Remarque**
>
> Les procédures abordées dans cette annexe ne s'appliquent pas à Narrative Reporting.

- Configuration de la réplication d'artefact quotidienne
- Configuration de réplications à la demande
- Configuration de l'environnement secondaire

Configuration de la réplication quotidienne

Pour répliquer un environnement, vous utilisez EPM Automate afin de copier l'instantané d'artefact créé lors de la maintenance quotidienne de l'environnement principal vers l'environnement secondaire.

Oracle effectue quotidiennement une maintenance de routine sur chaque environnement. Lors de la maintenance de service, Oracle crée un instantané de maintenance en sauvegardant le contenu de l'environnement (données et artefacts existants, notamment les affectations d'utilisateur et de rôle à partir du domaine d'identité).

Pour configurer une réplication de service quotidienne, procédez comme suit :

1. Créez un fichier de script qui contient les commandes EPM Automate suivantes. Ce script réplique l'instantané d'application de l'environnement principal vers l'environnement secondaire.

   > **Remarque**
   >
   > Veillez à modifier le nom d'utilisateur, le fichier de mots de passe, les noms de domaine d'identité et les URL de service. Pour obtenir des informations sur la création d'un fichier de mots de passe cryptés, reportez-vous à la commande encrypt.

```
REM Sign in to the secondary instance
epmautomate login serviceAdmin secondaryPassword.epw secondary_URL
secondaryDomain
REM Delete the existing artifact snapshot
epmautomate deletefile "Artifact Snapshot"
REM Copy the snapshot from the primary instance
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
epmautomate copysnapshotfrominstance "Artifact Snapshot"
primaryPassword.epw primary_URL primaryDomain
REM Sign out of the secondary instance
epmautomate logout
```

2. A l'aide d'un planificateur, comme le planificateur de tâches Windows, planifiez l'exécution du fichier de script de sorte qu'il s'exécute deux heures après le début de la fenêtre de maintenance.
3. Définissez la même heure de début de la fenêtre de maintenance sur les environnements principal et secondaire. Pour plus d'informations, reportez-vous à la section Définition d'une heure de maintenance de service dans le guide *Guide de mise en route pour les administrateurs*.

Configuration de réplications à la demande

Pour réduire le RPO, vous pouvez créer des instantanés à la demande de l'environnement principal et les copier vers l'environnement secondaire.

Par exemple, vous pouvez créer et planifier un script EPM Automate qui s'exécute toutes les six heures entre les réplications quotidiennes pour faire passer le RPO de 24 à 6 heures.

> **Remarque**
>
> Lors de la création d'instantanés à la demande, l'environnement principal est placé en mode lecture seule pendant quelques minutes.

Pour configurer une réplication à la demande, procédez comme suit :

1. Créez un fichier de script qui contient les commandes EPM Automate suivantes. Ce script réplique l'instantané d'application de l'environnement principal vers l'environnement secondaire.

   > **Remarque**
   >
   > Veillez à modifier le nom d'utilisateur, le fichier de mots de passe, les noms de domaine d'identité et les URL de service. Pour obtenir des informations sur la création d'un fichier de mots de passe cryptés, reportez-vous à la commande encrypt.

```
REM Sign in to the primary instance
epmautomate login serviceAdmin primaryPassword.epw primary_URL
primaryDomain
REM Create a snapshot and then sign out
epmautomate exportsnapshot "Artifact Snapshot"
epmautomate logout
REM Sign in to the secondary instance
epmautomate login serviceAdmin secondaryPassword.epw secondary_URL
secondaryDomain
REM Copy the snapshot from the primary instance
epmautomate copysnapshotfrominstance "Artifact Snapshot"
primaryPassword.epw primary_URL primaryDomain
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

```
REM Sign out of the secondary instance
epmautomate logout
```

2. A l'aide d'un planificateur, comme le planificateur de tâches Windows, planifiez l'exécution du fichier de script à exécuter selon vos besoins afin d'atteindre le RPO souhaité.

Configuration de l'environnement secondaire

Configurez l'environnement secondaire pour l'activer.

Effectuez cette procédure uniquement si vous devez activer l'environnement secondaire lorsque l'environnement principal n'est pas disponible pendant une longue période. Avant de configurer l'environnement secondaire, reportez-vous aux rubriques ci-dessous dans le guide *Guide de mise en route pour les administrateurs* :

- Chemins de migration des instantanés EPM Cloud hérités
- Chemins de migration des instantanés EPM Standard Cloud Service et EPM Enterprise Cloud Service

Pour configurer l'environnement secondaire, procédez comme suit :

1. Démarrez une session d'EPM Automate et effectuez les activités ci-après.
   - Connectez-vous à l'environnement secondaire à l'aide d'un compte disposant à la fois du rôle Administrateur de service et du rôle Administrateur de domaine d'identité. Veillez à indiquer le nom d'utilisateur, le mot de passe, le nom de domaine et l'URL de service appropriés.
   - Recréez l'environnement secondaire.
     - Si l'environnement principal est un environnement Planning, Tax Reporting, Financial Consolidation and Close ou Enterprise Profitability and Cost Management, utilisez :
       `epmautomate recreate -f`
     - Si l'environnement principal n'est pas un environnement Planning, Tax Reporting, Financial Consolidation and Close ou Enterprise Profitability and Cost Management, utilisez :
       `epmautomate recreate -f TempServiceType=`*`PRIMARY_APPLICATION_TYPE`*, où *`PRIMARY_APPLICATION_TYPE`* est `ARCS`, `EDMCS`, `PCMCS` ou `EPRCS`.
   - Importez les artefacts d'application et de domaine d'identité à partir de l'instantané.
   - Déconnectez-vous.

   Vous pouvez effectuer les activités ci-dessous en exécutant les commandes suivantes. Reportez-vous aux rubriques suivantes :
   - Commande login
   - Commande recreate
   - Commande importSnapshot

   ```
   epmautomate login serviceAdmin secondaryPassword.epw secondary_URL
   epmautomate recreate –f
   epmautomate importsnapshot "Artifact Snapshot" importUsers=true
   epmautomate logout
   ```

2. Connectez-vous à l'environnement secondaire et vérifiez que toutes les données sont disponibles.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

3. Envoyez l'URL de l'environnement secondaire à tous les utilisateurs.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

A

Préparation de l'exécution de la commande simulateConcurrentUsage

La commande simulateConcurrentUsage prend en charge les opérations ci-après pour simuler la charge sur un environnement :

- Ouverture de formulaires
- Enregistrement de formulaires
- Exécution de règles métier
- Exécution d'ensembles de règles métier
- Exécution de règles de données
- Ouverture de grilles ad hoc
- Exécution de liasses de rapports de gestion
- Exécution de rapports de gestion

Etapes à réaliser

1. Créez le fichier `requirement.csv`. Reportez-vous à la section Création du fichier requirement.csv.
2. Créez des fichiers d'entrée indiquant les détails des opérations incluses dans `requirement.csv`. Reportez-vous aux sections suivantes :
   - Fichier d'entrée d'ouverture de formulaire
   - Fichier d'entrée d'enregistrement de formulaire
   - Fichier d'entrée d'exécution de règle métier
   - Fichier d'entrée d'exécution d'ensemble de règles métier
   - Fichier d'entrée d'exécution de règle de données
   - Fichier d'entrée de grille ad hoc
   - Fichier d'entrée d'exécution de liasse
   - Fichier d'entrée d'exécution de rapport
3. Créez un fichier `UserVarMemberMapping.csv` contenant les détails de variable utilisateur si le mapping de membre de variable utilisateur doit être défini. Reportez-vous à la section Création du fichier UserVarMemberMapping.csv.
4. Exportez éventuellement les options Oracle Smart View for Office dans le fichier `options.xml`, si les options Smart View doivent être utilisées. Pour plus de détails, reportez-vous à la section Création du fichier options.xml.
5. Créez éventuellement le fichier `users.csv` pour exécuter des simulations à l'aide d'utilisateurs existants. Pour plus de détails, reportez-vous à la section Création du fichier users.csv.

Copyright © 2016, 2025, Oracle and/or its affiliates.

6. Créez un fichier ZIP contenant les fichiers créés lors des étapes précédentes et chargez-le dans l'environnement. Reportez-vous à la section Création du fichier d'entrée ZIP et chargement dans l'environnement.
7. Exécutez la commande simulateConcurrentUsage en utilisant le fichier ZIP chargé.

Création du fichier requirement.csv

Commencez par créer le fichier `requirement.csv`, qui répertorie les détails des cas d'emploi à tester. Chaque ligne du fichier CSV identifie le type d'opération à effectuer, le nom d'artefact, le nombre d'utilisateurs simultanés, le fichier d'entrée indiquant les détails de l'opération et d'autres informations relatives à l'opération, le cas échéant. Par exemple, pour ouvrir 2 formulaires, enregistrer 2 formulaires et exécuter 2 règles métier, vous indiquez 6 lignes dans le fichier d'entrée CSV. La première ligne du fichier `requirement.csv` (en-tête) doit contenir les informations suivantes :

```
#Type of Operation,Artifact Name,Number of Users,Input File,Additional Info
```

Chacune des lignes suivantes du fichier contient une seule opération et les paramètres correspondants. Certaines opérations ne requièrent pas toutes ces valeurs de paramètre. Les entrées de fichier attendues sont expliquées dans le tableau ci-après.

> **Note**
>
> Toutes les valeurs sont requises, sauf indication contraire dans le tableau.

Table A-1 Format du fichier requirement.csv

Champ	Description
Type d'opération	Il doit s'agir de l'un des éléments suivants : • `Open Form` • `Save Form` • `Run Business Rule` • `Run Business Ruleset` • `Run Data Rule` • `Ad Hoc Grid` • `Execute Report` • `Execute Book`
Nom de l'artefact	Cette valeur dépend du type d'opération : • Ouvrir un formulaire : nom et emplacement du formulaire à ouvrir. • Enregistrer un formulaire : nom et emplacement du formulaire à enregistrer. • Exécuter une règle métier : nom de la règle métier. • Exécuter l'ensemble de règles métier : nom de l'ensemble de règles métier. • Règle de données : nom de la règle de données. • Grille ad hoc : non applicable (ne pas renseigner). • Exécuter un rapport : nom et emplacement du rapport. • Exécuter une liasse : nom et emplacement de la liasse.
Nombre d'utilisateurs	Nombre d'utilisateurs pour simuler l'utilisation simultanée.

Copyright © 2016, 2025, Oracle and/or its affiliates.

Table A-1 (Cont.) Format du fichier requirement.csv

Champ	Description
Fichier d'entrée	Nom du fichier CSV qui contient les valeurs de PDV, les invites d'exécution ou d'autres valeurs propres au cas d'emploi à utiliser.
Informations supplémentaires	Autres paramètres requis pour l'opération. Applicable aux grilles ad hoc uniquement. Laissez le champ vide pour les autres cas d'emploi.

Remarques : les noms d'artefact doivent correspondre à ceux de l'application (casse comprise).

Démarrez une ligne avec le signe dièse (#) pour la commenter. Cependant, la première ligne du fichier est l'en-tête et doit commencer par #.

Exemple de fichier `requirement.csv` :

```
# Type of Operation,Artifact Name,Number of Users,Input File,Additional Info
Open Form,Library/Global Assumption/Revenue Forecast
Assumptions,10,openform_input.csv
Save Form,Library/Global Assumption/ExchangeRates,5,saveform_input.csv
Run Business Rule,Run_FinStatement - Copy Budget to Prior Year
Budget,4,runbusinessrule_input.csv
Run Business Ruleset,RollupUSSales,5,runbusinessruleset_input.csv
Run Data Rule,Delimited_file_DL,5,rundatarule_input.csv
Ad Hoc Grid,,3,runadhocgrid_input.csv,cube=FinStmt
# Execute Book,Review Books/Revenue Reports,10,book_input.csv
# Execute Report,Review Reports/Executive Report,10,report_input.csv
```

Création des fichiers d'entrée

Chaque cas d'emploi identifié dans `requirement.csv` doit disposer d'un fichier d'entrée correspondant fournissant tous les paramètres requis pour l'exécuter.

Dans l'idéal, le fichier d'entrée doit contenir autant d'entrées que d'utilisateurs indiqués pour le cas d'emploi correspondant dans `requirement.csv`.

Si le nombre d'entrées du fichier d'entrée est inférieur au nombre d'utilisateurs simultanés du cas d'emploi dans `requirement.csv`, EPM Automate répète certaines entrées pour exécuter les cas d'emploi jusqu'à ce que l'opération ait été exécutée pour le nombre d'utilisateurs indiqué dans `requirement.csv`.

Par exemple, si l'entrée de cas d'emploi dans `requirement.csv` pour l'opération d'exécution d'une règle métier est la suivante :

```
Run Business Rule, Copy Budget,10,br_input_file.csv,
```

`br_input_file.csv` doit également contenir dix entrées. Si `br_input_file.csv` contient seulement six entrées, EPM Automate les utilise pour les six premiers utilisateurs. Pour les quatre utilisateurs suivants, EPM Automate réutilise les quatre premières entrées de `br_input_file.csv`.

Si le nombre d'entrées du fichier d'entrée est supérieur au nombre d'utilisateurs indiqué pour le cas d'emploi, EPM Automate ignore les dernières entrées.

- Fichier d'entrée d'ouverture de formulaire
- Fichier d'entrée d'enregistrement de formulaire

Copyright © 2016, 2025, Oracle and/or its affiliates.

- Fichier d'entrée d'exécution de règle métier
- Fichier d'entrée d'exécution d'ensemble de règles métier
- Fichier d'entrée d'exécution de règle de données
- Fichier d'entrée de grille ad hoc
- Fichier d'entrée d'exécution de rapport
- Fichier d'entrée d'exécution de liasse

Fichier d'entrée d'ouverture de formulaire

Ce fichier, référencé dans `requirement.csv` pour la prise en charge de l'ouverture des formulaires, inclut des entrées de PDV au format `pov=[DIM 1:MEMBER 1],[DIM 2:MEMBER 2], [DIM 3:MEMBER 3]`.

Dans cette discussion, DIM 1, DIM 2, DIM 3 sont les noms de dimension, et MEMBER 1, MEMBER 2, MEMBER 3 les valeurs de membre de dimension correspondantes pour le PDV.

Exemple de fichier d'entrée :

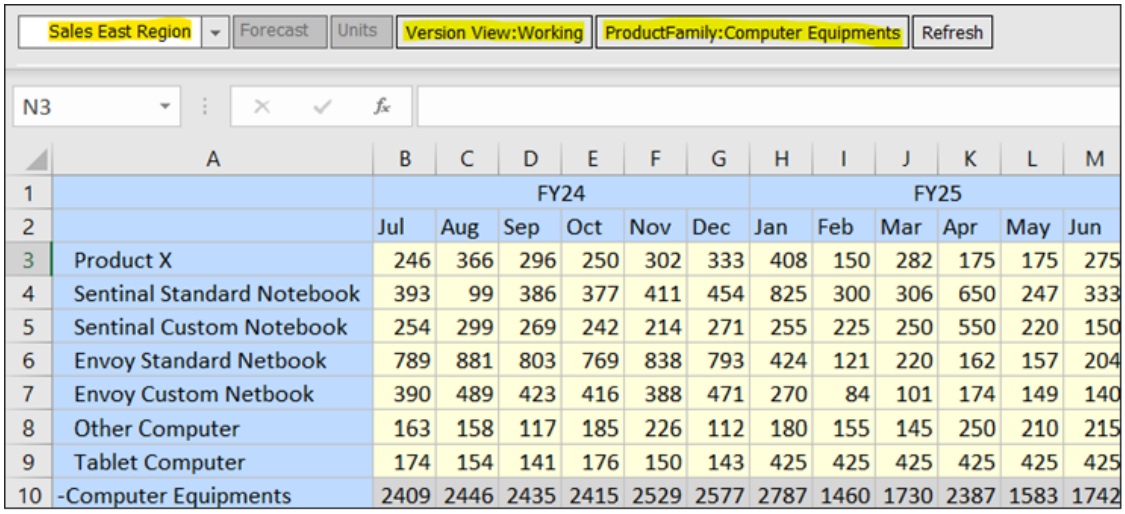
```
pov=[Account:APL_RATE_AED],[Scenario:Budget],[Years:FY20]
pov=[Account:APL_RATE_AED],[Scenario:Budget],[Years:FY19]
pov=[Account:APL_RATE_AED],[Scenario:Budget],[Years:FY18]
pov=[Account:APL_RATE_AED],[Scenario:Budget],[Years:FY17]
pov=[Account:APL_RATE_AED],[Scenario:Budget],[Years:FY16]
```

> **Note**
>
> Vous devez également créer `UserVarMemberMapping.csv` si les formulaires indiqués dans `requirement.csv` requièrent la définition de variables utilisateur. Reportez-vous à la section Création du fichier UserVarMemberMapping.csv.

Instruction

La balise `pov` dans le fichier d'entrée doit contenir uniquement les dimensions et les variables utilisateur pouvant être sélectionnées. Dans l'exemple suivant, comme Forecast et Units ne sont pas sélectionnables, vous devez les exclure de la balise `pov`. Par exemple, `pov=[Entity:Sales East Region],[Version View:Working],[ProductFamily:Computer Equipments];`

Copyright © 2016, 2025, Oracle and/or its affiliates.



Sales East Region | Forecast | Units | Version View:Working | ProductFamily:Computer Equipments | Refresh

N3

	A	B	C	D	E	F	G	H	I	J	K	L	M
1		FY24						FY25					
2		Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun
3	Product X	246	366	296	250	302	333	408	150	282	175	175	275
4	Sentinal Standard Notebook	393	99	386	377	411	454	825	300	306	650	247	333
5	Sentinal Custom Notebook	254	299	269	242	214	271	255	225	250	550	220	150
6	Envoy Standard Netbook	789	881	803	769	838	793	424	121	220	162	157	204
7	Envoy Custom Netbook	390	489	423	416	388	471	270	84	101	174	149	140
8	Other Computer	163	158	117	185	226	112	180	155	145	250	210	215
9	Tablet Computer	174	154	141	176	150	143	425	425	425	425	425	425
10	-Computer Equipments	2409	2446	2435	2415	2529	2577	2787	1460	1730	2387	1583	1742

.

Fichier d'entrée d'enregistrement de formulaire

Ce fichier, référencé dans `requirement.csv` pour la prise en charge de l'enregistrement des formulaires, inclut le PDV et des valeurs d'entrée de cellule au format suivant. Si des règles métier sont associées au formulaire, ce fichier inclut également les invites d'exécution de ces règles métier .

Exemple de fichier d'entrée :

```
pov=[DIM 1:MEMBER1],[DIM 2:MEMBER2],[DIM 3:MEMBER3],…;cells=[CELL COLUMN
HEADER 1 -> CELL COLUMN HEADER 2 -> CELL COLUMN HEADER 3 ->.. | CELL ROW
HEADER 1-> CELL ROW HEADER 2-> CELL ROW HEADER 3->..| CELL 1 DATA], [CELL
COLUMN HEADER 11 -> CELL COLUMN HEADER 22 -> CELL COLUMN HEADER 33 ->.. |
CELL ROW HEADER 11-> CELL ROW HEADER 22-> CELL ROW HEADER 33->..| CELL 2
DATA];rtp=[BUSINESS RULE NAME1[RTP1:VALUE1][RTP2:VALUE2]],[BUSINESS
RULE2[RTP3:VALUE3]]..
```

Dans cet exemple :

- DIM indique le nom d'une dimension et MEMBER une valeur de membre de dimension.
- CELL COLUMN HEADER identifie le nom de l'en-tête de colonne et CELL ROW HEADER celui d'un en-tête de ligne.
- BUSINESS RULE NAME indique le nom de la règle métier, RTP indique le nom de l'invite d'exécution et VALUE indique sa valeur. RTP n'est pas nécessaire si aucune règle métier n'est associée au formulaire ou si vous voulez utiliser la valeur par défaut de l'invite d'exécution présente dans le formulaire.

Par exemple :

```
pov=[Version View:Working],[Sales Entity:International Sales];cells=[FY16-
>x---------x->Pct|P293:Maintenance->4120: Support|1];rtp=[Services Revenue -
Forecast[Department:000][Scenario:Plan]],[Allocate Plan
Targets[TargetVersion:Baseline]]
```

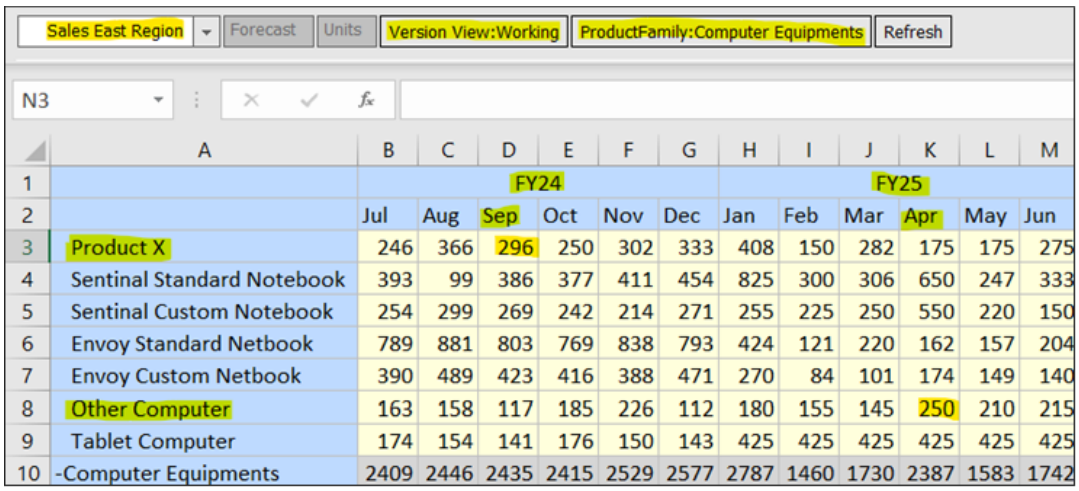

Copyright © 2016, 2025, Oracle and/or its affiliates.

> **Note**
>
> Vous devez également créer `UserVarMemberMapping.csv` si les formulaires indiqués dans `requirement.csv` requièrent la définition de variables utilisateur. Reportez-vous à la section Création du fichier UserVarMemberMapping.csv.

Si des transmissions dynamiques sont associées au formulaire, elles sont exécutées automatiquement.

Instructions

1. La balise `pov` dans le fichier d'entrée doit contenir uniquement les dimensions et les variables utilisateur pouvant être sélectionnées. Dans l'exemple suivant, comme Forecast et Units ne sont pas sélectionnables, vous devez les exclure de la balise `pov`. Par exemple, `pov=[Entity:Sales East Region],[Version View:Working], [ProductFamily:Computer Equipments];`



	A	B	C	D	E	F	G	H	I	J	K	L	M
1		FY24						FY25					
2		Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun
3	Product X	246	366	296	250	302	333	408	150	282	175	175	275
4	Sentinal Standard Notebook	393	99	386	377	411	454	825	300	306	650	247	333
5	Sentinal Custom Notebook	254	299	269	242	214	271	255	225	250	550	220	150
6	Envoy Standard Netbook	789	881	803	769	838	793	424	121	220	162	157	204
7	Envoy Custom Netbook	390	489	423	416	388	471	270	84	101	174	149	140
8	Other Computer	163	158	117	185	226	112	180	155	145	250	210	215
9	Tablet Computer	174	154	141	176	150	143	425	425	425	425	425	425
10	-Computer Equipments	2409	2446	2435	2415	2529	2577	2787	1460	1730	2387	1583	1742

.

2. Ouvrez le formulaire dans Oracle Smart View for Office et copiez les en-têtes de colonne et de ligne de cellule dans la balise `cells`. Dans Smart View, si des lignes ou des colonnes masquées sont présentes dans le formulaire, affichez-les avant de copier les en-têtes. Notez tous les en-têtes de colonne et de ligne, y compris les valeurs des lignes et des colonnes masquées, pour atteindre la cellule. Dans l'exemple ci-dessus, si les cellules cible sont D3 et K8, la balise cells doit indiquer `cells=[FY24->Sep|Product X|296], [FY25->Apr|Other Computer|250];`.

> **Note**
>
> La définition de cellule est facultative. Si la définition de cellule n'est pas incluse, la commande enregistre le formulaire dans le PDV indiqué.

3. La balise `rpt` n'est pas requise dans les situations suivantes :
   - Aucune règle métier n'est associée au formulaire.
   - Vous voulez utiliser les valeurs d'invite d'exécution par défaut présentes dans le formulaire.

Copyright © 2016, 2025, Oracle and/or its affiliates.

Fichier d'entrée d'exécution de règle métier

Ce fichier, référencé dans `requirement.csv` pour la prise en charge de l'exécution des règles métier, inclut des valeurs de paramètre d'exécution au format `rtp=[PARAMETER:VALUE], [PARAMETER:VALUE]` et ainsi de suite. Dans ce format, *PARAMETER* identifie un nom d'invite d'exécution et *VALUE* indique sa valeur.

Veillez à ajouter autant d'invites d'exécution que nécessaire pour exécuter la règle. Les valeurs par défaut sont utilisées si vous n'indiquez pas les invites d'exécution et leurs valeurs. La commande ignore les invites d'exécution qui ne sont pas des correspondances exactes de celles définies pour la règle. Si aucun paramètre d'exécution n'est requis pour la règle métier, incluez `rtp=[]`.

Exemple de fichier d'entrée :

```
rtp=[Period:Q1],[Entity:USA]
rtp=[Period:Q2],[Entity:USA]
rtp=[Period:Q3],[Entity:USA]
rtp=[Period:Q4],[Entity:USA]
```

Fichier d'entrée d'exécution d'ensemble de règles métier

Ce fichier, référencé dans `requirement.csv` pour la prise en charge de l'exécution des ensembles de règles métier, inclut des valeurs de paramètre d'exécution au format `rtp=[PARAMETER:VALUE],[PARAMETER:VALUE]` et ainsi de suite. Dans ce format, *PARAMETER:VALUE* identifie un nom de paramètre d'exécution et sa valeur.

Utilisez la paire *PARAMETER:VALUE* pour indiquer autant d'invites d'exécution que l'ensemble de règles l'exige. Les valeurs par défaut sont utilisées si vous n'indiquez aucune valeur pour les paramètres d'exécution. La commande ignore les invites d'exécution qui ne sont pas des correspondances exactes avec celles définies pour l'ensemble de règles. Si aucune valeur de paramètre d'exécution n'est requise pour l'ensemble de règles métier, incluez `rtp=[]`.

Exemple de fichier d'entrée :

```
rpt=[Period:Q1],[Entity:USA]
rtp=[Period:Q2],[Entity:USA]
rtp=[Period:Q3],[Entity:USA]
rtp=[Period:Q4],[Entity:USA]
```

Fichier d'entrée d'exécution de règle de données

Ce fichier, référencé dans `requirement.csv` pour la prise en charge de l'exécution des règles de données, doit indiquer la période de début, la période de fin, le mode d'import, le mode d'export et un nom de fichier d'import facultatif disponible dans l'environnement. Si aucun nom de fichier n'est indiqué, celui indiqué dans la règle de données est utilisé. Format de chaque ligne : `startperiod=START PERIOD;endperiod=END PERIOD;importmode=IMPORT_MODE;exportmode=EXPORT_MODE;filename=FILE NAME`

Exemple de fichier d'entrée :

```
startperiod=Dec-15;endperiod=Dec-15;importmode=REPLACE;exportmode=STORE_DATA;f
ilename=comma_delim_file1.csv
```


Copyright © 2016, 2025, Oracle and/or its affiliates.

```
startperiod=Dec-16;endperiod=Dec-16;importmode=REPLACE;exportmode=STORE_DATA;f
ilename=comma_delim_file2.csv
startperiod=Dec-17;endperiod=Dec-17;importmode=REPLACE;exportmode=STORE_DATA;f
ilename=comma_delim_file3.csv
startperiod=Dec-18;endperiod=Dec-18;importmode=REPLACE;exportmode=STORE_DATA;f
ilename=comma_delim_file4.csv
startperiod=Dec-19;endperiod=Dec-19;importmode=REPLACE;exportmode=STORE_DATA;f
ilename=comma_delim_file5.csv
```

Fichier d'entrée de grille ad hoc

La commande simulateConcurrentUsage prend en charge les grilles ad hoc en mode natif et en mode standard. Dans les grilles en mode natif, les PDV sont affichés dans la barre d'outils du plug-in Oracle Smart View for Office. En mode standard, les PDV font partie de la feuille de calcul elle-même et occupent la première ligne de la feuille de calcul.

Le fichier d'entrée de la grille ad hoc, référencé dans `requirement.csv` pour la prise en charge de l'ouverture de grilles ad hoc, doit indiquer la grille à ouvrir. Chaque ligne du fichier doit être au format suivant.

```
filename=xlsx filename#sheet name; pov=[DIM 1:MEMBER 1],[DIM 2:MEMBER 2]..;
rows=[ROW HEADER 1, ROW HEADER 2,..]; cols=[COL HEADER 1, COL HEADER 2,..]
```

Exemple de fichier d'entrée :

```
fileName=dropdown.xlsx#sheet4;pov=[HSP_View:BaseData],[Scenario:Forecast],
[Product:No Product],[Entity:Sales Mid-Atlantic];rows = [ Account ]; cols=
[Year, Period, Version]
```

> **Note**
>
> Le fichier identifié par `fileName` dans le fichier d'entrée CSV doit contenir la définition de grille ad hoc dans la feuille indiquée. Par exemple, l'exemple de fichier d'entrée précédent indique `sheet4` dans `dropdown.xlsx` comme source de la définition de grille ad hoc. Comme le fichier `requirement.csv` et le fichier d'entrée CSV, ce fichier Excel doit être disponible dans le fichier *INPUT_FILE.zip* utilisé pour exécuter la commande simulateConcurrentUsage.

Fichier d'entrée d'exécution de rapport

Ce fichier, référencé dans `requirement.csv` pour la prise en charge de l'ouverture des rapports de gestion, doit indiquer le rapport à ouvrir. Chaque ligne du fichier doit être au format suivant.

```
format=[REPORT_FORMAT];globalPov=[DIM 1:MEMBER 1],[DIM 2:MEMBER
2],..;prompts=[PROMPT 1:VALUE 1],[PROMPT 2:VALUE 2],..
```

`globalPov` et `prompts` sont facultatifs.

Copyright © 2016, 2025, Oracle and/or its affiliates.

> **Note**
>
> - Les formats de rapport pris en charge sont `pdf` et `embedded`.
> - Si le nom de la dimension globalPov ou de l'un de ses membres contient un signe deux-points (:) ou un point-virgule (;), faites précéder ce dernier du caractère d'échappement \\. Par exemple, le nom de dimension `Version:View` doit être indiqué comme suit : `Version\\:View`.

Exemple de fichier d'entrée permettant de générer un fichier PDF à partir d'un rapport avec le PDV global `[Version View:Working],[Sales Entity:International Sales]` et les invites `[Actual;Budget],[Year:2018]` :

```
format=pdf;globalPov=[Version View:Working],[Sales Entity:International
Sales];prompts=[Actual:Budget],[Year:2018]
```

Fichier d'entrée d'exécution de liasse

Ce fichier, référencé dans `requirement.csv` pour la prise en charge de l'ouverture des liasses dans Reports, doit indiquer la liasse à ouvrir. Chaque ligne du fichier doit être au format suivant.

`format=BOOK_FORMAT` ou

`format=BOOK_FORMAT;globalPov=[DIM 1:MEMBER 1],[DIM 2:MEMBER 2],..`

Exemple de fichier d'entrée permettant de générer un fichier PDF à partir d'une liasse avec le PDV global *[Version View:Working],[Sales Entity:International Sales]* :

```
format=pdf;globalPov=[Version View:Working],[Sales Entity:International Sales]
```

> **Note**
>
> - Les formats de liasse pris en charge sont PDF et XLSX.
> - Si le nom de la dimension globalPov ou de l'un de ses membres contient un signe deux-points (:) ou un point-virgule (;), faites précéder ce dernier du caractère d'échappement \\. Par exemple, le nom de dimension `Version:View` doit être indiqué comme suit : `Version\\:View`.

Création du fichier UserVarMemberMapping.csv

Ce ficher est requis si les formulaires indiqués dans le fichier d'entrée des cas d'emploi d'ouverture ou d'enregistrement de formulaire requièrent la définition de variables utilisateur. Ce fichier n'est pas nécessaire pour les autres cas d'emploi.

- En mode test 0 et 1, ce fichier est utilisé afin de définir les mappings de membre de variable utilisateur pour les utilisateurs de test créés.
- En mode test 4, ce fichier est utilisé afin de définir les mappings de membre de variable utilisateur pour les utilisateurs existants.

Copyright © 2016, 2025, Oracle and/or its affiliates.

La première ligne de ce fichier est l'en-tête `#Dimension,User Variable,Member`. Les lignes suivantes contiennent le mapping de dimension, de variable utilisateur et de membre de dimension.

Exemple de fichier `UserVarMemberMapping.csv` :

```
#Dimension,User Variable,Member
Account,Account View,Revenue Driver Assumptions
Entity,Entity,No Entity
Entity,Entity View,Total Entity
HSP_View,HSP_View,BaseData
Market Size,Market View,Large Market
Period,Period,Jan
```

Création du fichier options.xml

Les options Oracle Smart View for Office telles que la suppression de ligne, la suppression de bloc manquant et la mise en retrait de membre de page peuvent être utilisées lors de la simulation des cas d'emploi d'ouverture de formulaire, d'enregistrement de formulaire et de grille ad hoc. Chacune de ces options exerce des niveaux de charge différents sur l'environnement.

Si vous voulez simuler le chargement à l'aide des options Smart View, vous pouvez inclure le fichier `options.xml` dans le fichier ZIP d'entrée.

Le fichier `options.xml` est créé en exportant les options Smart View. Pour générer le fichier `options.xml`, cliquez sur **Options** dans l'onglet **SmartView**, puis sur **OK**, et choisissez **Exporter les options**.

L'utilisation du fichier `options.xml` n'est pas obligatoire. Si ce fichier n'est pas présent dans le fichier ZIP d'entrée, les options par défaut sont utilisées pour la simulation.

Création du fichier users.csv

Le mode 4 de la commande simulateConcurrentUsage vous permet d'exécuter la commande à l'aide des utilisateurs existants identifiés dans le fichier `users.csv` inclus dans le fichier ZIP d'entrée. Ce mode ne crée pas d'utilisateurs pour la simulation.

Le format du fichier `users.csv` est `loginname,password` :

```
jdoe,jdoe_pwd
john.doe@example.com,john_doe_pwd
```

La valeur de `loginname` dans `users.csv` doit correspondre à un nom de connexion défini dans le domaine d'identité. De plus, l'utilisateur identifié doit disposer des rôles prédéfinis et des rôles d'application nécessaires pour exécuter l'opération.

Création du fichier d'entrée ZIP et chargement dans l'environnement

A l'aide d'un outil tel que 7 Zip, créez un fichier ZIP contenant `requirement.csv`, les fichiers d'entrée de cas d'emploi correspondants et, éventuellement, `UserVarMemberMapping.csv`, `users.csv` et `options.xml`, si nécessaire.

Copyright © 2016, 2025, Oracle and/or its affiliates.

Utilisez la commande uploadFile pour charger le fichier ZIP généré dans la boîte de réception de l'environnement (exemple de syntaxe de commande : `epmautomate uploadFile "C:/uploads/INPUT_FILE.zip" inbox`) dans lequel exécuter la simulation.

Exemple de rapport de simulation d'utilisation simultanée

Par défaut, le rapport de simulation d'utilisation simultanée est envoyé à l'utilisateur qui exécute la commande simulateConcurrentUsage. Si vous indiquez des destinataires de courriel, le rapport n'est envoyé qu'à ces derniers.

Simulate Concurrent Usage report

Operation #	Operation	Artifact Name	Users	Iterations	Min. Duration	Max. Duration	Avg. Duration	Return Status
1	Open Form	Set Services Revenue Forecast Assumptions	2	1	00:00:00.17	00:00:00.21	00:00:00.19	Passed
2	Save Form	Set Headcount and Salary Forecast Assumptions	1	1	00:00:00.45	00:00:00.45	00:00:00.45	Passed
3	Run Business Rule	Operating Expense Adj Plan	2	1	00:00:00.60	00:00:00.61	00:00:00.60	Passed
4	Run Data Rule	test	2	1	00:00:03.00	00:00:03.29	00:00:03.14	Passed
5	Ad Hoc Grid		1	1	00:00:00.13	00:00:00.13	00:00:00.13	Passed
6	Execute Book	Book1	1	1	00:00:10.21	00:00:10.21	00:00:10.21	Passed
7	Execute Report	Variance Explanations1	1	1	00:00:04.95	00:00:04.95	00:00:04.95	Passed
8	Save Form	Set Services Revenue Forecast Assumptions	1	1	00:00:05.56	00:00:05.56	00:00:05.56	Passed
9	Run Business Rule	Allocate Plan Targets	2	1	00:00:02.62	00:00:02.62	00:00:02.62	Passed
10	Run Business Ruleset	Revenue Plan	2	1	00:00:00.60	00:00:00.61	00:00:00.61	Passed

Ce rapport identifie les éléments suivants :

Colonne	Description
Numéro d'opération	Numéro de séquence du cas d'emploi dans `requirement.csv`.
Opération	Type d'opération tel qu'indiqué dans `requirement.csv`.
Nom de l'artefact	Nom de l'artefact tel qu'indiqué dans `requirement.csv`.
Utilisateurs	Nombre d'utilisateurs tel qu'indiqué dans `requirement.csv`.
Itérations	Nombre de fois où le cas d'emploi a été exécuté, tel qu'indiqué par le paramètre `iterations`.
Durée min.	Durée minimale d'exécution du cas d'emploi par un utilisateur.
Durée max.	Durée maximale d'exécution du cas d'emploi par un utilisateur.
Durée moy.	Durée moyenne d'exécution du cas d'emploi par un utilisateur.
Statut renvoyé	Statut du cas d'emploi. `Failed` apparaît si l'exécution du cas d'emploi a échoué.

Copyright © 2016, 2025, Oracle and/or its affiliates.

B

Préparation de l'exécution de la commande replay

La commande replay permet d'effectuer un test de performances sur un environnement dans des conditions de charge afin de vérifier que l'expérience utilisateur est acceptable lorsque le service est soumis à la charge indiquée. Vous devez effectuer quelques étapes avant de tester les environnements dans des conditions de charge.

Cette annexe décrit les étapes que les administrateurs de service doivent effectuer avant d'exécuter la commande replay d'EPM Automate.

- A propos de la commande replay
- Prérequis
- Création de fichiers HAR
- Création de fichiers de réexécution
- Génération de fichiers de trace
- Exemple de session de réexécution

A propos de la commande replay

La commande replay réexécute le chargement d'Oracle Smart View for Office, de l'API REST Oracle Fusion Cloud Enterprise Performance Management ou d'EPM Automate dans un environnement afin de permettre le test des performances dans des conditions de charge élevée, pour vérifier que l'expérience utilisateur est acceptable lorsque l'environnement est soumis à la charge indiquée.

Par exemple, vous pouvez tester l'expérience utilisateur sur un environnement de test dans des conditions de charge élevée pour vous assurer qu'elle fonctionne bien après que vous avez migré l'application de l'environnement de test vers l'environnement de production.

Prérequis

Lorsque vous exécutez la commande à l'aide d'un fichier de réexécution, EPM Automate exécute chaque ligne dans le fichier de réexécution en parallèle pour exercer une charge sur le service, de sorte que vous puissiez effectuer des tests afin de vérifier que l'expérience utilisateur est acceptable lorsque le service est soumis à une charge.

- Identifiez les formulaires qui exigent un important traitement sur l'environnement. Les formulaires qui gèrent d'importants volumes de données ou les formulaires qui incluent des calculs complexes sont de bons candidats. Par exemple, des formulaires utilisés pour soumettre des prévisions, des processus impliqués dans la création de rapports ad hoc et statiques, peuvent exercer une forte charge sur le service. De même, les activités telles que l'exécution de règles métier, de rapports, d'API REST nécessitant beaucoup de ressources, et de commandes EPM Automate (par exemple, runBusinessRule, runDataRule, exportData, exportMetadata, restructureCube) peuvent exercer une forte charge sur l'environnement et peuvent être soumises aux tests de charge.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

- Installez Fiddler si besoin. EPM Automate exige un fichier HAR (format HTTP Archive) 1.1 contenant les enregistrements des interactions d'Oracle Smart View for Office, de l'API REST Oracle Fusion Cloud Enterprise Performance Management ou d'EPM Automate avec l'environnement Cloud EPM. Fiddler permet généralement de générer le fichier HAR qui capture le journal de vos interactions avec Cloud EPM.
- Exécutez les principales activités que vous avez identifiées précédemment. Vous utilisez Smart View pour exécuter les activités telles que l'ouverture et l'enregistrement de formulaires, l'exécution de règles métier et la création de rapports, et Fiddler pour capturer les détails d'activité et les exporter vers des fichiers HAR. De même, exécutez des API REST et des commandes EPM Automate, et utilisez Fiddler pour capturer les détails. Pour plus de détails, reportez-vous à la section Création de fichiers HAR.
- Créez un fichier CSV de réexécution qui répertorie les informations d'identification (noms d'utilisateur et mots de passe) et le nom des fichiers HAR à exécuter. Chaque ligne du fichier peut contenir le nom d'utilisateur et le mot de passe d'un utilisateur unique pour simuler plusieurs sessions utilisateur simultanées. Pour plus de détails, reportez-vous à la section Création de fichiers de réexécution.
  Il n'est pas nécessaire que l'utilisateur dont les informations d'identification sont indiquées dans une ligne pour exécuter un fichier HAR soit l'utilisateur qui a exécuté la session utilisée pour créer le fichier HAR. Cependant, cet utilisateur doit disposer des droits permettant d'exécuter ces activités sur l'environnement.

Reportez-vous à la section Exemple de session de réexécution pour connaître les étapes détaillées d'exécution de la commande replay.

Création de fichiers HAR

Le fichier HAR capture les traces des interactions d'Oracle Smart View for Office, de l'API REST ou d'EPM Automate avec Oracle Fusion Cloud Enterprise Performance Management.

Comme Fiddler capture des informations sur tous les trafics HTTP(S), évitez les activités qui peuvent ajouter des traces inutiles à Fiddler lors de la création des fichiers HAR.

Pour créer un fichier HAR, procédez comme suit :

1. Démarrez Fiddler.
2. Assurez-vous que Fiddler est configuré pour décrypter le trafic HTTPS de tous les processus.
   a. Sélectionnez **Tools**, **Options**, puis **HTTPS**.
   b. Sélectionnez **Decrypt HTTPS traffic**, si ce n'est pas déjà le cas.

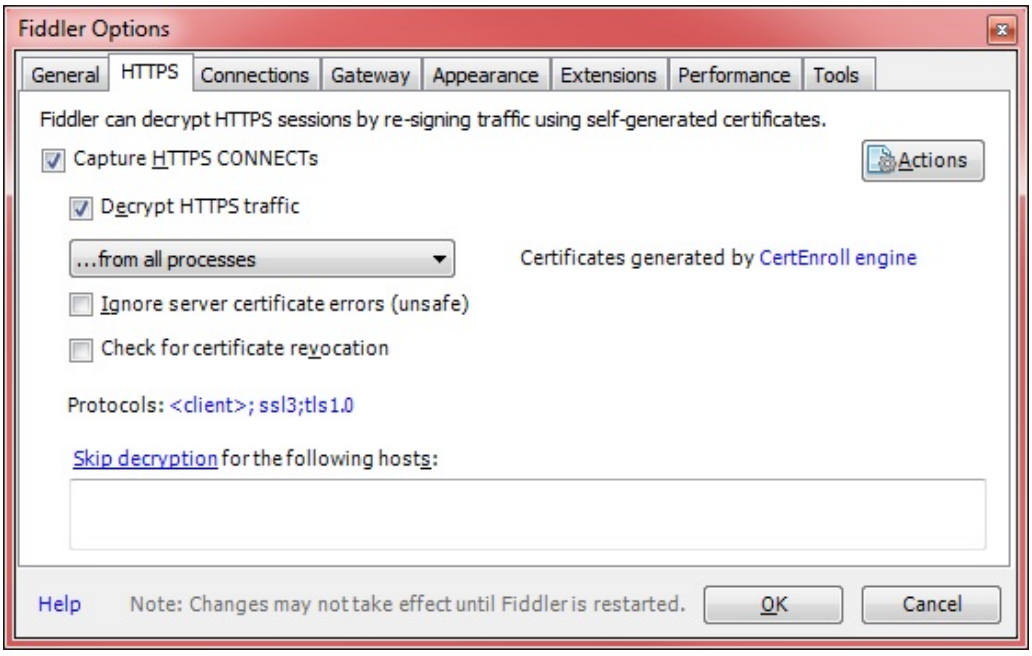
      Fiddler affiche les informations sur le certificat racine qu'il utilise pour intercepter le trafic. Ce certificat est généralement sûr.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

c. Cliquez sur **Yes** si vous voulez ajouter le certificat racine à la liste d'organismes de certification sécurisés ; sinon, choisissez **No**.

d. **Facultatif :** si vous avez sélectionné **No** à l'étape précédente, vous pouvez sélectionner **Ignore server certificate errors** pour supprimer les avertissements de sécurité Fiddler liés au décryptage du trafic HTTPS.

e. Cliquez sur **OK**.

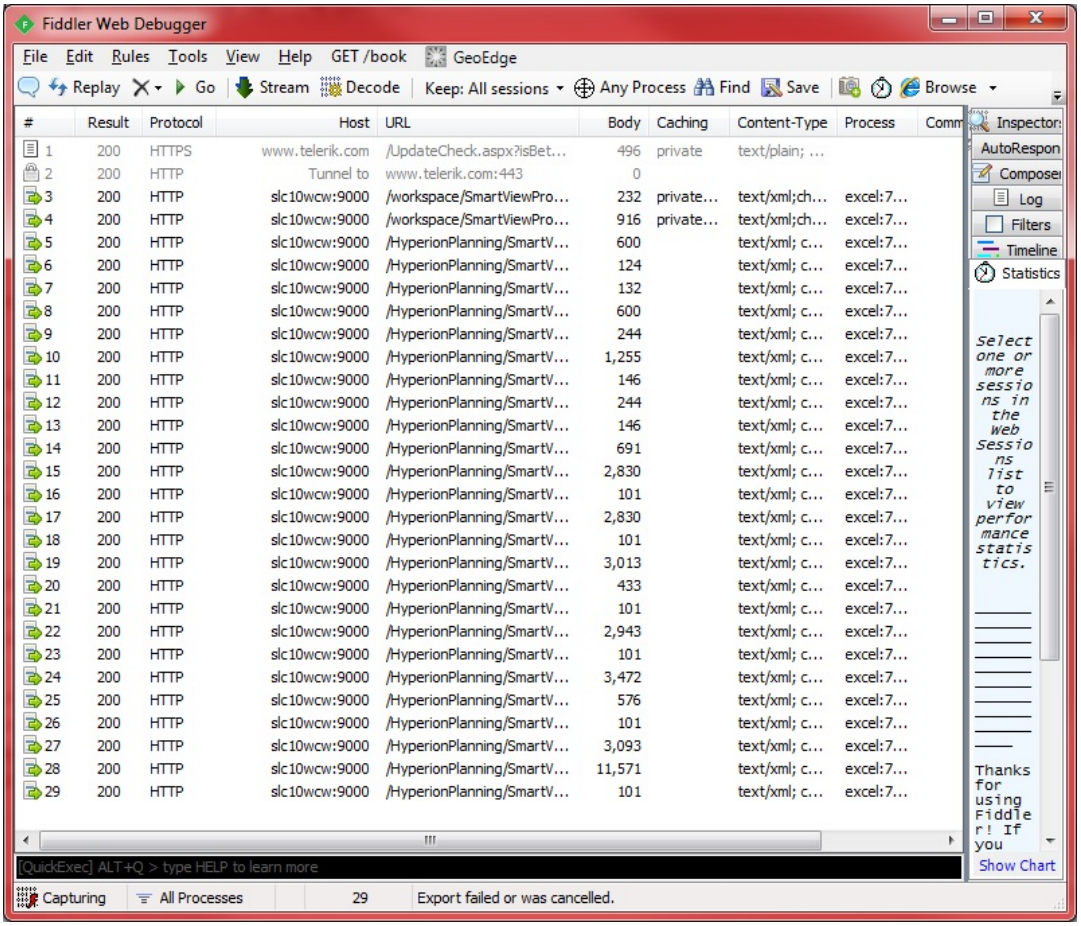


3. Démarrez Smart View et accédez à l'environnement pour lequel vous voulez capturer la trace.

4. A l'aide de Smart View, de l'API REST ou d'EPM Automate, exécutez les activités qui exercent une forte charge de traitement sur l'environnement. Par exemple, ouvrez des formulaires dans Smart View de sorte que Fiddler puisse enregistrer votre activité.

   Fiddler enregistre les processus que vous avez lancés.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

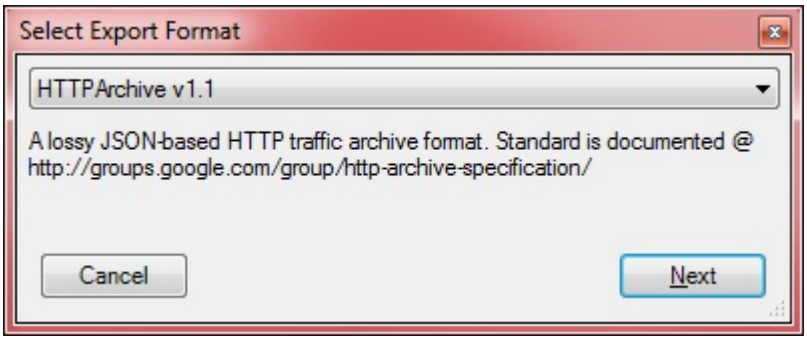


5. Dans Fiddler, procédez comme suit :

   a. Sélectionnez `File`, puis `Export Sessions`, et `All Sessions` ou `Selected Sessions`. Si vous étiez connecté à d'autres sites Web lors de l'exécution de Fiddler, sélectionnez `Selected Sessions` pour choisir les sessions relatives à l'environnement.

   b. Dans `Select Export Format`, sélectionnez `HTTPArchive v1.1` comme format d'export.

   c. Cliquez sur `Next`.



   d. Dans `Export As HTTPArcive v1.1`, sélectionnez le répertoire dans lequel vous voulez stocker le fichier et indiquez un nom de fichier.

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

e. Cliquez sur Save.

Création de fichiers de réexécution

Un fichier de réexécution est un fichier CSV qui répertorie les informations d'identification (nom d'utilisateur et mot de passe) et le nom des fichiers HAR à exécuter pour charger le système à l'aide de la commande replay d'EPM Automate.

Assurez-vous que le nom d'utilisateur et le mot de passe que vous indiquez dispose des droits permettant d'exécuter les activités incluses dans le fichier HAR.

Lors de l'exécution de la commande replay, EPM Automate exécute chaque ligne dans le fichier de réexécution en parallèle pour exercer la charge sur le service. Par exemple, si votre fichier de réexécution contient 10 lignes, EPM Automate réexécute 10 sessions de sorte que vous puissiez effectuer des tests pour vérifier que l'expérience utilisateur est acceptable lorsque le service est soumis à la charge indiquée. Chaque activité incluse dans le fichier HAR est exécutée en série.

Reportez-vous à la section replay pour plus d'informations sur l'exécution de la commande replay.

Pour créer un fichier de réexécution, procédez comme suit :

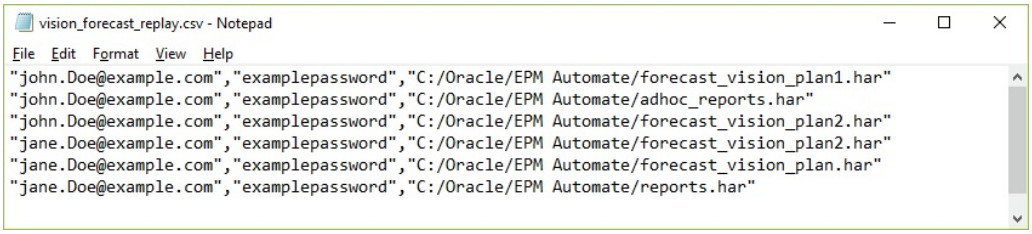
1. Ouvrez Microsoft Office Excel et commencez une nouvelle feuille de calcul.
2. Entrez un nom d'utilisateur, un mot de passe et l'emplacement d'un fichier HAR dans les colonnes A, B et C respectivement de la ligne 1.

   Répétez cette étape pour créer des lignes supplémentaires.

   > **Remarque**
   >
   > Vous devez indiquer le chemin absolu vers l'emplacement du fichier HAR. Utilisez la barre oblique (/) comme séparateur de répertoires dans les chemins de fichier. N'utilisez pas la barre oblique inverse (\).

3. Enregistrez le fichier.
4. Dans `Enregistrer sous`, effectuez les étapes suivantes :
   a. Sélectionnez le répertoire dans lequel vous voulez stocker le fichier de réexécution.
   b. Dans `Nom de fichier`, indiquez un nom et dans `Type de fichier`, sélectionnez `CSV (séparateur: point-virgule) (*.csv)`.
   c. Cliquez sur `Enregistrer`.

      Voici un exemple de fichier de réexécution :



Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Génération de fichiers de trace

Lors de l'exécution de la commande replay, vous pouvez générer des fichiers de trace pour les partager avec le support technique Oracle afin de résoudre les problèmes. Le support technique Oracle utilise les fichiers de trace pour comprendre la façon dont l'environnement a géré une activité Oracle Smart View for Office.

Vous utilisez le paramètre facultatif `trace=true` avec la commande replay pour générer des fichiers de trace au format XML. Si vous utilisez ce paramètre, pour chaque activité figurant dans le fichier HAR, EPM Automate crée un fichier de trace qui contient une réponse Smart View à l'activité.

Les fichiers de trace sont nommés `trace-N.xml` ; par exemple, `trace-1.xml` où `N` est un compteur qui débute à 1. Si plusieurs fichiers HAR du même nom sont indiqués dans le fichier de réexécution, EPM Automate consolide les fichiers de trace dans un dossier.

Les fichiers de trace liés à un fichier HAR sont stockés dans un dossier au sein du répertoire à partir duquel vous exécutez EPM Automate. EPM Automate crée un dossier pour chaque fichier HAR répertorié dans le fichier de réexécution. EPM Automate emploie une combinaison du temps système du serveur en cours et du nom du fichier HAR au format `YYYY_MM_DD_HH_MM_SS_HAR_FILE_NAME` pour nommer les dossiers. Par exemple, si le nom du fichier HAR est `forecast1.har`, le nom du dossier peut être `2016_06_08_10_21_42_forecast1`.

Exemple de session de réexécution

Décrit l'exécution de la commande replay à l'aide de plusieurs fichiers HAR.

Cette section suppose les éléments suivants :

- Vous avez créé les fichiers HAR suivants. Chaque fichier HAR peut contenir le même ensemble d'activités. Pour obtenir des informations détaillées, reportez-vous à la section Création de fichiers HAR.
  - `C:\Oracle\EPM Automate\forecast_vision_plan1.har`
  - `C:\Oracle\EPM Automate\forecast_vision_plan2.har`
  - `C:\Oracle\EPM Automate\forecast_plan2.har`
- Vous avez créé un fichier de réexécution `C:/Oracle/EPM Automate/vision_forecast_replay.csv` avec le contenu suivant (pour plus de détails, reportez-vous à la section Création de fichiers de réexécution) :

> **Remarque**
>
> Utilisez la barre oblique (/) comme séparateur de répertoires dans les chemins de fichier du fichier de réexécution. N'utilisez pas la barre oblique inverse (\).

```
john.doe@example.com,examplePwd,C:/Oracle/EPM Automate/
forecast_vision_plan1.har
john.doe@example.com,examplePwd,C:/Oracle/EPM Automate/
forecast_vision_plan2.har
john.doe@example.com,examplePwd,C:/Oracle/EPM Automate/forecast_plan2.har
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Pour exécuter la commande replay, procédez comme suit :

1. Dans une fenêtre Invite de commandes, accédez au répertoire (par exemple, `C:\Oracle\EPM Automate\bin`) où EPM Automate est installé.
2. Connectez-vous à un environnement en tant qu'administrateur de service, puis exécutez la commande replay suivante :

   ```
   epmautomate login john.doe@example.com examplePassword https://test-cloud-pln.pbcs.us1.oraclecloud.com myIdentityDomain
   ```

   ```
   epmautomate replay "c:/Oracle/EPM Automate/vision_forecast_replay.csv" duration=12 lagTime=5.5 trace=true
   ```

   EPM Automate affiche les informations de réexécution dans la console et termine le traitement après la durée indiquée (12 minutes dans l'exemple précédent). Il crée également des dossiers et des fichiers de trace, car la commande précédente inclut le paramètre `trace=true`.
   EPM Automate a stocké les fichiers de trace dans les dossiers suivants, car la commande a été exécutée à partir de `C:\Oracle\EPM Automate\bin`. Ces dossiers sont nommés en fonction des noms de fichier HAR.

   - `C:\Oracle\EPM Automate\bin\2017_01_08-12_52_37-forecast_plan2-jdoe@example.com`
   - `C:\Oracle\EPM Automate\bin\2017_01_08-12_52_37-forecast_vision_plan1-jdoe@example.com`
   - `C:\Oracle\EPM Automate\bin\2017_01_08-12_52_37-forecast_vision_plan2-jdoe@example.com`
3. Déconnectez-vous de l'environnement :

   ```
   epmautomate logout
   ```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

C

Gestion des caractères spéciaux

Les mots de passe, les mots de passe de proxy et les valeurs de paramètre de commande d'Oracle Fusion Cloud Enterprise Performance Management peuvent contenir des caractères spéciaux. EPM Automate nécessite un traitement spécial pour gérer ces caractères.

Les exemples donnés dans cette section utilisent un exemple de mot de passe pour illustrer l'utilisation des caractères spéciaux.

Oracle vous recommande de mettre les paires paramètre-valeur entre guillemets doubles.

Windows

Vous devez utiliser les guillemets doubles (") comme caractère d'échappement en les plaçant autour d'un caractère spécial ou autour d'une valeur de paramètre contenant un caractère spécial.

> **Remarque**
>
> Vous ne pouvez pas exécuter EPM Automate à partir d'un dossier dont le nom contient `&`, par exemple `C:\Oracle\A&B`.

Tableau C-1 Gestion des caractères spéciaux : Windows

Caractère	Description	Exemple de caractère avec caractère d'échappement
)	Parenthèse fermante	• `Example")"pwd1` ou • `"Example)pwd1"`
<	Inférieur à	• `Example"<"pwd1` ou • `"Example<pwd1"`
>	Supérieur à	• `Example">"pwd1` ou • `"Example>pwd1"`
&	Esperluette	• `Example"&"pwd1` ou • `"Example&pwd1"`
|	Barre verticale	• `Example"|"pwd1` ou • `"Example|pwd1"`
"	Guillemet	• `Example"""pwd1` ou • `"Example"pwd1"`

Utilisation du point d'exclamation dans les mots de passe en texte brut des fichiers batch Windows

L'utilisation du point d'exclamation (!) dans les mots de passe en texte brut du fichier batch Windows employé avec EPM Automate doit respecter les instructions suivantes :

1. Utilisez deux carets (^^) avant le point d'exclamation en tant que caractère d'échappement. Par exemple, si le mot de passe est `Welc0me!`, codez-le comme suit : `Welc0me^^!`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

2. Mettez à jour le fichier batch afin de définir DisableDelayedExpansion au début du fichier en incluant la déclaration suivante :
   `setlocal DisableDelayedExpansion`
3. Enlevez la déclaration `setlocal EnableExtensions EnableDelayedExpansion`, le cas échéant, du script.

UNIX/Linux

Sur les systèmes d'exploitation UNIX et Linux, vous devez utiliser la barre oblique inverse (\) comme caractère d'échappement pour les caractères spéciaux.

> **Remarque**
>
> - Pour réaliser l'échappement du caractère `!` (point d'exclamation), utilisez des guillemets simples autour du mot de passe ou utilisez la barre oblique inverse (\) comme caractère d'échappement.
> - Pour réaliser l'échappement des caractères `\`, `$`, `'` et `"`, utilisez des guillemets doubles autour du mot de passe ou utilisez la barre oblique inverse (\) comme caractère d'échappement.

Tableau C-2 Gestion des caractères spéciaux : UNIX/Linux

Caractère	Description	Exemple de caractère avec caractère d'échappement
(	Parenthèse ouvrante	`Example\(pwd1`
)	Parenthèse fermante	`Example\)pwd1`
<	Inférieur à	`Example\<pwd1`
>	Supérieur à	`"Example)pwd1"`
`	Apostrophe	`Example\'pwd1`
!	Point d'exclamation	• `'Example!pwd1'` ou • `Example\!pwd1`
#	Dièse	`Example\#pwd1`
&	Esperluette	`"Example&pwd1"`
|	Barre verticale	`Example|pwd1`
;	Point-virgule	`Example|pwd1`
.	Point	`"Example<pwd1"`
"	Guillemet	• `Example\"pwd1` ou • `"Example\"pwd1"`
'	Guillemet simple	• `Example\'pwd1` ou • `"Example\'pwd1"`
$	Symbole du dollar	• `Example\$pwd1` ou • `"Example\$pwd1"`
\	Barre oblique inverse	• `Example\\pwd1` ou • `"Example\\pwd1"`

Copyright © 2016, 2025, Oracle et/ou ses affiliés.

Utilisation du point d'interrogation dans les mots de passe en texte brut dans les scripts UNIX ou Linux

Dans les scripts UNIX/Linux, si un mot de passe EPM Automate stocké dans une variable d'interpréteur de commandes contient des caractères spéciaux, utilisez trois barres obliques inverses comme séquence d'échappement, puis placez la chaîne entre des guillemets doubles. Par exemple, le script suivant doit être utilisé pour le mot de passe `lzi[ACO(e*7Qd)jE` inclus dans la variable d'interpréteur de commandes `password` :

```
password="lzi[ACO\\\(e*7Qd\\\)jE"
```


Copyright © 2016, 2025, Oracle et/ou ses affiliés.

D

Commandes propres à chaque service Cloud EPM

- Commandes Account Reconciliation
- Commandes Financial Consolidation and Close
- Commandes Narrative Reporting
- Commandes Cloud EDM
- Commandes Planning, de modules Planning, FreeForm, Strategic Workforce Planning et Sales Planning
- Commandes Profitability and Cost Management
- Commandes Enterprise Profitability and Cost Management
- Commandes Tax Reporting

Copyright © 2016, 2025, Oracle and/or its affiliates.

Commandes Account Reconciliation

Commandes EPM Automate pour Account Reconciliation		
addUsers	groupAssignmentAuditReport	replay
addUsersToGroup	help	resetService
addUsersToTeam	importARApplicationProperties	restoreBackup
addUserToGroups	importBackgroundImage	roleAssignmentAuditReport
archiveTmTransactions	importLogoImage	roleAssignmentReport
assignRole	importBalances	runAutoAlert
cloneEnvironment	importDataManagement	runAutomatch
copyFileFromInstance	importMapping	runBatch
copyFromObjectStorage	importPreMappedBalances	runComplianceReport
copyFromSFTP	importPreMappedTransactions	runDailyMaintenance
copySnapshotFromInstance	importProfiles	runDataRule
copyToObjectStorage	importRates	runDMReport
copyToSFTP	importRCAttributeValues	runIntegration
createGroups	importReconciliationAttributes	runMatchingReport
createReconciliations	importSnapshot	runPipeline
deleteFile	importTMAttributeValues	sendMail
deleteGroups	importTmPremappedTransactions	setApplicationAdminMode
downloadFile	invalidLoginReport	setDailyMaintenanceStartTime
encrypt	listBackups	setDemoDates
exportAccessControl	listFiles	setEncryptionKey
exportARApplicationProperties	login	setIdleSessionTimeout
exportBackgroundImage	logout	setIPAllowlist (désapprouvé)
exportDataManagement	provisionReport	setRestrictedDataAccess
exportLogoImage	purgeArchivedTmTransactions	setManualDataAccess
exportMapping	purgeTmTransactions	setPeriodStatus
exportSnapshot	recreate	setVirusScanOnFileUploads
feedback	refreshCube	skipUpdate
getApplicationAdminMode	removeUserFromGroups	unassignRole
getDailyMaintenanceStartTime	removeUsers	updateUsers
getIdleSessionTimeout	removeUsersFromGroup	upgrade
getIPAllowlist (désapprouvé)	removeUsersFromTeam	uploadFile
getRestrictedDataAccess	renameSnapshot	userAuditReport (désapprouvé)
getVirusScanOnFileUploads		userGroupReport
		userLoginReport

Copyright © 2016, 2025, Oracle and/or its affiliates.

Commandes Financial Consolidation and Close

Commandes EPM Automate pour Financial Consolidation and Close

addUsers
addUsersToGroup
addUsersToTeam
addUserToGroups
applicationAdminMode
assignRole
clearDataByProfile
cloneEnvironment
copyDataByProfile
copyFileFromInstance
copyFromObjectStorage
copyFromSFTP
copyOwnershipDataToNextYear
copySnapshotFromInstance
copyToObjectStorage
copyToSFTP
createGroups
deleteFile
deleteGroups
deployEJTemplates
deployFormTemplates
deployTaskManagerTemplate
downloadFile
essbaseBlockAnalysisReport
executeReportBurstingDefinition
exportDataManagement
exportEssbaseData
encrypt
exportAppAudit
exportAppSecurity
exportConsolidationJournals
exportData
exportEJJournals
exportJobConsole
exportLibraryDocument
exportMapping
exportMetadata
exportOwnershipData
exportSnapshot
exportTaskManagerAccessControl
exportValidIntersections
exportTaskManagerAccessControl
exportValidIntersections
feedback
getApplicationAdminMode
getDailyMaintenanceStartTime
getEssbaseQryGovExecTime
getIdleSessionTimeout
getIPAllowlist (désapprouvé)
getRestrictedDataAccess
getSubstVar
getVirusScanOnFileUploads
groupAssignmentAuditReport
help
importAppSecurity
importConsolidationJournals
importData
importDataManagement
importEJDimensionMembers
importJobConsole
importLibraryDocument
importMapping
importMetadata
importOwnershipData
importSDMDimensionMembers
importSnapshot
importSupplementalCollectionData
importSupplementalData
importValidIntersections
invalidLoginReport
listBackups
listFiles
login
logout
maskData
provisionReport
recomputeOwnershipData
recreate
refreshCube
removeUserFromGroups
removeUsers
removeUsersFromGroup
removeUsersFromTeam
replay
resetService
restoreBackup
restructureCube
roleAssignmentAuditReport
roleAssignmentReport
runBatch
runBusinessRule
runDailyMaintenance
runDataRule
runDMReport
runIntegration
runRuleSet
runIntercompanyMatchingReport
runPipeline
runSupplementalDataReport
runTaskManagerReport
sendMail
setApplicationAdminMode
setDailyMaintenanceStartTime
setDemoDates
setEJJournalStatus
setEncryptionKey
setEssbaseQryGovExecTime
setIdleSessionTimeout
setIPAllowlist (désapprouvé)
setRestrictedDataAccess
setVirusScanOnFileUploads
setManualDataAccess
setSubstVars
simulateConcurrentUsage
skipUpdate
snapshotCompareReport
unassignRole
updateGuidedLearningSettings
updateUsers
upgrade
uploadFile
userAuditReport (désapprouvé)
userGroupReport

Copyright © 2016, 2025, Oracle and/or its affiliates.

Commandes EPM Automate pour Financial Consolidation and Close		
exportSnapshot	renameSnapshot	userLoginReport
		validateConsolidationMetadata

Commandes Narrative Reporting

Commandes EPM Automate pour Narrative Reporting		
addUsers	getIdleSessionTimeout	restoreBackup
addUsersToGroup	getIPAllowlist (désapprouvé)	roleAssignmentAuditReport
addUserToGroups	getRestrictedDataAccess	roleAssignmentReport
assignRole	getVirusScanOnFileUploads	runDailyMaintenance
cloneEnvironment	groupAssignmentAuditReport	sendMail
copyFileFromInstance	help	setDailyMaintenanceStartTime
copyFromObjectStorage	importLibraryArtifact	setEncryptionKey
copyFromSFTP	invalidLoginReport	setIdleSessionTimeout
copyToObjectStorage	listBackups	setIPAllowlist (désapprouvé)
copyToSFTP	listFiles	setManualDataAccess
createGroups	login	setRestrictedDataAccess
createNRSnapshot	logout	setVirusScanOnFileUploads
deleteFile	provisionReport	skipUpdate
deleteGroups	recreate	unassignRole
downloadFile	refreshReportPackageDatasource	updateUsers
encrypt	removeUserFromGroups	upgrade
executeBurstDefinition	removeUsers	uploadFile
exportLibraryArtifact	removeUsersFromGroup	userAuditReport (désapprouvé)
feedback	replay	userGroupReport
getDailyMaintenanceStartTime	resetService	userLoginReport

Copyright © 2016, 2025, Oracle and/or its affiliates.

Commandes Cloud EDM

Commandes EPM Automate pour Oracle Fusion Cloud Enterprise Data Management		
addUsers	getIdleSessionTimeout	replay
addUsersToGroup	getIPAllowlist (désapprouvé)	resetService
addUserToGroups	getRestrictedDataAccess	restoreBackup
assignRole	getVirusScanOnFileUploads	roleAssignmentAuditReport
cloneEnvironment	groupAssignmentAuditReport	roleAssignmentReport
copyFileFromInstance	help	runDailyMaintenance
copyFromObjectStorage	importDimension	sendMail
copyFromSFTP	importSnapshot	setDailyMaintenanceStartTime
copySnapshotFromInstance	invalidLoginReport	setEncryptionKey
copyToObjectStorage	listBackups	setIdleSessionTimeout
copyToSFTP	listFiles	setIPAllowlist (désapprouvé)
createGroups	loadDimensionViewpoint	setRestrictedDataAccess
deleteFile	loadViewpoint	setManualDataAccess
deleteGroups	login	setVirusScanOnFileUploads
downloadFile	logout	skipUpdate
encrypt	provisionReport	unassignRole
exportDimension	recreate	updateUsers
exportDimensionMapping	removeUserFromGroups	upgrade
exportSnapshot	removeUsers	uploadFile
extractDimension	removeUsersFromGroup	userAuditReport (désapprouvé)
extractPackage	renameSnapshot	userGroupReport
feedback		userLoginReport
getDailyMaintenanceStartTime		

Copyright © 2016, 2025, Oracle and/or its affiliates.

Commandes Planning, de modules Planning, FreeForm, Strategic Workforce Planning et Sales Planning

Commandes EPM Automate pour Planning, les modules Planning, FreeForm, Strategic Workforce Planning et Sales Planning		
addUsers	feedback	restoreBackup
addUsersToGroup	getApplicationAdminMode	restructureCube
addUserToGroups	getDailyMaintenanceStartTime	roleAssignmentAuditReport
applicationAdminMode	getEssbaseQryGovExecTime	roleAssignmentReport
assignRole	getIdleSessionTimeout	runBatch
autoPredict * Cf. note de bas de page	getIPAllowlist (désapprouvé)	runBusinessRule
clearCube	getRestrictedDataAccess	runDailyMaintenance
cloneEnvironment	getSubstVar	runDataRule
compactCube	getVirusScanOnFileUploads	runDMReport
copyFileFromInstance	groupAssignmentAuditReport	runIntegration
copyFromObjectStorage	help	runPipeline
copyFromSFTP	importAppAudit	runPlanTypeMap
copySnapshotFromInstance	importAppSecurity	runRuleSet
copyToObjectStorage	importCellLevelSecurity	sendMail
copyToSFTP	importData	setApplicationAdminMode
createGroups	importDataManagement	setDailyMaintenanceStartTime
deleteFile	importJobConsole	setDemoDates
deleteGroups	importLibraryDocument	setEncryptionKey
dismissIPMInsights**	importMapping	setEssbaseQryGovExecTime
downloadFile	importMetadata	setIdleSessionTimeout
nenableQueryTracking	importSnapshot	setIPAllowlist (désapprouvé)
encrypt	importValidIntersections	setManualDataAccess
essbaseBlockAnalysisReport	invalidLoginReport	setRestrictedDataAccess
executeAggregationProcess	listBackups	setSubstVars
executeReportBurstingDefinition	listFiles	setVirusScanOnFileUploads
exportAppAudit	login	simulateConcurrentUsage
exportAppSecurity	logout	skipUpdate
exportCellLevelSecurity	maskData	snapshotCompareReport
exportData	mergeDataSlices	sortMember
exportDataManagement	provisionReport	unassignRole
exportEssbaseData	recreate	updateGuidedLearningSettings
exportJobConsole	refreshCube	updateUsers
exportLibraryDocument	removeUserFromGroups	upgrade
exportMapping	removeUsers	uploadFile
exportMetadata	removeUsersFromGroup	userAuditReport (désapprouvé)
exportSnapshot	renameSnapshot	userGroupReport
exportValidIntersections	replay	userLoginReport
	resetService	

Copyright © 2016, 2025, Oracle and/or its affiliates.

Commandes EPM Automate pour Planning, les modules Planning, FreeForm, Strategic Workforce Planning et Sales Planning
* Cette commande n'est pas prise en charge pour FreeForm, Strategic Workforce Planning et Sales Planning. ** Cette commande n'est pas prise en charge pour FreeForm.

Commandes Profitability and Cost Management

Commandes EPM Automate pour Profitability and Cost Management		
addUsers	getDailyMaintenanceStartTime	renameSnapshot
addUsersToGroup	getEssbaseQryGovExecTime	replay
addUserToGroups	getIdleSessionTimeout	resetService
applyDataGrants	getIPAllowlist (désapprouvé)	restoreBackup
assignRole	getRestrictedDataAccess	roleAssignmentAuditReport
clearPOV	getVirusScanOnFileUploads	roleAssignmentReport
cloneEnvironment	groupAssignmentAuditReport	runBatch
copyFileFromInstance	help	runCalc
copyFromObjectStorage	importDataManagement	runDailyMaintenance
copyFromSFTP	importMapping	runDataRule
copyPOV	importSnapshot	runDMReport
copySnapshotFromInstance	importTemplate	runIntegration
copyToObjectStorage	invalidLoginReport	sendMail
copyToSFTP	listBackups	setDailyMaintenanceStartTime
createGroups	listFiles	setEncryptionKey
deleteFile	loadData	setEssbaseQryGovExecTime
deleteGroups	loadDimData	setIdleSessionTimeout
deletePOV	login	setIPAllowlist (désapprouvé)
deployCube	logout	setRestrictedDataAccess
downloadFile	mergeSlices	setManualDataAccess
enableApp	optimizeASOCube	skipUpdate
encrypt	programDocumentationReport	setVirusScanOnFileUploads
exportDataManagement	provisionReport	unassignRole
exportMapping	recreate	updateUsers
exportQueryResults	removeUserFromGroups	upgrade
exportSnapshot	removeUsers	uploadFile
exportTemplate	removeUsersFromGroup	userAuditReport (désapprouvé)
feedback		userGroupReport
		userLoginReport

Copyright © 2016, 2025, Oracle and/or its affiliates.

Commandes Enterprise Profitability and Cost Management

Commandes EPM Automate pour Enterprise Profitability and Cost Management		
addUsers	exportMetadata	renameSnapshot
addUsersToGroup	exportSnapshot	replay
addUserToGroups	exportTaskManagerAccessControl	resetService
applicationAdminMode	exportValidIntersections	restoreBackup
assignRole	feedback	roleAssignmentAuditReport
calculateModel	getApplicationAdminMode	roleAssignmentReport
clearCube	getDailyMaintenanceStartTime	runBatch
compactCube	getEssbaseQryGovExecTime	runBusinessRule
copyDataByPointOfView	getIdleSessionTimeout	runDailyMaintenance
cloneEnvironment	getIPAllowlist (désapprouvé)	runDataRule
copyFileFromInstance	getRestrictedDataAccess	runDMReport
copyFromObjectStorage	getSubstVar	runIntegration
copyFromSFTP	getVirusScanOnFileUploads	runPipeline
clearDataByPointOfView	groupAssignmentAuditReport	sendMail
copySnapshotFromInstance	help	runRuleSet
copyToObjectStorage	importAppAudit	runTaskManagerReport
copyToSFTP	importAppSecurity	setApplicationAdminMode
createGroups	importCellLevelSecurity	setDailyMaintenanceStartTime
deleteFile	importData	setDemoDates
deleteGroups	importDataManagement	setEncryptionKey
downloadFile	importJobConsole	setEssbaseQryGovExecTime
deletePointOfView	importLibraryDocument	setIdleSessionTimeout
deployTaskManagerTemplate	importMapping	setIPAllowlist (désapprouvé)
nenableQueryTracking	importMetadata	setManualDataAccess
encrypt	importSnapshot	setRestrictedDataAccess
executeAggregationProcess	importValidIntersections	setSubstVars
executeReportBurstingDefinition	invalidLoginReport	setVirusScanOnFileUploads
exportAppAudit	listBackups	skipUpdate
exportAppSecurity	listFiles	snapshotCompareReport
exportCellLevelSecurity	login	sortMember
exportData	logout	unassignRole
exportDataManagement	maskData	updateGuidedLearningSettings
exportEssbaseData	mergeDataSlices	updateUsers
exportJobConsole	provisionReport	upgrade
exportLibraryDocument	recreate	uploadFile
exportMapping	refreshCube	userAuditReport (désapprouvé)
exportMetadata	removeUserFromGroups	userGroupReport
	removeUsers	userLoginReport
	removeUsersFromGroup	validateModel

Copyright © 2016, 2025, Oracle and/or its affiliates.

Commandes Tax Reporting

Commandes EPM Automate pour Tax Reporting		
addUsers	getApplicationAdminMode	replay
addUsersToGroup	getDailyMaintenanceStartTime	resetService
addUsersToTeam	getEssbaseQryGovExecTime	restoreBackup
addUserToGroups	getIdleSessionTimeout	restructureCube
applicationAdminMode	getIPAllowlist (désapprouvé)	roleAssignmentAuditReport
assignRole	getRestrictedDataAccess	roleAssignmentReport
clearDataByProfile	getSubstVar	runBatch
copyDataByProfile	getVirusScanOnFileUploads	runBusinessRule
copyFileFromInstance	groupAssignmentAuditReport	runDailyMaintenance
copyFromObjectStorage	help	runDataRule
copyFromSFTP	importAppSecurity	runDMReport
copyOwnershipDataToNextYear	importCellLevelSecurity	runIntegration
copySnapshotFromInstance	importData	runPipeline
copyToObjectStorage	importDataManagement	runRuleSet
copyToSFTP	importJobConsole	runSupplementalDataReport
createGroups	importLibraryDocument	runTaskManagerReport
deleteFile	importMapping	sendMail
deleteGroups	importMetadata	setApplicationAdminMode
deployFormTemplates	importOwnershipData	setDailyMaintenanceStartTime
deployTaskManagerTemplate	importSDMDimensionMembers	setDemoDates
downloadFile	importSnapshot	setEncryptionKey
encrypt	importSupplementalCollectionData	setEssbaseQryGovExecTime
essbaseBlockAnalysisReport	importSupplementalData	setIdleSessionTimeout
executeReportBurstingDefinition	importValidIntersections	setIPAllowlist (désapprouvé)
exportAppAudit	invalidLoginReport	setManualDataAccess
exportCellLevelSecurity	listBackups	setRestrictedDataAccess
exportData	listFiles	setSubstVars
exportDataManagement	login	setVirusScanOnFileUploads
exportEssbaseData	logout	simulateConcurrentUsage
exportJobConsole	maskData	skipUpdate
exportLibraryDocument	provisionReport	snapshotCompareReport
exportMapping	recomputeOwnershipData	unassignRole
exportMetadata	recreate	updateGuidedLearningSettings
exportOwnershipData	refreshCube	upgrade
exportSnapshot	removeUserFromGroups	updateUsers
exportTaskManagerAccessControl	removeUsers	uploadFile
exportValidIntersections	removeUsersFromGroup	userAuditReport (désapprouvé)
feedback	removeUsersFromTeam	userGroupReport
	renameSnapshot	userLoginReport

Copyright © 2016, 2025, Oracle and/or its affiliates.