Oracle® Fusion Cloud EPM

Trabajar con EPM Automate

F28903-24

Oracle Fusion Cloud EPM Trabajar con EPM Automate,

F28903-24

Copyright © 2016, 2025, Oracle y/o sus filiales.

Autor 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.

Tabla de contenidos

Accesibilidad a la documentación

Comentarios sobre la documentación

1 Acerca de EPM Automate

Instalación de EPM Automate 1-1
- Requisitos de capacidad y puertos 1-2
- Plataformas soportadas 1-2
- Java Runtime Environment y EPM Automate 1-4
- Uso de OpenJDK 1-4
- Instrucciones para Windows 1-5
- Instrucciones para Linux/UNIX/macOS X 1-5
- Ejecución en el servidor de comandos de EPM Automate 1-6

Actualización de EPM Automate 1-6

Desinstalación de EPM Automate 1-7

Descripción del nivel de cifrado de EPM Automate 1-7

Uso del protocolo de autorización OAuth 2.0 con entornos de OCI (Gen2) 1-7

2 Referencia de comandos

Acerca de la ejecución de comandos de EPM Automate 2-1
- Requisitos previos 2-1
- Ubicaciones de archivo predeterminadas 2-3
- Activar protocolo de seguridad de capa de transporte 1.2 2-4
- Uso de comandos de EPM Automate 2-4
- Especificación de varios valores para un parámetro 2-5
- Comportamiento durante el mantenimiento diario 2-5

Ejecución de EPM Automate 2-6
- Windows 2-6
- Linux 2-7
- Ejecución de varias instancias de EPM Automate 2-8

Comandos de un vistazo 2-10

Comandos de EPM Automate 2-16
addUsers 2-16
addUsersToGroup 2-17
addUsersToTeam 2-18
addUserToGroups 2-19
applicationAdminMode 2-20
applyDataGrants 2-21
archiveTmTransactions 2-21
assignRole 2-23
autoPredict 2-24
calculateModel 2-25
clearCube 2-28
clearDataByPointOfView 2-28
clearDataByProfile 2-29
clearPOV 2-30
cloneEnvironment 2-31
compactCube 2-36
copyDataByPointOfView 2-36
copyDataByProfile 2-37
copyFileFromInstance 2-38
copyFromObjectStorage 2-38
copyFromSFTP 2-40
copyOwnershipDataToNextYear 2-41
copyPOV 2-42
copySnapshotFromInstance 2-43
copyToObjectStorage 2-44
copyToSFTP 2-46
createGroups 2-47
createNRSnapshot 2-47
createReconciliations 2-48
deleteFile 2-48
deleteGroups 2-50
deletePointOfView 2-50
deletePOV 2-51
deployCube 2-52
deployEJTemplates 2-52
deployFormTemplates 2-53
deployTaskManagerTemplate 2-54
dismissIPMInsights 2-55
downloadFile 2-55
enableApp 2-56
enableQueryTracking 2-57

encrypt 2-57
essbaseBlockAnalysisReport 2-58
executeAggregationProcess 2-59
executeBurstDefinition 2-60
executeReportBurstingDefinition 2-61
exportAccessControl 2-61
exportAppAudit 2-62
exportAppSecurity 2-63
exportARApplicationProperties 2-64
exportBackgroundImage 2-65
exportCellLevelSecurity 2-65
exportConsolidationJournals 2-66
exportData 2-66
exportDataManagement 2-67
exportDimension 2-67
exportDimensionMapping 2-68
exportEJJournals 2-69
exportEssbaseData 2-70
exportJobConsole 2-71
exportLibraryArtifact 2-74
exportLibraryDocument 2-75
exportLogoImage 2-76
exportMapping 2-77
exportMetadata 2-77
exportOwnershipData 2-78
exportQueryResults 2-79
exportSnapshot 2-81
exportTemplate 2-82
exportTaskManagerAccessControl 2-82
exportValidIntersections 2-83
extractDimension 2-84
extractPackage 2-85
feedback 2-86
getApplicationAdminMode 2-87
getDailyMaintenanceStartTime 2-88
getEssbaseQryGovExecTime 2-88
getIdleSessionTimeout 2-89
getIPAllowlist 2-89
getRestrictedDataAccess 2-90
getSubstVar 2-91
getVirusScanOnFileUploads 2-91
groupAssignmentAuditReport 2-92

help 2-93
importAppAudit 2-93
importAppSecurity 2-94
importARApplicationProperties 2-95
importBackgroundImage 2-95
importBalances 2-96
importCellLevelSecurity 2-96
importConsolidationJournals 2-97
importData 2-98
importDataManagement 2-99
importDimension 2-100
importJobConsole 2-101
importLibraryArtifact 2-101
importLogoImage 2-103
importMapping 2-103
importMetadata 2-104
importOwnershipData 2-106
importPreMappedBalances 2-106
importPreMappedTransactions 2-107
importProfiles 2-108
importRates 2-108
importRCAttributeValues 2-109
importReconciliationAttributes 2-110
importSnapshot 2-111
importSupplementalCollectionData 2-114
importSupplementalData 2-115
importTemplate 2-116
importTMAttributeValues 2-116
importTmPremappedTransactions 2-118
importValidIntersections 2-119
invalidLoginReport 2-120
listBackups 2-121
listFiles 2-122
loadData 2-123
loadDimensionViewpoint 2-124
loadDimData 2-125
loadViewpoint 2-126
login 2-127
logout 2-130
maskData 2-130
mergeDataSlices 2-131
mergeSlices 2-131

optimizeASOCube 2-132
programDocumentationReport 2-133
provisionReport 2-134
purgeArchivedTmTransactions 2-136
purgeTmTransactions 2-137
recomputeOwnershipData 2-138
recreate 2-139
refreshCube 2-142
removeUserFromGroups 2-142
removeUsers 2-143
removeUsersFromGroup 2-144
removeUsersFromTeam 2-145
renameSnapshot 2-146
replay 2-147
resetService 2-148
restoreBackup 2-149
restructureCube 2-151
roleAssignmentAuditReport 2-151
roleAssignmentReport 2-153
runAutomatch 2-154
runBatch 2-155
runBusinessRule 2-155
runCalc 2-156
runComplianceReport 2-158
runDailyMaintenance 2-159
runDataRule 2-160
runDMReport 2-162
runIntegration 2-163
runIntercompanyMatchingReport 2-167
runMatchingReport 2-168
runPipeline 2-169
runPlanTypeMap 2-171
runRuleSet 2-171
runSupplementalDataReport 2-172
runTaskManagerReport 2-173
sendMail 2-174
setApplicationAdminMode 2-175
setDailyMaintenanceStartTime 2-176
setDemoDates 2-177
setEJJournalStatus 2-178
setEncryptionKey 2-179
setEssbaseQryGovExecTime 2-180

setIdleSessionTimeout 2-180
setIPAllowlist 2-181
setManualDataAccess 2-182
setPeriodStatus 2-183
setRestrictedDataAccess 2-184
setSubstVars 2-184
setVirusScanOnFileUploads 2-185
simulateConcurrentUsage 2-186
skipUpdate 2-190
snapshotCompareReport 2-192
sortMember 2-193
unassignRole 2-194
updateGuidedLearningSettings 2-196
updateUsers 2-196
upgrade 2-197
uploadFile 2-198
userAuditReport 2-199
userGroupReport 2-201
validateConsolidationMetadata 2-201
validateModel 2-202
Códigos de salida 2-203

3 Escenarios de ejemplo de ejecución de comandos

Acerca de la copia de scripts de ejemplo 3-1
Escenarios de ejemplo para todos los servicios 3-1
Copia de seguridad de instantáneas de aplicación en un equipo 3-3
Información a los usuarios de la finalización del mantenimiento diario 3-5
Copia de una instantánea en o de Oracle Object Storage 3-12
Creación de usuarios y asignación a roles predefinidos 3-14
Recuento del número de usuarios con licencia (usuarios asignados a roles) 3-18
Creación de informes de auditoría de usuarios asignados a roles 3-20
Creación de informe de auditoría de asignación de roles y revocación 3-24
Enmascaramiento de registros de acceso e informe de actividades para cumplir las normas de privacidad 3-27
Automatización de descargas de informes de actividad en un equipo local 3-32
Descarga de logs de acceso desde un entorno 3-36
Automatización de la clonación de entornos 3-39
Clonación diaria del entorno principal al entorno de espera una vez completado el mantenimiento diario en el entorno principal 3-43
Eliminación de archivos innecesarios desde un entorno 3-52
Búsqueda y descarga de archivos de un entorno 3-54
Nueva creación de un entorno antiguo de Cloud EPM para auditorías 3-55

Automatización de auditoría y conformidad sobre el acceso a la base de datos 3-66
Replicación de usuarios y asignación de role predefinidos 3-77
Replicación de los usuarios de un dominio de identidad en otro 3-78
Replicar asignaciones de roles predefinidos de un entorno a otro 3-84
Creación de una cadencia de cambio de versión trimestral de Cloud EPM 3-91
Instrucciones y script para Windows 3-92
Instrucciones y script para UNIX/Linux 3-95
Script de Groovy 3-99
Creación de una cadencia de cambio de versión trimestral de Cloud EPM con ciclos de prueba de seis semanas 3-103
Escenarios de ejemplo para Planning, Consolidation, Tax Reporting y Enterprise Profitability and Cost Management 3-117
Automatización de la exportación de un gran número de celdas desde un cubo de almacenamiento agregado 3-117
Importación de metadatos a una aplicación 3-127
Importación de datos, ejecución de un script de cálculo y copia de datos desde una base de datos de almacenamiento de bloques a una base de datos de almacenamiento agregado 3-129
Exportación y descarga de metadatos y datos 3-131
Exportación y descarga de datos de aplicación 3-134
Automatización del archivado de los registros de auditoría de aplicación 3-136
Script de Windows 3-137
Script de Linux 3-138
Carga de un archivo de datos en un entorno y ejecución de una regla de carga de datos 3-139
Automatización de la integración de datos diaria 3-142
Escenarios de ejemplo para Account Reconciliation 3-144
Carga de balances preformateados en un periodo 3-144
Carga e importación de una instantánea de copia de seguridad 3-146
Archivado de transacciones confrontadas antiguas y depuración de transacciones archivadas 3-148
Escenarios de ejemplo para Profitability and Cost Management 3-155
Importación de metadatos en la aplicación 3-155
Importación de datos y ejecución de reglas del programa 3-157
Escenarios de ejemplo para Oracle Enterprise Data Management Cloud 3-160
Sincronización de dimensiones y asignaciones de Oracle Enterprise Data Management Cloud con aplicaciones de Cloud EPM 3-160
Sincronización de dimensiones de Cloud EPM con aplicaciones de Oracle Enterprise Data Management Cloud 3-162
Automatización de la ejecución de scripts 3-163
Supervisión de las actividades de EPM Automate 3-164

4 Ejecución de comandos sin instalar EPM Automate

Entornos que soportan la ejecución de comandos en el servidor 4-1

Fuentes de información 4-2
Comandos soportados 4-2
Métodos que usar para ejecutar EPM Automate con un script Groovy del servidor 4-2
Clonación de un entorno con un script de Groovy del servidor 4-3
Envío por correo electrónico de un informe de actividad con un script de Groovy del servidor 4-5

5 Replicación de un entorno de Cloud EPM

Configuración de la replicación diaria 5-1
Configuración de replicaciones a petición 5-2
Configuración del entorno secundario 5-3

A Preparación para ejecutar el comando simulateConcurrentUsage

Creación del archivo requirement.csv A-2
Creación de archivos de entrada A-3
Archivo de entrada de apertura de formulario A-4
Archivo de entrada de almacenamiento de formulario A-4
Archivo de entrada de ejecución de regla de negocio A-5
Archivo de entrada de ejecución de conjunto de reglas de negocio A-5
Archivo de entrada de ejecución de regla de datos A-6
Archivo de entrada de cuadrícula ad hoc A-6
Archivo de entrada de ejecución de informe A-7
Archivo de entrada de ejecución de libro A-7
Creación del archivo UserVarMemberMapping.csv A-8
Creación del archivo options.xml A-8
Creación del archivo users.csv A-8
Creación y carga del archivo ZIP de entrada en el entorno A-9
Informe Simulación de uso simultáneo de ejemplo A-9

B Preparación para ejecutar el comando replay

Acerca del comando replay B-1
Requisitos previos B-1
Creación de archivos HAR B-2
Creación de archivos de reproducción B-5
Generación de archivos de rastreo B-6
Sesión de reproducción de ejemplo B-6

C Manejo de caracteres especiales

D Comandos específicos de cada servicio de Cloud EPM

Comandos de Account Reconciliation D-2

Comandos de Financial Consolidation and Close D-3

Comandos de Narrative Reporting D-4

Comandos de Oracle Enterprise Data Management Cloud D-5

Comandos de Planning, Planning Modules, FreeForm, Strategic Workforce Planning y Sales Planning D-6

Comandos de Profitability and Cost Management D-7

Comandos de Enterprise Profitability and Cost Management D-8

Comandos de Tax Reporting D-9

Accesibilidad a la documentación

Para obtener información acerca del compromiso de Oracle con la accesibilidad, visite el sitio web del Programa de Accesibilidad de Oracle en http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Acceso a Oracle Support

Los clientes de Oracle que hayan adquirido soporte disponen de acceso a soporte electrónico a través de My Oracle Support. Para obtener información, visite http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info o http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs si tiene problemas de audición.

Comentarios sobre la documentación

Para hacernos llegar sus comentarios sobre esta documentación, haga clic en el botón Comentarios en la parte inferior de la página de cualquier tema de Oracle Help Center. También puede enviar un correo electrónico a epmdoc_ww@oracle.com.

1
Acerca de EPM Automate

EPM Automate permite a los usuarios realizar tareas de forma remota en los entornos de Oracle Fusion Cloud Enterprise Performance Management.

Los administradores del servicio de Cloud EPM pueden automatizar muchas tareas repetibles, incluidas las siguientes:

- Importación y exportación de metadatos, datos, artefactos e instantáneas de aplicación, plantillas y asignaciones de Data Management
- Carga de archivos en entornos, visualización de archivos y supresión de archivos del servicio
- Descarga de instantáneas, informes y archivos de metadatos y datos del servicio
- Ejecución de reglas de negocio en datos y actualización de la aplicación
- Copiar datos de una base de datos a otra; por lo general, desde una base de datos de almacenamiento de bloques a una base de datos de almacenamiento agregada o desde una base de datos de almacenamiento de bloques a otra base de datos de almacenamiento de bloques
- Ejecutar una regla de lote de Data Management
- Generación de informes, informes de aprovisionamiento e informes de auditoría de usuarios de Data Management
- Importación de datos de balance previamente asignados, tipos de cambio, transacciones previamente asignadas, datos de balance y perfiles en Account Reconciliation
- Copiar perfiles en un periodo para iniciar el proceso de conciliación
- Despliegue el cubo de cálculo de aplicaciones de Profitability and Cost Management
- Borre, copie y suprima puntos de vista en aplicaciones de Enterprise Profitability and Cost Management y Profitability and Cost Management
- Reproduce la carga de la API de REST de Oracle Smart View for Office en un entorno con el fin de activar las pruebas de rendimiento si hay una carga muy elevada.
- Importación de datos adicionales de un archivo en Financial Consolidation and Close

Puede crear scripts que sean capaces de realizar una amplia serie de tareas y automatizar su ejecución mediante un programador. Por ejemplo, puede crear un script para descargar la copia de seguridad de mantenimiento diaria de los entornos y crear copias de seguridad locales de los artefactos y datos.

Tutorial: Cómo ejecutar comandos de Planning con EPM Automate

Instalación de EPM Automate

Instale EPM Automate para ejecutar comandos. Algunos comandos también se pueden ejecutar directamente en Oracle Fusion Cloud Enterprise Performance Management con scripts de Groovy sin instalar EPM Automate.

El instalador de EPM Automate para Windows, Linux/UNIX y macOS X está disponible en su entorno de Cloud EPM.

Dado que Windows de la versión 10 y versiones posteriores solo permite a los administradores de Windows instalar EPM Automate, solo los administradores de Windows pueden instalarlo y actualizarlo. EPM Automate lo puede actualizar el usuario que lo ha instalado u otro administrador de Windows.

En esta sección figuran los siguientes apartados:

- Requisitos de capacidad y puertos
- Plataformas soportadas
- Java Runtime Environment y EPM Automate
- Uso de OpenJDK
- Instrucciones para Windows
- Instrucciones para Linux/UNIX/macOS X
- Ejecución en el servidor de comandos de EPM Automate

Requisitos de capacidad y puertos

Como EPM Automate es un cliente ligero, no necesita una huella del cliente grande. Todo el procesamiento tiene lugar en Oracle Fusion Cloud Enterprise Performance Management.

Puede instalar EPM Automate en máquinas estándar del cliente, en máquina virtuales y en máquinas de Oracle Integration Cloud que puedan acceder a hosts externos en una conexión HTTP segura.

EPM Automate se conecta a Cloud EPM con el puerto TLS estándar (puerto 443). No tiene que abrir más puertos de salida adicionales para EPM Automate.

En este momento, EPM Automate no soporta la autenticación TLS mutua (mTLS).

Plataformas soportadas

EPM Automate se puede instalar en máquinas virtuales y en máquinas de Oracle Integration Cloud (OIC) que puedan acceder a hosts externos en una conexión HTTP segura.

Note:

- EPM Automate se puede emplear solo en sistemas operativos de 64 bits que están actualmente soportados por el proveedor del sistema operativo.
- EPM Automate no funciona con el proxy SOCKS; solo funciona con el proxy HTTP/HTTPS.
- EPM Automate soporta los mecanismos de autentificación Básica, Resumen, Kerberos, Negociación y NTLM para conectarse al servidor proxy.
- EPM Automate puede conectar a Oracle Fusion Cloud Enterprise Performance Management mediante puertas de enlace de API, como Google APIGEE, IBM Data Power y otros servidores proxy inversos.
  Para que esto funcione, configure la puerta de enlace o el proxy inverso. Para ello, establezca el destino como la URL de su entorno de Cloud EPM sin ningún contexto como `/epmcloud`. Ejemplo: `https://epm-idDomain.epm.dataCenterRegion.oraclecloud.com`. A continuación, use la URL de proxy inverso en lugar de la URL de Cloud EPM en el comando login . Para obtener información de configuración, consulte la documentación del gateway o del servidor proxy.

  Al configurar los valores de proxy, asegúrese de transferir el código de respuesta de Cloud EPM a EPM Automate sin realizar ninguna modificación en él para permitir a EPM Automate procesar los códigos de respuesta de la forma correcta como 200, 206, 400, 404, 500, 501, etc. Por ejemplo, para IBM DataPower, establezca `proxy HTTP Response` en `ON`. Además, la puerta de enlace de la API debería permitir los métodos HTTP (GET, POST, PUT, PATCH y DELETE).

En equipos Linux y UNIX, EPM Automate busca las siguientes variables de entorno para determinar la configuración del proxy HTTP o HTTPS:

- proxyHost
- proxyPort

Ejemplos de configuraciones de proxy HTTP:

```
export proxyHost=host.example.com

export proxyPort=8000
```

Ejemplos de configuraciones de proxy HTTPS:

```
export proxyHost=host.example.com

export proxyPort=8080
```

> **Note:**
>
> EPM Automate puede usar el protocolo de autenticación de OAuth 2.0 para acceder a entornos de Cloud EPM (si se han configurado para OAuth) para ejecutar comandos, especialmente para automatizar la ejecución de comandos.
>
> En entornos en los que se use la autentificación básica, EPM Automate no funciona con las credenciales de inicio de sesión único (proveedor de identidad). Puesto que los usuarios no pueden conectarse con credenciales corporativas, las cuentas del usuario para acceder a EPM Automate deben mantenerse en el dominio de identidad. Si la suscripción está configurada para inicio de sesión único, también debe permitir a los usuarios de EPM Automate conectarse con sus credenciales de dominio de identidad. Consulte Cómo permitir la conexión con credenciales de dominio de identidad en *Administración de Oracle Cloud Identity Management*.

Instrucciones de descarga: Descarga e instalación de clientes en *Guía de introducción para administradores.*

Java Runtime Environment y EPM Automate

La instalación de EPM Automate en Windows instala la versión adecuada de Java Runtime Environment (JRE). Sin embargo, JRE no está incluido en los instaladores de Linux, Unix y macOS X. Debe tener acceso a una instalación de JRE (de la versión 8 a la versión 11) para utilizar EPM Automate.

Tiene derecho a utilizar Oracle Java Standard Edition (SE) con EPM Automate sin necesidad de comprar una suscripción de Java SE por separado. Para obtener información detallada sobre la licencia de Oracle JDK con EPM Automate, consulte Documento 1557737.1 de los Servicios de Soporte Oracle: "Support Entitlement for Java SE When Used As Part of Another Oracle Product" (Derechos de soporte para Java SE cuando se utiliza como parte de otro producto Oracle) .

Uso de OpenJDK

Puede utilizar OpenJDK versión 14 o superior en lugar de JRE en las plataformas Linux, Unix y macOS X.

OpenJDK, el JDK gratis de Oracle con licencia GPL listo para la producción, se puede descargar de https://openjdk.java.net. Las instrucciones para instalar OpenJDK también están disponibles en este sitio web.

Antes de empezar una sesión de EPM Automate, defina la variable de entorno `JAVA_HOME` para que apunte a la instalación de OpenJDK:

Ejemplo de macOS X (asumiendo un shell de Bash) para utilizar OpenJDK versión 14 instalado en el directorio inicial.

```
cd ~/
export JAVA_HOME=$(/usr/jdk-14.jdk/Contents/Home)
```

Ejemplo de Linux (asumiendo un shell de Bash) para utilizar OpenJDK versión 14 instalado en el directorio inicial.

```
cd ~/
export JAVA_HOME=/openjdk/jdk-14.0.2
```

Instrucciones para Windows

De forma predeterminada, EPM Automate se instala en `C:/Oracle/EPM Automate`.

Para instalar EPM Automate:

1. En la computadora con Windows donde desea instalar EPM Automate, obtenga acceso a un entorno.
2. En la página de inicio, puede acceder a **Configuración y acciones** haciendo clic en el nombre de usuario.
3. Haga clic en **Descargas**.
4. En la página Descargas, haga clic en **Descargar para Windows** en la sección EPM Automate.
5. Guarde el instalador en el equipo.
6. Haga clic con el botón derecho en el instalador (`EPM Automate.exe`) y seleccione `Ejecutar como administrador`.
7. En **Control de cuentas de usuario**, haga clic en **Sí**.
8. Siga las peticiones de datos en pantalla para completar la instalación.

Instrucciones para Linux/UNIX/macOS X

EPM Automate requiere acceso a un despliegue de una versión soportada de JRE (de la versión 8 a la 11). La variable de entorno `JAVA_HOME` debe establecerse para apuntar a la instalación de JRE.

Para instalar EPM Automate:

1. Acceda a un entorno.
2. En la página de inicio, puede acceder a **Configuración y acciones** haciendo clic en el nombre de usuario.
3. Haga clic en **Descargas**.
4. En la página Descargas, haga clic en **Descargar para Linux/macOS X** en la sección EPM Automate.
5. Guarde el instalador (`EPMAutomate.tar`) en un directorio en el que tenga privilegios de lectura/escritura/ejecución.
6. Realice estos pasos:
   a. Extraer el contenido del instalador
   b. Definir las variables de entorno necesarias
   c. Cambie al directorio `epmautomate/bin` dentro del directorio en el que se haya extraído el contenido del instalador
   d. Ejecute `epmautomate.sh`:

En los scripts siguientes, recuerde utilizar las rutas de directorio reales donde sea necesario. Por ejemplo, si se extrae `EPMAutomate.tar` en `HOME/oracle/epmautomate/bin`, úselo como el valor de `EPMAutomate_extract_directory`.

Ejemplo de macOS X (shell de Bash predeterminado) para instalar y ejecutar desde su escritorio de inicio.

```
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
```

Ejemplo de Linux (asumiendo un shell de bash) para instalar y ejecutar desde el directorio inicial. Se presupone el uso de la versión de JDK 1.8.0_191.

```
cd ~/
tar xf directory_containing_downloaded_installer/EPMAutomate.tar
export JAVA_HOME=/opt/jdk1.8.0_191
export PATH ~/Downloads/epmautomate/bin:$PATH
cd EPMAutomate_extract_directory/epmautomate/bin
./epmautomate.sh
```

Ejecución en el servidor de comandos de EPM Automate

Algunos comandos de EPM Automate se pueden ejecutar directamente en Oracle Fusion Cloud Enterprise Performance Management con Groovy. No es necesario instalar EPM Automate para ejecutar comandos con scripts de Groovy.

Tenga en cuenta que la ejecución en el servidor de los comandos no es igual que la ejecución de scripts de Groovy en un equipo cliente para ejecutar comandos de EPM Automate.

Para obtener información detallada, consulte Ejecución de comandos sin instalar EPM Automate.

Actualización de EPM Automate

EPM Automate le permite actualizar de forma silenciosa a la versión más reciente.

Después de iniciar sesión, EPM Automate identifica la versión instalada actualmente. Si la versión instalada no es la más reciente disponible, le informa de que hay una nueva versión disponible.

Ejecute el comando upgrade para actualizar de forma silenciosa EPM Automate.

Desinstalación de EPM Automate

> **Note:**
>
> Utilice estos pasos solo si desea eliminar EPM Automate de una máquina cliente. Si desea actualizar la versión actual de EPM Automate, utilice las instrucciones descritas en Actualización de EPM Automate.

Windows

Para desinstalar EPM Automate:

1. En **Configuración**, abra **Aplicaciones** y, a continuación, **Aplicaciones instaladas**.
2. Busque EPM Automate en la lista de aplicaciones instaladas.
3. En el menú de **desbordamiento**, seleccione **Desinstalar**.
4. Haga clic en **Desinstalar**.
5. Suprima los archivos restantes del directorio de instalación, normalmente, C:\Oracle\Epm Automate.

Linux/UNIX/macOS X

Suprima el directorio de EPM Automate y el archivo `EPMAutomate.tar`.

Descripción del nivel de cifrado de EPM Automate

Oracle Fusion Cloud Enterprise Performance Management utiliza seguridad de capa de transporte (TLS) con un algoritmo hash criptográfico SHA-2/SHA-256 para proteger la comunicación con EPM Automate.

Uso del protocolo de autorización OAuth 2.0 con entornos de OCI (Gen2)

EPM Automate puede usar el protocolo de autenticación de OAuth 2.0 para acceder a entornos de Oracle Fusion Cloud Enterprise Performance Management de OCI (GEN 2) para ejecutar comandos, especialmente para automatizar la ejecución de comandos.

Para activar el acceso mediante OAuth 2.0, un administrador de dominio de identidad debe registrar su aplicación como cliente público en Oracle Cloud Identity Services. OAuth se aplica en la aplicación; no en la suscripción.

Para obtener instrucciones detalladas sobre la configuración de OAuth 2.0 para sus entornos de OCI (Gen 2), consulte Autentificación con OAuth 2. Solo para entornos OCI (Gen 2) en la guía *API de REST*.

> **Note:**
>
> La autenticación básica funciona incluso cuando OAuth está activado para un entorno. Asegúrese de no sobrescribir el archivo de contraseñas cifrado existente si tiene previsto usarlo en el futuro.

Creación de un archivo de contraseñas cifrado que incluye un símbolo de refrescamiento y un ID de cliente

Los administradores del servicio que deseen usar OAuth 2.0 para el acceso a entornos de EPM Automate necesitan esta información para crear su archivo de contraseñas cifrado que, a continuación, se usa para conectarse al entorno:

- Símbolo de refrescamiento
  Consulte Autentificación con OAuth 2. Solo para entornos OCI (Gen 2) en la guía *API de REST* para obtener instrucciones detalladas sobre cómo obtener el símbolo de refrescamiento.
- ID de cliente
  El ID de cliente se genera cuando el administrador de dominio de identidad configura la aplicación para OAuth. Se puede ver en la separador Configuración de la aplicación, en **Información general**.

Para crear el archivo de contraseña cifrado para la autenticación de OAuth:

1. Inicie una sesión de EPM Automate.
2. Ejecute un comando similar al siguiente:
   `epmautomate encrypt REFRESH_TOKEN ENCRYPTION_KEY PASSWORD_FILE ClientID=CLIENT_ID`, donde `REFRESH_TOKEN` es el símbolo de refrescamiento descifrado del almacén seguro, `ENCRYPTION_KEY` es cualquier clave privada para cifrar la contraseña y `PASSWORD_FILE` es el nombre y la ubicación del archivo que almacena el símbolo de refrescamiento cifrado. El archivo de contraseña debe llevar la extensión `.epw`.

   Consulte encrypt para obtener instrucciones detalladas.
3. Utilice el archivo de contraseña recién generado para conectarse mediante OAuth. En el caso de ejecuciones de scripts automatizadas, asegúrese de actualizar los scripts para que apunten al archivo de contraseñas recién generado.

2

Referencia de comandos

- Acerca de la ejecución de comandos de EPM Automate
- Ejecución de EPM Automate
- Comandos de un vistazo
- Comandos de EPM Automate
- Códigos de salida

Algunos comandos de EPM Automate se aplican a todos los procesos de negocio, mientras que otros se aplican a un grupo de procesos de negocio. A no ser que se especifique lo contrario, un comando aplicable a un proceso de negocio específico (por ejemplo, Planning) no funciona con un proceso de negocio diferente (por ejemplo, Financial Consolidation and Close). Los intentos de ejecutar un comando en un proceso de negocio que no lo soporta producirán un error. Para obtener una lista de comandos aplicables a cada proceso de negocio, consulte Comandos específicos de cada servicio de Cloud EPM.

Acerca de la ejecución de comandos de EPM Automate

En todos los servicios de Oracle Fusion Cloud Enterprise Performance Management se utilizan comandos de EPM Automate para la administración remota de los entornos.

- Requisitos previos
- Ubicaciones de archivo predeterminadas
- Activar protocolo de seguridad de capa de transporte 1.2
- Uso de comandos de EPM Automate
- Especificación de varios valores para un parámetro
- Comportamiento durante el mantenimiento diario

Requisitos previos

En esta sección se enumeran los requisitos para utilizar EPM Automate, como el uso de las credenciales de Oracle Fusion Cloud Enterprise Performance Management y las ubicaciones de archivo predeterminadas en los entornos.

General

Los usuarios de Cloud EPM pueden utilizar las credenciales de dominio de identidad para conectarse a un entorno mediante el uso de EPM Automate. Los roles predefinidos y de aplicación asignados al usuario deciden los comandos que un usuario puede ejecutar.

- Además, se requiere el rol de administrador de dominio de identidad para ejecutar comandos que agreguen o supriman usuarios en el dominio de identidad.
- Cualquier archivo necesario para ejecutar un comando debe existir en el entorno. Utilice el comando uploadFile para cargar archivos.

Consulte Ubicaciones de archivo predeterminadas para obtener más información sobre la ubicación de archivo predeterminada utilizada por cada servicio.

- Uso de extensión de archivo en comandos:
  - Especifique el nombre completo del archivo, incluida la extensión de archivo (por ejemplo, data.csv), para ejecutar comandos que realizan operaciones de archivos. Ejemplo de comandos de operaciones de archivos incluye `deletefile` `listfiles` y `uploadfile`.
  - No utilice extensiones de archivos para ejecutar comandos que realizan operaciones de migración. Las operaciones de migración requieren que se especifique el nombre de una instantánea.
- Los valores de parámetro que contienen un carácter de espacio (por ejemplo, comentarios, nombres de ubicación y rutas de carpetas) deben ir entre comillas.

Planning

- Trabajos
  Para muchos de los comandos que se describen en la siguiente sección se requieren trabajos. Los trabajos son acciones, como la importación o exportación de datos, que se pueden iniciar inmediatamente o programar para que se realicen posteriormente, por ejemplo, la importación o exportación de datos y el refrescamiento de la base de datos.

  Con la consola de trabajos, debe crear los trabajos adecuados para realizar las siguientes operaciones. Para obtener instrucciones detalladas sobre la creación de trabajos en Planning, consulte Gestión de trabajos en *Administración de Planning*.
  - Importar datos a una aplicación
  - Exportar datos de una aplicación
  - Importar metadatos a una aplicación
  - Exportar metadatos de una aplicación
  - Copie los datos desde una base de datos de almacenamiento de bloques en una base de datos de almacenamiento agregado o desde una base de almacenamiento agregado a otra base de datos de almacenamiento de bloques
- Reglas de negocio
  Las reglas de negocio que desee ejecutar deben existir en la aplicación.

  Puede utilizar Calculation Manager para crear reglas de negocio, que a continuación se desplegarán en la aplicación. Consulte *Diseño con Calculation Manager*.

Data Management

- Reglas de datos
  Las reglas de carga de datos definen cómo Data Management carga los datos de un archivo. Tiene que haber predefinido reglas de carga de datos para cargar los datos con EPM Automate.
- Lotes
  Puede cargar datos utilizando los lotes definidos en Data Management. Si utilizan un lote, los usuarios pueden combinar muchas reglas de carga en un lote y ejecutarlas en serie o en paralelo.

Ubicaciones de archivo predeterminadas

Ubicación de carga predeterminada

De forma predeterminada, todos los archivos cargados en Oracle Fusion Cloud Enterprise Performance Management se almacenan en una ubicación predeterminada a la que puede acceder Migration.

Debe cargar archivos que vaya a procesar Migración, por ejemplo, instantáneas que dese importar en el servicio, en la ubicación predeterminada.

Buzón de entrada y buzón de salida

Las ubicaciones de la bandeja de entrada y del buzón de salida pueden variar según el proceso de negocio de Cloud EPM. Usa el buzón para cargar archivos que desea importar o procesar de alguna de otra forma con procesos de negocio distintos de Profitability and Cost Management. Data Management puede procesar los archivos del buzón o de uno de sus directorios.

Normalmente, Cloud EPM almacena archivos que haya generado mediante los procesos de negocio, por ejemplo, archivos de exportación de datos o metadatos, en el buzón de salida.

- El buzón en la que EPM Automate carga los archivos y la bandeja de salida que almacena los archivos para descarga es accesible para estas aplicaciones. Debe cargar archivos en esta ubicación si planea procesarlos mediante un proceso que es nativo para estas aplicaciones. También puede cargar los archivos en la bandeja de salida.
  - Planning
  - Planning Modules
  - Account Reconciliation
  - Financial Consolidation and Close
  - Tax Reporting
  - Narrative Reporting
  - Enterprise Profitability and Cost Management

  Puede utilizar el explorador de bandeja de entrada/buzón de salida para examinar los archivos almacenados en la ubicación predeterminada. Las instantáneas de la aplicación que cree con EPM Automate no aparecen en el explorador de bandeja de entrada/buzón de salida; puede verlas en el separador Instantáneas de Migración.
- Los archivos que se van a procesar mediante un proceso de Profitability and Cost Management se deben cargar en `profitinbox`. También puede cargar los archivos en profitoutbox. Los archivos exportados por los procesos de Profitability and Cost Management se almacenan en `profitinbox`. Utilice el explorador de archivos para examinar estos archivos.
- Los archivos que se van a procesar mediante Data Management deben estar disponibles en la bandeja de entrada o en una carpeta dentro de ella. De forma predeterminada, los archivos exportados mediante Data Management se almacenan en el buzón de salida, mientras que las salidas de informe de Data Management se almacenan en la carpeta `outbox/report` de Data Management. Utilice el explorador de archivos de Data Management para examinar estos archivos.

- Oracle Fusion Cloud Enterprise Data Management utiliza la ubicación predeterminada para importar y exportar archivos que se cargan, se copian o se descargan. Los archivos de la ubicación predeterminada pueden verse utilizando el comando `ListFiles`.

Archivos de registro

Cada ejecución de un comando de EPM Automate genera un archivo de depuración, el cual se suprime automáticamente si el comando se ejecuta correctamente. Si se produce un error durante la ejecución del comando, el archivo de depuración del comando fallido se mantiene en el directorio desde el que ejecuta EPM Automate. De forma predeterminada, este directorio es `Oracle/epm automate/bin` (Windows) o `home/user/epmautomate/bin` (Linux/UNIX).

Los archivos de depuración de EPM Automate utilizan la siguiente convención de nomenclatura:

`commandname_date_timestamp.log`. Por ejemplo, si ejecuta un comando `listfiles` fallido a las 09:28:02 el 23 de noviembre de 2020, el nombre del archivo de depuración es `listfiles_23_11_2020_09_28_02.log`.

No se puede suprimir la creación del archivo de depuración para un comando fallido. Sin embargo, puede escribir la información de depuración y la salida del comando a un archivo de un directorio diferente mediante la adición de `-d` junto con un nombre de archivo de depuración y flujos de error y salida (`-d >> c:\logs\LOG_FILE_NAME.log 2>&1`) al final del comando, como se muestra en el siguiente ejemplo de Windows:

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

Activar protocolo de seguridad de capa de transporte 1.2

EPM Automate se debe instalar en un sistema operativo que admita el protocolo de seguridad de capa de transporte (TLS) 1.2 o superior.

Para garantizar el máximo nivel de seguridad para la autentificación y el cifrado de datos, EPM Automate solo admite TLS 1.2. Si TLS 1.2 no está activado en la computadora desde la que se ejecuta EPM Automate, se muestra el error `EPMAT-7: No es posible conectar. Protocolo no soportado: se muestra el error HTTPS`. Para resolverlo, pida al administrador de TI que active TLS 1.2.

Los procedimientos para activar TLS 1.2 dependen del sistema operativo. Use estas fuentes de información; puede haber recursos web similares para otros sistemas operativos admitidos:

- Realice la actualización para activar TLS 1.1 y TLS 1.2 como protocolos de seguridad predeterminados en WinHTTP en Windows para obtener información sobre la activación de TLS 1.2 para computadoras con Windows.
- Protección de la configuración de TLS para obtener información sobre la activación de TLS 1.2 en OpenSSL para Red Hat Enterprise Linux.

Uso de comandos de EPM Automate

Secuencia de parámetros de comandos

Todos los parámetros obligatorios para un comando se deben transferir en la secuencia identificada en la sintaxis del comando. Los parámetros obligatorios y sus valores preceden a los parámetros opcionales, que se pueden transferir en cualquier secuencia. Los parámetros opcionales no son posicionales.

Por ejemplo, tenga en cuenta la siguiente sintaxis del comando `login`:

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

Este comando tiene tres parámetros obligatorios: `USERNAME`, `PASSWORD` y `CLOUD_EPM_BASE_URL`, que deben aparecer en la secuencia identificada en la sintaxis. El comando devolverá un error si no se mantiene esta secuencia. Los parámetros opcionales `ProxyServerUserName`, `ProxyServerPassword` y `ProxyServerDomain` y sus valores se pueden especificar en cualquier secuencia.

¿Son los comandos de EPM Automate sensibles a mayúsculas/minúsculas?

Los comandos de EPM Automate no son sensibles a mayúsculas/minúsculas. La forma de escribir un comando no tiene ningún efecto en la ejecución del comando. Por ejemplo, puede escribir el comando `addUsers` como `addusers`, `ADDUSERS` o `AdDuSeRs`.

¿Distinguen entre mayúsculas y minúsculas los parámetros de comando de EPM Automate?

Los parámetros de comando de EPM Automate no distinguen entre mayúsculas y minúsculas. La forma de escribir un nombre de parámetro de comando no tiene ningún efecto en la ejecución del comando. Por ejemplo, puede escribir el parámetro `FileName` como `filename`, `fileName` o `fIlEnAmE` sin que ello afecte a la ejecución del comando.

Especificación de varios valores para un parámetro

Algunos comandos de EPM Automate aceptan varios valores de parámetro separados por coma; por ejemplo, una petición de datos de tiempo de ejecución de miembros de tipo en reglas de negocio, conjuntos de reglas y plantilla en una aplicación de Planning.

Para establecer más de un miembro para un tipo de miembros de una petición de tiempo de ejecución llamada `Entities` en un comando de EPM Automate, utilice una , (coma) como se muestra en el siguiente ejemplo para ejecutar el comando `runbusinessrule`.

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

Los nombres de miembro que contienen caracteres especiales, como espacio y coma, deben ir entre comillas y se parados mediante \ (barra diagonal inversa), como se muestra en el siguiente ejemplo:

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

Comportamiento durante el mantenimiento diario

No ejecute comandos de EPM Automate mientras haya en curso un mantenimiento diario de un entorno.

El usuario no puede realizar ninguna actividad durante el mantenimiento diario. Si se intentan ejecutar los comandos de EPM Automate, ya sea directamente o con scripts, mientras está en curso el mantenimiento diario, se mostrará el siguiente error:

```
EPMAT-11: Error del servidor interno. Debido al mantenimiento diario,
el entorno del servicio Oracle EPM Cloud no se encuentra disponible
temporalmente.
```

Ejecución de EPM Automate

Utilice sus credenciales de Oracle Fusion Cloud Enterprise Performance Management para iniciar sesión mediante EPM Automate. No se puede iniciar sesión con las credenciales de inicio de sesión único.

Los usuarios de Cloud EPM pueden utilizar las credenciales de dominio de identidad para conectarse a un entorno mediante el uso de EPM Automate. Los roles predefinidos y de aplicación asignados al usuario determinan los comandos que un usuario puede ejecutar.

Además, solo los administradores del servicio pueden ejecutar algunos comandos, mientras que los roles de administrador de dominio de identidad puede que también sean necesarios para ejecutar algunos comandos.

Generación del archivo log de depuración

Oracle Support le solicitará un archivo de registro de depuración de la sesión para solucionar problemas que se han producido al ejecutar EPM Automate. Soporta la opción `-d` para generar mensajes de depuración, que se podrán redireccionar a un archivo mediante la directiva `>`. Puede crear un archivo de depuración para un comando o un archivo de ejecución por lotes o script que contenga varios comandos.

Sintaxis: comando `epmautomate [parámetros_comando] -d > archivo_log 2>&1`

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

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

Windows

Antes de ejecutar EPM Automate, asegúrese de que puede acceder al entorno desde el equipo desde el que está ejecutando.

EPM Automate crea un archivo `.prefs`, que contiene información del usuario y archivos de registro en el directorio actual. En los equipos con Windows, el contenido del archivo `.prefs` solo lo puede ver el usuario que lo ha creado y los administradores de Windows. En entornos de Linux, UNIX y macOSX, se genera el archivo `.prefs` con permiso 600, que solo permite al propietario leer el archivo y escribir en él.

EPM Automate muestra el error `FileNotFoundException: .prefs (Access is denied)` en entornos Windows si no tiene permiso de escritura en el directorio de Windows desde el que ejecuta. Para resolver este error, asegúrese de que la cuenta de Windows del usuario actual tenga acceso de lectura/escritura al directorio desde el que se ejecuta EPM Automate. Además, este usuario debe tener el acceso adecuado a cualquier otro directorio desde el que se acceda a un archivo (por ejemplo, mientras se ejecuta el comando `uploadFile`) o se escriba (por ejemplo, al ejecutar el comando `downloadFile`).

> **Nota:**
>
> No se puede ejecutar EPM Automate desde una carpeta que contiene `&` en su nombre; por ejemplo, `C:\Oracle\A&B`.

Para ejecutar EPM Automate en un cliente de Windows:

1. Haga clic en **Inicio** y, a continuación, en **Todos los programas**, seleccione **EPM Automate** y, por último, **Iniciar EPM Automate**. Aparecerá el símbolo del sistema de EPM Automate.
2. **Opcional:** vaya al directorio desde el que desea realizar operaciones.
3. **Opcional:** Genere un archivo de cifrado de contraseñas. El archivo de cifrado de contraseñas sirve para utilizar una contraseña cifrada al iniciar sesión.

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

4. Inicie una sesión como administrador de servicio. Utilice un comando como el siguiente:
   - Con una contraseña cifrada:

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

   - Con una contraseña no cifrada:

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

5. Introduzca los comandos para ejecutar las tareas que desea realizar.

   Consulte Códigos de salida para obtener información sobre el estado de ejecución de los comandos.
6. Cierre la sesión en el entorno. Utilice el siguiente comando.

   ```
   epmautomate logout
   ```

Linux

> **Nota:**
>
> Asegúrese de que *`JAVA_HOME`* está definido en la variable `PATH` del archivo `.profile` o como una variable de entrono de shell. Se requiere una versión de JRE soportada (de la versión 8 a la 11).

Para ejecutar EPM Automate en un cliente de Linux:

1. Abra una ventana de terminal y vaya al directorio en el que ha instalado EPM Automate.

2. **Opcional:** Genere un archivo de cifrado de contraseñas. El archivo de cifrado de contraseñas sirve para utilizar una contraseña cifrada en lugar de una no cifrada al iniciar sesión.

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

3. Inicie una sesión como administrador de servicio. Utilice un comando como el siguiente:
   - Con una contraseña cifrada:

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

   - Con una contraseña no cifrada:

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

4. Introduzca los comandos para ejecutar las tareas que desea realizar.

   Consulte Códigos de salida para obtener información sobre el estado de ejecución de los comandos.

5. Cierre la sesión en el entorno. Utilice el siguiente comando.

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

Ejecución de varias instancias de EPM Automate

Puede ejecutar varias instancias de EPM Automate con respecto a un entorno del mismo directorio. De manera similar, puede ejecutar varias instancias con respecto a diferentes entornos desde mismo directorio o directorios diferentes.
Por ejemplo, puede que tenga que refrescar de manera simultánea el cubo de la aplicación de Planning en `https://cloudpln.pbcs.us1.oraclecloud.com` y `https://testcloudpln.pbcs.us1.oraclecloud.com`. En este escenario, tiene dos opciones:

- Ejecutar dos instancias de EPM Automate desde el mismo directorio para refrescar los cubos de la aplicación en entornos diferentes.
- Ejecutar EPM Automate desde directorios separados para conectar los entornos y, a continuación, refrescar los cubos de la aplicación.

En ambos escenarios, cada instancia funciona de forma independiente; cerrar sesión en una instancia no significa que se cierre sesión en las otras. Las actividades iniciadas con EPM Automate siguen ejecutándose hasta su finalización en el entorno, incluso si cierra la sesión desde otra instancia.

Esta sección contiene scripts de ejemplo de Windows y Unix/Linux (`caller` y `multisession`) que se pueden usar para crear dos sesiones de EPM Automate para realizar tareas. Para ejecutar varias sesiones simultáneas, debe agregar la siguiente información de conexión en el script `caller`, el cual llama al script `multisession` para ejecutar los comandos `login`, `uploadfile`, `listfiles` y `logout`. Puede modificar el script `multisession` para realizar tareas distintas de estas. Asegúrese de que ambos scripts se almacenan en el mismo directorio.

- EPM Automate utiliza la variable de entorno `EPM_SID` para distinguir las sesiones múltiples. Esta variable debe establecerse en un valor único en el script de emisor de llamada para

cada sesión. En los scripts de ejemplo, se define en valores únicos de la siguiente manera:

- En `caller.BAT`, `EPM_SID` se define en `!RANDOM!`, que lo asigna a un número único generado por el sistema. Este número también se utiliza para generar archivos de registro para cada sesión. Si desea realizar el seguimiento del archivo de registro para cada sesión, puede especificar un número único en lugar de `!RANDOM!`.
- En `caller.sh`, `EPM_SID` se define en el identificador de proceso, que es único. Si desea realizar un seguimiento del archivo de registro para cada sesión, puede especificar un `EPM_SID` único modificando la sentencia `export EPM_SID=$$` en el script `multisession` para que se utilice el valor transferido y, a continuación, se transfiera un valor único para este parámetro en el script `caller` para cada sesión; por ejemplo, especificando el valor de `EPM SID` en `caller.sh` de la siguiente manera:

  ```
  $SCRIPT_DIR/multisession.sh EPM_SID "USERNAME" "PASSWORD" "URL" "/home/
  user/Snapshot1.zip" &
  $SCRIPT_DIR/multisession.sh EPM_SID "USERNAME" "PASSWORD" "URL" "/home/
  user/Snapshot2.zip" &
  ```

- `USERNAME`: ID de conexión de un administrador de servicio
- `PASSWORD`: contraseña del administrador de servicio
- `URL`: URL de conexión del entorno

Scripts de Windows de ejemplo

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
```

Script de shell Bourne de ejemplo

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" &
$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
```

Comandos de un vistazo

Es una lista alfabética de todos los comandos de EPM Automate.

Tabla 2-1 Todos los comandos de EPM Automate

Nombre del comando	PLN, SWP, SP, FF	FCC	TR	PCM	EPCM	AR	EDM	NR
addUsers	✓	✓	✓	✓	✓	✓	✓	✓
addUsersToGroup	✓	✓	✓	✓	✓	✓	✓	✓
addUsersToTeam		✓	✓			✓		
addUserToGroups	✓	✓	✓	✓	✓	✓	✓	✓
applicationAdminMode	✓	✓	✓		✓			
applyDataGrants				✓				
archiveTmTransactions						✓		
assignRole	✓	✓	✓	✓	✓	✓	✓	✓
autoPredict* Véase la nota al pie	✓							
calculateModel					✓			
clearCube	✓				✓			
clearDataByPointOfView					✓			
clearDataByProfile		✓	✓					
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				✓				

Tabla 2-1 (Continuación) Todos los comandos de EPM Automate

Nombre del comando	PLN, SWP, SP, FF	FCC	TR	PCM	EPCM	AR	EDM	NR
enableQueryTracking	✓				✓			
encrypt	✓	✓	✓	✓	✓	✓	✓	✓
essbaseBlockAnalysisReport	✓	✓	✓					
executeAggregationProcess	✓				✓			
executeBurstDefinition								✓
executeReportBurstingDefinition	✓	✓	✓		✓			
exportAccessControl						✓		
exportAppAudit	✓	✓	✓		✓			
exportAppSecurity	✓	✓	✓		✓			
exportARApplicationProperties						✓		
exportBackgroundImage						✓		
exportCellLevelSecurity	✓		✓		✓			
exportData	✓	✓	✓		✓			
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	✓	✓	✓	✓	✓	✓	✓	✓
getRestrictedDataAccess	✓	✓	✓	✓	✓	✓	✓	✓

Tabla 2-1 (Continuación) Todos los comandos de EPM Automate

Nombre del comando	PLN, SWP, SP, FF	FCC	TR	PCM	EPCM	AR	EDM	NR
getSubstVar	✓	✓	✓		✓			
getVirusScanOnFileUploads	✓	✓	✓	✓	✓	✓	✓	✓
groupAssignmentAuditReport	✓	✓	✓	✓	✓	✓	✓	✓
help	✓	✓	✓	✓	✓	✓	✓	✓
importAppAudit	✓				✓			
importAppSecurity	✓	✓	✓		✓			
importARApplicationProperties						✓		
importBackgroundImage						✓		
importBalances						✓		
importCellLevelSecurity	✓		✓		✓			
importConsolidationJournals		✓						
importData	✓	✓	✓	✓	✓			
importDataManagement	✓	✓	✓	✓	✓			
importDimension							✓	
importJobConsole	✓	✓	✓		✓			
importLibraryArtifact								✓
importLogoImage						✓		
importMapping	✓	✓	✓	✓	✓	✓		
importMetadata	✓	✓	✓		✓			
importOwnershipData		✓	✓					
importPreMappedBalances						✓		
importPreMappedTransactions						✓		
importProfiles						✓		
importRates						✓		
importRCAttributeValues						✓		
importReconciliationAttributes						✓		
importSnapshot	✓	✓	✓	✓	✓	✓	✓	
importSupplementalCollectionData		✓	✓					
importSupplementalData		✓	✓					
importTemplate				✓				
importTMAttributeValues						✓		
importValidIntersections	✓	✓	✓		✓			
invalidLoginReport	✓	✓	✓	✓	✓	✓	✓	✓
listBackups	✓	✓	✓	✓	✓	✓	✓	✓
listFiles	✓	✓	✓	✓	✓	✓	✓	✓
loadData				✓				
loadDimData				✓				
loadDimensionViewpoint							✓	
loadViewpoint							✓	
login	✓	✓	✓	✓	✓	✓	✓	✓

Tabla 2-1 (Continuación) Todos los comandos de EPM Automate

Nombre del comando	PLN, SWP, SP, FF	FCC	TR	PCM	EPCM	AR	EDM	NR
logout	✓	✓	✓	✓	✓	✓	✓	✓
maskData	✓	✓	✓		✓			
mergeDataSlices	✓				✓			
mergeSlices				✓				
optimizeASOCube				✓				
programDocumentationReport				✓				
provisionReport	✓	✓	✓	✓	✓	✓	✓	✓
purgeArchivedTmTransactions						✓		
purgeTmTransactions						✓		
recomputeOwnershipData		✓	✓					
recreate	✓	✓	✓	✓	✓	✓	✓	✓
refreshCube	✓	✓	✓		✓			
removeUserFromGroups	✓	✓	✓	✓	✓	✓	✓	✓
removeUsers	✓	✓	✓	✓	✓	✓	✓	✓
removeUsersFromGroup	✓	✓	✓	✓	✓	✓	✓	✓
removeUsersFromTeam		✓	✓			✓		
renameSnapshot	✓	✓	✓	✓	✓	✓	✓	
replay	✓	✓	✓	✓	✓	✓	✓	✓
resetService	✓	✓	✓	✓	✓	✓	✓	✓
restoreBackup	✓	✓	✓	✓	✓	✓	✓	✓
restructureCube	✓	✓	✓					
roleAssignmentAuditReport	✓	✓	✓	✓	✓	✓	✓	✓
roleAssignmentReport	✓	✓	✓	✓	✓	✓	✓	✓
runAutomatch						✓		
runBatch	✓	✓	✓	✓	✓	✓		
runBusinessRule	✓	✓	✓		✓			
runCalc				✓				
runComplianceReport						✓		
runDailyMaintenance	✓	✓	✓	✓	✓	✓	✓	✓
runDataRule	✓	✓	✓	✓	✓	✓		
runDMReport	✓	✓	✓	✓	✓	✓		
runIntegration	✓	✓	✓	✓	✓	✓		
runIntercompanyMatchingReport		✓						
runMatchingReport						✓		
runPipeline	✓	✓	✓		✓			
runPlanTypeMap	✓							
runRuleSet	✓	✓	✓		✓			
runSupplementalDataReport		✓	✓					
runTaskManagerReport		✓	✓		✓			
sendMail	✓	✓	✓	✓	✓	✓	✓	✓

Tabla 2-1 (Continuación) Todos los comandos de EPM Automate

Nombre del comando	PLN, SWP, SP, FF	FCC	TR	PCM	EPCM	AR	EDM	NR
setApplicationAdminMode	✓	✓	✓		✓	✓		
setDailyMaintenanceStartTime	✓	✓	✓	✓	✓	✓	✓	✓
setDemoDates		✓	✓		✓	✓		
setEJJournalStatus		✓						
setEncryptionKey	✓	✓	✓	✓	✓	✓	✓	✓
setEssbaseQryGovExecTime	✓	✓	✓	✓	✓			
setIdleSessionTimeout	✓	✓	✓	✓	✓	✓	✓	✓
setIPAllowlist	✓	✓	✓	✓	✓	✓	✓	✓
setManualDataAccess	✓	✓	✓	✓	✓	✓	✓	✓
setPeriodStatus						✓		
setRestrictedDataAccess	✓	✓	✓	✓	✓	✓	✓	✓
setSubstVars	✓	✓	✓		✓			
setVirusScanOnFileUploads	✓	✓	✓	✓	✓	✓	✓	✓
simulateConcurrentUsage	✓	✓	✓					
skipUpdate	✓	✓	✓	✓	✓	✓	✓	✓
snapshotCompareReport	✓	✓	✓		✓			
sortMember	✓				✓			
unassignRole	✓	✓	✓	✓	✓	✓	✓	✓
updateGuidedLearningSettings	✓	✓	✓		✓			
updateUsers	✓	✓	✓	✓	✓	✓	✓	✓
upgrade	✓	✓	✓	✓	✓	✓	✓	✓
uploadFile	✓	✓	✓	✓	✓	✓	✓	✓
userAuditReport	✓	✓	✓	✓	✓	✓	✓	✓
userGroupReport	✓	✓	✓	✓	✓	✓	✓	✓
validateConsolidationMetadata		✓						
validateModel					✓			

- * Este comando solo está soportado si los cubos híbridos de Oracle Essbase están activados en la aplicación. Strategic Workforce Planning y Sales Planning no soportan Essbase híbrido. Este comando no está soportado para FreeForm.
- ** Este comando no está soportado para Formato libre.

Abreviaturas

- PLN: Planning (incluido Planning Modules)
- 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

Comandos de EPM Automate

En esta sección se detalla cada comando de EPM Automate. La información disponible para cada comando incluye los servicios que pueden utilizar el comando, el uso del comando y un ejemplo.

addUsers

Crea un lote de usuarios en un dominio de identidad utilizando un archivo de valores separados por comas (.csv) codificado por ANSI o UTF-8 que se ha cargado en el entorno. También informa a los nuevos usuarios de su nombre de usuario y contraseña temporal.

Un administrador del servicio utiliza el comando uploadFile para cargar archivos en un entorno. Todas las columnas del archivo CSV son obligatorias. Este comando valida el valor de cada columna de una definición y muestra mensajes de error que identifica cada uno de los valores que faltan o no válidos. El formato del archivo CSV es el siguiente:

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

Consulte Importación de un lote de cuentas de usuarios en *Introducción a Oracle Cloud* para una descripción detallada del formato de archivo CSV.

El valor de `User Login` especificado en el archivo de importación no distingue mayúsculas y minúsculas. Por ejemplo, el valor `John.doe@example.com` se considera que es idéntico a `John.Doe@example.com` o a cualquier variación de sus mayúsculas o minúsculas.

Si una definición de usuario en el archivo CSV coincide con una cuenta de usuario que existe en el dominio de identidad, no se realizarán cambios en la cuenta de usuario existente. Este comando crea cuentas solo para nuevos usuarios cuya información de cuenta se incluye en el archivo. Dado que las cuentas de usuario son comunes a todos los entornos que soporta un dominio de identidad, los nuevos usuarios están disponibles para todos los entornos que comparten el dominio de identidad.

Cuando finalice, el comando imprime información sobre cada una de las entradas con fallos en la consola. Consulte esta información para saber el motivo por el que se ha producido un fallo en la ejecución del comando en algunas entradas del archivo CSV.

Se aplica a

Planning, Planning Modules, 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 y Sales Planning.

Roles necesarios

Administrador de dominio de identidad y cualquier rol predefinido

Sintaxis

`epmautomate addUsers FILE_NAME [userPassword=PASSWORD] [resetPassword=true|false]`, donde:

- *FILE_NAME* es el nombre de un archivo CSV que contiene información del usuario. El archivo de entrada que contiene caracteres de varios bytes debe utilizar la codificación de caracteres UTF-8.
- `userPassword`, opcionalmente, indica la contraseña predeterminada para todos los nuevos usuarios que se crean en el dominio de identidad. Si se especifica, esta contraseña debe cumplir los requisitos mínimos para contraseñas de dominios de identidad. Si no se especifica el parámetro, se asigna una contraseña temporal única a cada usuario.
  Si se especifica, se usa el valor del parámetro `userPassword` como contraseña para todos los usuarios especificados en el archivo CSV. La asignación de la misma contraseña a todos los usuarios puede ser aconsejable si desea crear usuarios solo con fines de prueba. Si está creando usuarios reales de Oracle Fusion Cloud Enterprise Performance Management y desea asignar una contraseña específica a cada uno de los usuarios, use este comando sin especificar un valor para el parámetro opcional `userPassword`.
- `resetPassword`, de forma opcional, indica si los nuevos usuarios deben cambiar la contraseña la primera vez que inician sesión. El valor predeterminado es `true`. A menos que este parámetro se defina en `false`, los nuevos usuarios se verán obligados a cambiar la contraseña la primera vez que inicien sesión.
  Este comando envía a cada usuario nuevo un correo electrónico con detalles sobre sus cuentas (nombre de usuario y contraseña) si `resetPassword` está establecido en `true`. Si `resetPassword` se establece en `false`, el correo electrónico no se envía. Si establece `resetPassword` en false, debe especificar `userPassword`. De lo contrario, se asignará una contraseña temporal única a cada usuario, pero como no se envía ningún correo electrónico, los usuarios no sabrán las contraseñas y no podrán iniciar sesión.

Ejemplos

- Agregue usuarios de prueba en el dominio de identidad con la misma contraseña sin requerir que cambien la contraseña:
  ```
  epmautomate addUsers user_file.CSV userPassword=Example@Pwd12
  resetPassword=false
  ```
- Agregue usuarios al domino de identidad con una contraseña temporal y pídales que la cambien:
  ```
  epmautomate addUsers user_file.CSV
  ```

addUsersToGroup

Agrega un lote de usuarios a un grupo existente en Control de acceso mediante el uso de un archivo CSV con codificación ANSI o UTF-8 que se haya cargado en el entorno.

Un administrador del servicio utiliza el comando uploadFile para cargar archivos en un entorno. El valor de User Login no distingue mayúsculas y minúsculas. El formato del archivo es el siguiente:

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

> **Nota:**
>
> El usuario se agrega a un grupo solo si se cumplen las condiciones siguientes:
>
> - El valor de User Login incluido en el archivo existe en el dominio de identidad que presta servicio al entorno. Los valores de User Login no distinguen mayúsculas y minúsculas.
> - El usuario tiene asignado un rol predefinido en el dominio de identidad.

Cuando finalice, el comando imprime información sobre cada una de las entradas con fallos en la consola. Consulte esta información para saber el motivo por el que se ha producido un fallo en la ejecución del comando en algunas entradas del archivo CSV.

Se aplica a

Planning, Planning Modules, 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 y Sales Planning.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Gestionar

Sintaxis

`epmautomate addUsersToGroup` *`FILE_NAME GROUP_NAME`* donde:

- *`FILE_NAME`* es el nombre de un archivo CSV que contiene los nombres de inicio de sesión de los usuarios que desea asignar a un grupo en Control de acceso.
- *`NOMBRE_GRUPO`* es el nombre del grupo existente en Control de acceso. Este valor no distingue mayúsculas y minúsculas.

Ejemplo

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

addUsersToTeam

Agrega los usuarios de Oracle Fusion Cloud Enterprise Performance Management que se muestran en un archivo CSV a un equipo existente.

Si un usuario incluido en el archivo CSV ya es miembro del equipo, con este comando se ignora al usuario. Los valores de este archivo no distinguen entre mayúsculas y minúsculas. Un administrador del servicio utiliza el comando uploadFile para cargar este archivo en el entorno. El formato del archivo CSV es el siguiente:

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

> **Nota:**
>
> De forma predeterminada, se designa un usuario primario para realizar las tareas asignadas al equipo.

Se aplica a

Financial Consolidation and Close, Tax Reporting y Account Reconciliation.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido y el rol de aplicación Equipos - Gestionar
- Cualquier rol predefinido y el rol de aplicación Usuarios - Gestionar

Sintaxis

`epmautomate addUsersToTeam` *`FILE TEAM_NAME`* donde:

- `FILE` identifica un archivo CSV con formato UTF-8 que muestra los identificadores de inicio de sesión de los usuarios para agregarlos al equipo. Antes de ejecutar este comando, utilice el comando uploadFile para cargar archivos en un entorno.
- `NOMBRE_EQUIPO` identifica un nombre de equipo según la definición de Control de acceso. Este valor no distingue mayúsculas y minúsculas.

Ejemplo

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

addUserToGroups

Agrega un usuario como un miembro de los grupos de Control de acceso identificados en un archivo CSV ANSI o UTF-8 codificado.

Un administrador del servicio utiliza el comando uploadFile para cargar archivos en un entorno. El formato del archivo es el siguiente:

```
Group Name
Group1
Group2
```

Los valores de nombres de grupos no distinguen mayúsculas y minúsculas.

Cuando finalice, el comando imprime información sobre cada una de las entradas con fallos en la consola. Consulte esta información para saber el motivo por el que se ha producido un fallo en la ejecución del comando en algunas entradas del archivo CSV.

Se aplica a

Planning, Planning Modules, 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 y Sales Planning.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Gestionar

Sintaxis

`epmautomate addUserToGroups FILE_NAME User_Login`, donde:

- `FILE_NAME` es el nombre de un archivo CSV que contiene los nombres de los grupos de Control de acceso a los que desea asignar el usuario.
- `User_Login` es el ID de inicio de sesión de un usuario de Oracle Fusion Cloud Enterprise Performance Management que se va asignar a grupos de Control de acceso. Este identificador de inicio de sesión de usuario, que no distingue mayúsculas y minúsculas, debe existir en el dominio de identidad que presta servicio al entorno y se debe asignar a un rol predefinido.

Ejemplo

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

applicationAdminMode

Pone la aplicación en modo de administración, de forma que se limita el acceso a la aplicación solo a administradores de servicio.

Este comando es útil para evitar que los usuarios trabajen en la aplicación cuando los administradores de servicio realizan operaciones administrativas. La aplicación permanece en el modo de administración hasta que se cambie para que todos los usuarios puedan acceder.

> **Nota:**
>
> Este comando está anticuado, pero no se ha eliminado de EPM Automate. Oracle le recomienda utilizar el comando setApplicationAdminMode en su lugar.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate applicationAdminMode VALUE`, donde `VALUE` especifica si se debe poner la aplicación en el modo de administración. Los valores aceptables son los siguientes:

- `true` para poner la aplicación en modo de administración.
- `false` para volver a poner la aplicación en modo normal de forma que todos los usuarios puedan acceder a ella.

Ejemplos

- Poner la aplicación en modo de administración: `epmautomate applicationAdminMode true`
- Devolver la aplicación a su funcionamiento normal: `epmautomate applicationAdminMode false`

applyDataGrants

Refresca los otorgamientos de datos, que controlan el acceso a los segmentos de datos de Oracle Essbase, para que coincidan con los otorgamientos de datos definidos en una aplicación de Profitability and Cost Management.

Los otorgamientos de datos de nivel de grupo y usuario que realice en la aplicación de Profitability and Cost Management se sincronizan de forma automática en Essbase. Utilice este comando para sincronizar el acceso a datos de Essbase si sospecha que hay discordancia entre los otorgamientos de datos en la aplicación y los filtros en Essbase.

El tiempo necesario para realizar esta operación dependerá del tamaño de la aplicación. Asegúrese de que la operación de refrescamiento de los otorgamientos de datos termina antes de que se realice una copia de seguridad de la aplicación durante el próximo periodo de mantenimiento. Debido a que la aplicación no se debe utilizar mientras esta operación está en curso, Oracle recomienda programar esta operación en un momento en el que los usuarios no estén trabajando con la aplicación.

Se aplica a

Profitability and Cost Management

Roles necesarios

Administrador del servicio, Usuario avanzado

Sintaxis

`epmautomate applyDataGrants` *`APPLICATION_NAME`* donde *`APPLICATION_NAME`* es el nombre de la aplicación de Profitability and Cost Management para la que se van a volver a crear otorgamientos de datos.

Ejemplo

```
epmautomate applyDataGrants BksML12
```

archiveTmTransactions

Archiva las transacciones coincidentes, incluidos los detalles de ajuste y soporte, que son iguales anteriores a una antigüedad especificada. Las transacciones coincidentes se registran en un archivo ZIP.

Utilice este comando para guardar el tamaño de aplicación de Account Reconciliation óptimo para archivar y, posteriormente, depurar las transacciones coincidentes antiguas según las políticas de retención de transacciones de la organización.

Se aplica a

Account Reconciliation

Roles necesarios

Administrador del servicio, Usuario avanzado, Usuario, Visor
Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

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

- `matchType` es el identificador (`TextID`) del tipo de coincidencia para el que se deben archivar las transacciones coincidentes.
- `age` identifica el número de días desde la confrontación de la transacción. Se archivará una transacción coincidente anterior o igual a este valor.
- Opcionalmente, `filterOperator`, es una de las siguientes condiciones de filtro para identificar las cuentas que contienen transacciones coincidentes para su archivado. Este valor se combina con `filterValue` para identificar las cuentas de las que deben archivarse las transacciones coincidentes:
  - `equals`
  - `not_equals`
  - `starts_with`
  - `ends_with`
  - `contains`
  - `not_contains`
- Opcionalmente, `filterValue`, es un valor de filtro para identificar las transacciones que se deben archivar. Si `filterOperator` tiene el valor `equals` o `not_equals`, puede usar una lista separada por comas para especificar varios valores; por ejemplo, `filterValue=101-120 filterValue=102-202`. Si se especifican varios valores, se seleccionan las transacciones de las cuentas que coincidan con cualquier combinación de valor de filtro y operador de filtro para su archivado.

  > **Note:**
  >
  > Si no se especifican `filterOperator` y `filterValue`, se archivan todas las transacciones coincidentes anteriores o iguales al valor `age` de todas las cuentas para el valor `matchType` especificado.

- `logFilename`, opcionalmente, es el nombre de un archivo log para registrar la información sobre la actividad del comando. Si no se especifica un nombre de archivo, se genera de forma automática un archivo de registro denominado `Archive_Transactions_matchType_JOBID.log`.
- Opcionalmente, `filename`, es el nombre de un archivo .ZIP que debe contener las transacciones archivadas. Si no se especifica, el comando, de forma predeterminada, crea `Archived_Transactions_matchType_JOBID.zip`. Utilice el comando downloadFile para descargar este archivo en un equipo local.

> **Note:**
>
> Este comando ejecuta el trabajo de archivado de confrontación de transacciones utilizando los parámetros que especifique. El identificador del trabajo se devuelve en la salida del comando para facilitar su uso con el comando purgeArchivedTmTransactions. Puede supervisar el trabajo desde la consola de trabajos.

Ejemplos

- Archive las transacciones coincidentes antiguas sin utilizar filtros, pero usando un registro personalizado y un nombre de archivo .ZIP:
  ```
  epmautomate archiveTmTransactions cashrecon 180 logFile=tmlogs.log
  filename=trans.zip
  ```
- Archive las transacciones coincidentes antiguas utilizando filtros:
  - `epmautomate archiveTmTransactions cashrecon 180 filterOperator=equals filterValue=101-120 FilterValue=102-202`
  - `epmautomate archiveTmTransactions cashrecon 180 filterOperator=contains filterValue=11`

assignRole

Asigna un rol a los usuarios (incluido el usuario que ejecuta este comando), cuyos ID de inicio de sesión se incluyen en el archivo CSV con codificación ANSI o UTF-8. Utilice este comando para asignar usuarios a un rol predefinido o un rol de aplicación.

Antes de utilizar este comando, un administrador del servicio utiliza el comando uploadFile para cargar el archivo de entrada en un entorno. El formato del archivo es el siguiente:

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

Consulte Asignación de un rol a muchos usuarios en *Introducción a Oracle Cloud*.

> **Nota:**
>
> - Los valores de User Login del archivo no distinguen mayúsculas y minúsculas.
> - Utilice las comillas dobles para incluir los nombres de rol que contienen caracteres de espacio.

Cuando finalice, el comando imprime información sobre cada una de las entradas con fallos en la consola. Consulte esta información para saber el motivo por el que se ha producido un fallo en la ejecución del comando en algunas entradas del archivo CSV.

Se aplica a

Planning, Planning Modules, 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 y Sales Planning.

Roles necesarios

Para asignar roles predefinidos:

- Para asignar roles predefinidos:
  - Administrador del servicio
  - Administrador de dominio de identidad y cualquier rol predefinido
- Para asignar roles de aplicación:
  - Administrador del servicio
  - Cualquier rol predefinido y el rol de aplicación Control de acceso - Gestionar

Sintaxis

`epmautomate assignRole` *`FILE_NAME ROLE`* donde:

- *`FILE_NAME`* es el nombre de un archivo CSV que contiene ID de inicio de sesión de usuario. Especifique la extensión CSV en minúsculas.
- `ROLE` es una de las siguientes opciones. Este valor no distingue mayúsculas y minúsculas:
  - Si asigna usuarios a roles de dominio de identidad predefinidos, `ROLE` debe identificar un rol predefinido aplicable al servicio. Consulte Descripción de los roles predefinidos en *Guía de introducción para administradores*.
    Para obtener una descripción de estos roles, consulte Gestión de asignaciones de roles en el nivel de aplicación en *Administración del control de acceso*
  - Si está asignando usuarios a roles de aplicación, `ROLE` debe identificar un rol que pertenezca a la aplicación en el entorno actual. Los roles de aplicación se enumeran en el separador **Roles** de Control de acceso. Para obtener una descripción de los roles de aplicación para cada proceso de negocio, consulte estos temas en *Administración del control de acceso para Oracle Enterprise Performance Management Cloud*:
    - Account Reconciliation
    - Enterprise Profitability and Cost Management
    - Planning, FreeForm, Financial Consolidation and Close y Tax Reporting
    - Profitability and Cost Management
    - Oracle Enterprise Data Management
    - Narrative Reporting

Ejemplos

- Asigne los usuarios a un rol predefinido del dominio de identidad:
  ```
  epmautomate assignRole admin_role_file.csv "Service Administrator"
  ```
- Asigne usuarios a un rol de aplicación:
  ```
  epmautomate assignRole example_file.csv "Task List Access Manager"
  ```

autoPredict

Genera predicciones de rendimiento futuro según una definición de Predicción automática existente en Planning o en Planning Modules.

Este comando inicia un trabajo que utiliza los datos históricos de cada miembro identificado en la definición de Predicción automática especificada en la aplicación. Para obtener información detallada sobre las aplicaciones que utilizan la función Predicción automática y configurar las predicciones, consulte Configuración de predicciones para su ejecución automática con la función Predicción automática en *Administración de Planning*.

Se aplica a

Planning, Planning Modules, si están activados los cubos híbridos de Oracle Essbase en la aplicación.

Roles necesarios

Administrador del servicio

Sintaxis

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

- *`PREDICTION_DEFINITION`* es el nombre de la definición de predicción automática disponible en la aplicación.
- Opcionalmente, `forceRun` especifica si se ejecuta la predicción en el caso de que la definición subyacente no haya cambiado después de la ejecución inicial. El valor predeterminado es `false`
  Defina el valor de este parámetro en `true` para ejecutar el trabajo de predicción automática incluso si no se ha producido ningún cambio en la definición de trabajo. Utilice el valore predeterminado (`false`) para ejecutar la predicción una vez la primera vez que se ejecuta el trabajo.
- Opcionalmente, `paginatedDim` especifica una dimensión que se va a utilizar para acelerar el trabajo de predicción automática mediante la ejecución de predicciones en paralelo en subprocesos independientes. Para que estos subprocesos paralelos sean eficientes, especifique una dimensión que provocará datos con una difusión uniforme para cada subproceso de la predicción.

Ejemplo

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

calculateModel

Ejecuta un proceso de cálculo en las aplicaciones deEnterprise Profitability and Cost Management.

Se aplica a

Enterprise Profitability and Cost Management

Roles necesarios

Administrador del servicio

Sintaxis

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

- *POV_NAME* es el nombre del PDV de datos que se va a calcular. Para calcular varios PDV, muestre los nombres de PDV separados por una coma como delimitador. No utilice ningún otro delimitador para separar nombres de PDV. Incluya la lista de nombres de PDV entre comillas dobles cuando haya espacios en los nombres de miembro.
- *MODEL_NAME* es el nombre del modelo que se va a calcular. Incluya el nombre de modelo entre comillas dobles si el nombre contiene espacios.
- *EXECUTION_TYPE* es uno de los siguientes, que identifica el tipo de ejecución de la regla.
  - `ALL_RULES` para usar todas las reglas para calcular el PDV.
    Si especifica este valor, no especifique parámetros de tiempo de ejecución relacionados con una sola regla o subconjunto de reglas como `rulesetSeqNumStart`, `rulesetSeqNumEnd` y `ruleName`.
  - `RULESET_SUBSET` para usar un subconjunto de un conjunto de reglas para calcular el PDV.
    Si usa este valor, debe especificar los valores `rulesetSeqNumStart` y `rulesetSeqNumEnd` como parámetros de tiempo de ejecución.
  - `SINGLE_RULE` para ejecutar una regla específica para calcular el PDV.
    Si usa este valor, solo debe especificar un valor `ruleName` como parámetro de tiempo de ejecución.
  - `RUN_FROM_RULE` para ejecutar cálculos en un PDV a partir de una regla específica.
    Si usa este valor, solo debe especificar un valor `ruleName` como parámetro de tiempo de ejecución.
  - `STOP_AFTER_RULE` para dejar de calcular el PDV después de que una regla específica haya terminado los cálculos.
    Si usa este valor, solo debe especificar un valor `ruleName` como parámetro de tiempo de ejecución.
- `povDelimiter`, opcionalmente, es el delimitador utilizado en los valores de PDV. El delimitador predeterminado es :: (dos puntos dobles). El delimitador debe ir entre comillas dobles. Solo están soportados estos delimitadores:
  - _ (carácter de subrayado)
  - # (almohadilla)
  - & (ampersand)
  - ~ (virgulilla)
  - % (porcentaje)
  - ; (punto y coma)
  - : (dos puntos)
  - - (guion)
- `optimizeForReporting=true|false`, opcionalmente, especifica si los cálculos se van a realizar con o sin optimización para los informes. El valor predeterminado es `false`. Establezca este valor en `false` para ahorrar tiempo de procesamiento omitiendo el paso de creación de agregación; por ejemplo, al ejecutar una sola regla o una serie secuencial de PDV. Cuando ejecute varios trabajos de cálculo simultáneos, establezca `optimizeForReporting=true` para todos los trabajos, para que solo el último trabajo en finalizar realice la agregación, evitando el procesamiento redundante e impidiendo que los trabajos en ejecución se ralenticen.

- `captureDebugScripts=true|false`, opcionalmente, identifica si se van a generar scripts de depuración en la bandeja de entrada. Puede que Oracle necesite estos scripts para solucionar los problemas de cálculo. El valor predeterminado es `false`.
- `comment="COMMENT"`, opcionalmente, especifica un comentario sobre el proceso entre comillas dobles.
- `PARAMETER=VALUE`, opcionalmente, indica los parámetros de tiempo de ejecución y sus valores para ejecutar el cálculo. Especifique tantos pares de parámetro y valor como requiera el proceso. Parámetros válidos y sus valores:
  - `rulesetSeqNumStart` es el número de secuencia de la primera regla del conjunto de reglas que se va a ejecutar. Solo válido si se ha usado `EXECUTION_TYPE=RULESET_SUBSET`.
  - `rulesetSeqNumEnd` especifica el número de secuencia de la última regla del conjunto de reglas que se va a ejecutar. Solo válido si se ha usado `EXECUTION_TYPE=RULESET_SUBSET`.
  - `ruleName` es el nombre de la regla que se va a ejecutar. Incluya el valor entre comillas si contiene el carácter de espacio. Válido solo si el valor de `EXECUTION_TYPE` se establece en `SINGLE_RULE`, `RUN_FROM_RULE` o `STOP_AFTER_RULE`.
  - `clearCalculatedData=true|false` especifica si se van a borrar los cálculos existentes. El valor predeterminado es `false`.
  - `executeCalculations=true|false` especifica si se van a realizar cálculos. El valor predeterminado es `false`.

> **Note:**
>
> Los valores de parámetros (`true` y `false`) deben ir en minúscula.

Ejemplos

- Ejecutar todas las reglas para calcular un solo PDV:
```
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"
```
- Ejecutar todas las reglas para calcular varios PDV utilizando un delimitador personalizado:
```
epmautomate calculateModel
"FY22_Jan_Actual_Working,FY22_Feb_Actual_Working,FY22_Mar_Actual_Working" "10
Actuals Allocation Process" ALL_RULES clearCalculatedData=true
executeCalculations=true optimizeForReporting=true captureDebugScripts=true
comment="Runing calculation for multiple POVs" povDelimiter="_"
```
- Ejecutar un rango de conjuntos de reglas para calcular 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"
```
- Ejecutar una regla específica para calcular 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

Suprime los datos específicos de los cubos de entrada y de informes utilizando la configuración especificada en un trabajo de tipo `clear cube`.

Con este comando no se suprime la definición de aplicación en las tablas relacionales de la aplicación. Consulte Borrado de cubos en *Administración de Planning*.

Se aplica a

Planning, Planning Modules, FreeForm, Enterprise Profitability and Cost Management, Strategic Workforce Planning y Sales Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate clearCube` *`JOB_NAME`* donde: *`JOB_NAME`* es el nombre de un trabajo definido en la aplicación.

Ejemplo

```
epmautomate clearCube ClearPlan1
```

clearDataByPointOfView

Borra los datos de un PDV específico para un cubo de Enterprise Profitability and Cost Management.

Se aplica a

Enterprise Profitability and Cost Management

Roles necesarios

Administrador del servicio

Sintaxis

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

- *`POV_NAME`* es el nombre de un PDV de la aplicación.
- `cubeName`, opcionalmente, es el nombre del cubo en el que se van a borrar datos. El valor predeterminado es `PCM_CLC`.
- `PARAMETER=`*`VALUE`* indica los parámetros de tiempo de ejecución opcionales y sus valores. Especifique tantos pares de parámetro y valor como requiera el proceso. Parámetros válidos y sus valores:
  - `povDelimiter` es el delimitador utilizado en los valores de PDV. El valor predeterminado es `::` (dos puntos dobles). Este valor se debe escribir entre comillas dobles. Ejemplo: `povDelimiter="_"`.

Además de los valores predeterminados, solo están soportados estos delimitadores: _ (carácter de subrayado), # (almohadilla), & (ampersand), ~ (virgulilla), % (porcentaje), ; (punto y coma), : (dos puntos), - (guion).

– `clearInput=true|false` especifica si se van a borrar los datos de entrada. El valor predeterminado es `false`.

– `clearAllocatedValues=true|false` especifica si se van a borrar los valores asignados. El valor predeterminado es `false`.

– `clearAdjustmentValues=true|false` especifica si se van a borrar los valores de ajuste. El valor predeterminado es `false`.

> **Note:**
>
> * Los valores de parámetros (`true` o `false`) deben ir en minúscula.
> * Al menos uno de los parámetros `clearInput`, `clearAllocatedValues` o `clearAdjustmentValues` se debe establecer en `true`.

Ejemplos

- Borrar datos de un PDV del cubo PCM_CLC predeterminado con el delimitador de PDV predeterminado:
```
epmAutomate clearDataByPointOfView FY21::Jan::Actual::Working clearInput=true
clearAllocatedValues=true clearAdjustmentValues=true
```
- Borrar los datos de entrada y los valores asignados de un PDV de un cubo específico mediante un delimitador de PDV personalizado:
```
epmAutomate clearDataByPointOfView FY21_Jan_Actual_Working cubeName=PCM_REP
povDelimiter="_" clearInput=true clearAllocatedValues=true
```
- Borrar los datos de entrada de un PDV de un cubo específico mediante un delimitador de PDV personalizado:
```
epmAutomate clearDataByPointOfView FY21:Jan:Actual:Working cubeName=PCM_REP
povDelimiter=":" clearInput=true
```

clearDataByProfile

Borra datos de los elementos (por ejemplo, regiones) identificados en un perfil de borrado de datos definido en Financial Consolidation and Close y Tax Reporting.

Se aplica a

Financial Consolidation and Close y Tax Reporting

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate clearDataByProfile` *`PROFILE_NAME`* donde *`PROFILE_NAME`* es el nombre de un perfil de borrado de datos.

Ejemplo

```
epmautomate clearDataByProfile clearDataProfile_01
```

clearPOV

Borra los artefactos del modelo y los datos desde una combinación de punto de vista (PDV) o una región de datos dentro del PDV en una aplicación de Profitability and Cost Management.

Se aplica a

Profitability and Cost Management

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

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

- *`APPLICATION_NAME`* es el nombre de una aplicación de Profitability and Cost Management
- *`POV_NAME`* es un PDV de la aplicación. Este valor es necesario.
- `QUERY_NAME`, de forma opcional, es el nombre de una consulta exactamente como se define en Profitability and Cost Management. Si se especifica, esta consulta se utilizará para borrar la región de datos dentro del PDV.

> **Nota:**
>
> Si especifica un nombre de consulta, debe establecer el valor de todos los parámetros de tiempo de ejecución (véase más adelante) en false.

- `PARAMETER=`*`VALUE`* indica los parámetros de tiempo de ejecución y sus valores para borrar el PDV. Especifique tantos pares de parámetro y valor como requiera el proceso. Parámetros válidos, al menos uno de los cuales es necesario y sus valores:
  - `isManageRule=true|false` especifica si se van a borrar las reglas.
  - `isInputData=true|false` especifica si se van a borrar los datos de entrada
  - `isAllocatedValues=true|false` especifica si se van a borrar los valores de asignación
  - `isAdjustmentValues=true|false` especifica si se van a borrar los valores de ajuste

> **Nota:**
>
> Los valores de parámetros (`true` o `false`) deben ir en minúscula.
>
> Para borrar regiones de datos en un PDV (si se especifica un `NOMBRE_CONSULTA`), debe establecer el valor de los parámetros de tiempo de ejecución (`isManageRule`, `isInputData`, `isAllocatedValues` e `isAdjustmentValues`) en `false`.

- `stringDelimiter="DELIMITER"` especifica el delimitador utilizado en los valores de PDV. El delimitador debe ir entre comillas dobles. El valor predeterminado es _ (guion bajo)

Ejemplos

- Borrar todos los artefactos y los datos del modelo de un PDV: `epmautomate clearPOV BksML12 2012_Jan_Actual isManageRule=true isInputData=true isAllocatedValues=true isAdjustmentValues=true stringDelimiter="_"`
- Borrar regiones de datos en 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 el entorno actual y, opcionalmente, los artefactos del dominio de identidad (asignaciones de usuarios y roles predefinidos), los registros de Data Management, los registros de auditoría, los registros de la consola de trabajos, las propiedades de la aplicación, el contenido de la bandeja de entrada y del buzón de salida y las instantáneas almacenadas. Este comando es una alternativa al uso de la función Clonar entorno en Migración.

Inicie la clonación tras el mantenimiento diario programado de los entornos de origen y de destino. Si el mantenimiento diario del entorno de origen de inicia mientras la clonación está en curso, el proceso de clonación se terminará. El proceso de clonación del entorno de destino no se ve afectado incluso si la clonación está en curso a la hora de inicio del mantenimiento diario. En este escenario, el mantenimiento diario se ejecutará una vez finalizada la clonación.

Si la clonación del entorno tarda mucho tiempo, vuelva a programar la hora de inicio del mantenimiento diario en el entorno de origen para evitar que termine el proceso de clonación. Consulte las siguientes fuentes de información para obtener información sobre el restablecimiento de la hora de inicio del mantenimiento diario.

- setDailyMaintenanceStartTime
- Gestión del mantenimiento diario en la *Guía de introducción para administradores*.
- Visualización y establecimiento de la hora de la ventana de mantenimiento diario en la guía *API de REST*.

Note:

- **Data Management:** la clonación de registros de Data Management puede tardar mucho tiempo si las tablas temporales contienen una gran cantidad de registros. Del mismo modo, la clonación del contenido del buzón y de la bandeja de salida, así como de las instantáneas almacenadas, puede llevar mucho tiempo, especialmente si contienen una gran cantidad de datos.
- **Entornos heredados:** la clonación mantiene la versión actual de Oracle Essbase como se describe en estos escenarios:
  - Escenario 1: está clonando un entorno heredado de origen que usa una versión de Essbase que no soporta cubos híbridos para un entorno heredado de destino que usa una versión de Essbase que sí soporta cubos híbridos. En este caso, la versión de Essbase en el entorno de destino se devuelve a una versión anterior para que coincida con la versión del entorno de origen.
  - Escenario 2: está clonando un entorno heredado de origen que usa una versión de Essbase que sí soporta cubos híbridos para un entorno heredado de destino que usa una versión de Essbase que no soporta cubos híbridos. En este caso, la versión de Essbase en el entorno de destino se actualiza a una versión posterior para que coincida con la versión del entorno de origen.
  - Escenario 3: está clonando un entorno heredado de origen que usa una versión de Essbase que no soporta cubos híbridos para un entorno de EPM Standard Cloud Service o EPM Enterprise Cloud Service de destino que usa una versión de Essbase que sí soporta cubos híbridos de manera predeterminada. En este caso, la versión de Essbase en el entorno de destino no se devuelve a una versión anterior para que coincida con la versión del entorno de origen.
- **Planning:** la clonación puede fallar si el proceso de negocio de Planning contiene un miembro de período definido cuyo nombre se ha cambiado que se ha suplantado por un miembro de período personalizado. Por ejemplo, si ha cambiado el nombre del miembro de período *YearTotal* definido a *unused_YearTotal* y, a continuación, ha agregado, un miembro de período de tipo alternativo con el nombre de miembro definido original (*YearTotal* en este ejemplo). En este escenario, la clonación del entorno puede fallar.

Para obtener información detallada sobre estos temas, consulte Clonación de entornos de Cloud EPM en *Administración de la migración*.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Se necesita el rol Administrador de dominio de identidad para clonar usuarios y roles predefinidos.

Sintaxis

`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]`, donde:

> **Note:**
>
> - El parámetro `dataManagement` no se aplica a entornos de Oracle Enterprise Data Management Cloud ni a Narrative Reporting.
> Clone los registros de Data Management solo si tanto el entorno de origen como el de destino están en la misma actualización mensual o el entorno de destino está en una actualización más actual que el entorno de origen. Por ejemplo, puede clonar registros del entorno 22.01 de Data Management a otro entorno 22.01 o solo a un entorno de 22.02.
> - El parámetro `jobConsole` solo se aplica a Planning, Planning Modules, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning y Strategic Workforce Planning.
> - El parámetro `appAudit` solo se aplica a Planning, Planning Modules, FreeForm, Enterprise Profitability and Cost Management, Sales Planning y Strategic Workforce Planning.
> La información de auditoría para Financial Consolidation and Close y Tax Reporting está, de forma predeterminada, incluida en la instantánea.
> - Si el parámetro `dataManagement`, `jobConsole` o `appAudit` no se puede aplicar a un entorno, el comando ignora el valor que especifique.
> - El parámetro `ApplicationProperties` se aplica solo a Account Reconciliation.

- *`TARGET_USERNAME`* es el ID de un administrador del servicio del entorno de destino. Debe utilizar el nombre de usuario del dominio de identidad de destino (no el nombre de usuario de inicio de sesión único). Si tiene previsto clonar las asignaciones de usuarios y roles en el entorno de destino, este usuario debe tener también el rol de administrador del dominio de identidad.
- *`TARGET_PASSWORD`* es la ubicación del archivo de contraseña cifrado del usuario identificado con *`TARGET_USERNAME`*.
- *`TARGET_URL`* es la `CLOUD-EPM_BASE_URL` del entorno que se convertirá en el entorno clonado.
- *`SnapshotName`*, opcionalmente, es el nombre de una instantánea que se debe usar para la clonación. Esta instantánea debe estar presente en el entorno de origen. El valor predeterminado es `Artifact Snapshot`, que usa la última instantánea de mantenimiento para clonar el entorno.
- *`UsersAndPreDefinedRoles`*, opcionalmente, identifica si se va a clonar a usuario y sus asignaciones de rol predefinido (los grupos de Control de acceso siempre se clonan). El valor predeterminado es `false`.

Para que esta opción funcione, el usuario identificado con *TARGET_USER_NAME* debe tener el rol de administrador del dominio de identidad del entorno de destino.

La importación de usuarios y sus roles predefinidos fallará si un usuario que no es un Administrador de dominio de identidad clona un entorno después de seleccionar esta casilla de verificación. Se registrará el siguiente error en el Informe de estado de la migración: `Error al importar el artefacto del directorio externo <nombre_artefacto>. El usuario <nombre_usuario> no tiene autorización para realizar esta operación. El usuario debe tener el rol de administrador de dominio de identidad para realizar esta operación.`

- Si no importa los usuarios y un usuario de la instantánea de origen no está asignado a un rol predefinido en el entorno de destino, se muestra un error (`EPMIE-00070: Error al buscar el usuario durante la importación de las funciones asignadas`).
- La asignación del rol Administrador de dominio de identidad no se clona. Los usuarios que solo tienen la asignación del rol Administrador de dominio de identidad no se clonan en el entorno de destino.
  Se clonan los usuarios asignados a una combinación del rol Administrador de dominio de identidad y los roles predefinidos en el entorno de origen, pero se asignan solo a los roles predefinidos respectivos en el entorno de destino. Estos usuarios no tendrán el rol Administrador de dominio de identidad en el entorno de destino.
- Los cambios en los roles predefinidos del usuario se actualizarán en función de los roles asignados en la instantánea de origen. Sin embargo, las asignaciones de roles del destino no se eliminarán para que coincidan con las de la instantánea de origen. Por ejemplo, asuma que a `jdoe` se le ha asignado el rol predefinido Usuario avanzado en el entorno de destino, pero solo tiene el rol Usuario en la instantánea de origen. En esta situación, este comando asigna a `jdoe` el rol Usuario y no elimina la asignación de rol Usuario avanzado en el entorno de destino.
- Con este comando no se suprimen los usuarios existentes del entorno de destino si no existen en la instantánea de origen. Por ejemplo, `jdoe` tiene una cuenta en el entorno de destino, pero esta cuenta no está en la instantánea de origen. En esta situación, la cuenta de `jdoe` en el entorno de destino no se suprime.
- Con este comando se agregan usuarios que no existan en el entorno de destino; no se actualizan las propiedades del usuario actual en el entorno de destino, incluso aunque sean distintos en la instantánea de origen. Por ejemplo, si el apellido de `jdoe` en la instantánea de origen está escrito de forma distinta en el entorno de destino, el cambio no se realizará en el entorno de destino. Se asigna una contraseña aleatoria a los nuevos usuarios del entorno de destino. Los usuarios nuevos recibirán mensajes de correo electrónico de activación de la cuenta en que se les solicite que cambien las contraseñas.
- Con este comando no se cambian las contraseñas de los usuarios existentes en el entorno de destino, incluso aunque sean diferentes en las instantáneas de origen.

- `dataManagement=true|false`, opcionalmente, clona registros de Data Management del entorno de origen en el entorno de destino. El valor predeterminado es `true`, que clona los registros de Data Management. Establezca este valor en `false` si no desea clonar registros de Data Management.
- `appAudit=true|false`, opcionalmente, clona los registros de auditoría del entorno de origen en el entorno de destino. El valor predeterminado es `true`, que clona los datos de auditoría de la aplicación. Establezca este valor en `false` si no desea clonar los datos de auditoría de la aplicación en el entorno de destino.

- Con `jobConsole=true|false`, opcionalmente, se clonan los registros de auditoría del entorno de origen en el entorno de destino El valor predeterminado es `true`. Establezca este valor en `false` si no desea clonar registros de la consola de trabajos.
- Con `storedSnapshotsAndFiles`, opcionalmente, se identifica si el comando debe clonar el contenido del buzón y de la bandeja de salida y las instantáneas almacenadas. El valor predeterminado es `false`.

> **Note:**
>
> Solo se clonan las carpetas de nivel superior de la bandeja de entrada y el buzón de salida; no las subcarpetas. Si necesita conservar el contenido de las subcarpetas, realice una copia de seguridad de estas en un equipo local y, a continuación, cárguelas en el entorno de destino.

- Con *`DailyMaintenanceStartTime`*, opcionalmente, se restablece la hora de inicio del mantenimiento del entorno de destino clonado en la del entorno de origen. El valor predeterminado es `true`. Para mantener la hora de inicio del mantenimiento del entorno de destino, establezca este valor en `false`.
- `ApplicationProperties`, opcionalmente, clona la configuración de la aplicación de Account Reconciliation (experiencia Redwood, tema, nombre de proceso de negocio, imagen del logotipo e imagen de fondo). El valor predeterminado es `true`.
  Para que el entorno de destino conserve su configuración de aplicación actual, establezca este valor en `false`.

Ejemplos

- Clone el entorno, las asignaciones de usuarios y roles predefinidos, los datos de auditoría, los registros de la consola de trabajos y los registros de Data Management. Cambie también la hora de inicio del mantenimiento del entorno de destino para que coincida con la del entorno de origen:
  ```
  epmAutomate cloneEnvironment serviceAdmin Password.epw https://test-cloudpln.pbcs.us1.oraclecloud.com UsersAndPreDefinedRoles=true
  ```
- Clone el entorno, incluido el contenido del buzón y de la bandeja de salida, las instantáneas almacenadas, pero no las asignaciones de usuarios y roles predefinidos, los registros de Data Management, los datos de auditoría y los registros de la consola de trabajos, sin cambiar la hora de inicio del mantenimiento del entorno de destino:
  ```
  epmAutomate cloneEnvironment serviceAdmin Password.epw https://test-cloudpln.pbcs.us1.oraclecloud.com DataManagement=false appAudit=false jobConsole=false storedSnapshotsAndFiles=true DailyMaintenanceStartTime=false
  ```
- Clone el entorno completo (asignaciones de usuarios y roles predefinidos, los registros de la consola de trabajos, el contenido del buzón y la bandeja de salida, las instantáneas almacenadas y los registros de Data Management) con una instantánea personalizada. Cambie también la hora de inicio del mantenimiento del entorno de destino para que coincida con la del entorno de origen:
  ```
  epmAutomate cloneEnvironment serviceAdmin Password.epw https://test-cloudpln.pbcs.us1.oraclecloud.com UsersAndPreDefinedRoles=true storedSnapshotsAndFiles=true SnapshotName=SampleSnapshot
  ```
- **Solo para Account Reconciliation:** clona el entorno, los usuarios y las asignaciones de roles predefinidos y registros de Data Management, pero conserva la configuración de las aplicaciones del entorno de destino:

```
epmAutomate cloneEnvironment serviceAdmin Password.epw https://test-
cloudarcs.arcs.epm.us.oraclecloud.com UsersAndPreDefinedRoles=true
ApplicationProperties=false
```

compactCube

Compacta el archivo de esquema de un cubo de almacenamiento agregado (ASO) con la configuración definida en un trabajo del tipo *Compactar cubo*. La compactación del esquema ayuda a mantener el tamaño óptimo de un archivo de esquema del cubo. Al compactar no se borran los datos del cubo ni se introducen los cambios del proceso de negocio en Oracle Essbase.

Se aplica a

Planning, Planning Modules, FreeForm, Enterprise Profitability and Cost Management, Sales Planning y Strategic Workforce Planning

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate compactCube` *`ASO_CUBE_NAME`*, donde *`ASO_CUBE_NAME`* es el nombre del cubo de ASO, cuyo esquema desea compactar.

Ejemplo

```
epmautomate compactCube Vis1Aso
```

copyDataByPointOfView

Copia datos de un PDV de origen de un cubo en un PDV de destino del mismo cubo de Enterprise Profitability and Cost Management o de otro.

Se aplica a

Enterprise Profitability and Cost Management

Roles necesarios

Administrador del servicio

Sintaxis

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

- *`SOURCE_POV_NAME`* es el nombre del PDV de origen del que se van a copiar los datos.
- *`TARGET_POV_NAME`* es el nombre de un PDV de destino válido en el que se van a copiar los datos del origen.
- `copyType` identifica los datos que se van a copiar del PDV de origen. Los valores válidos son:
  - `ALL_DATA` para copiar todos los datos de entrada y calculados en el PDV de destino.

  - `INPUT` para copiar todos los datos de entrada, incluidos los datos del controlador, en el PDV de destino.
- *`SOURCE_CUBE_NAME`* es el nombre del cubo que contiene el PDV de origen.
- *`TARGET_CUBE_NAME`* es el nombre del cubo que contiene el PDV de destino.
- `PARAMETER=`*`VALUE`* indica los parámetros de tiempo de ejecución opcionales y sus valores. Especifique tantos pares de parámetro y valor como requiera el proceso. Parámetros válidos y sus valores:
  - `povDelimiter`, opcionalmente, es el delimitador utilizado en los valores de PDV. El valor predeterminado es `::` (dos puntos dobles). Este valor se debe escribir entre comillas dobles. Ejemplo: `povDelimiter="_"`.
    Además de los valores predeterminados, solo están soportados estos delimitadores: _ (carácter de subrayado), # (almohadilla), & (ampersand), ~ (virgulilla), % (porcentaje), ; (punto y coma), : (dos puntos), - (guion).
  - `createDestPOV=true|false` especifica si se va a crear el PDV de destino si no existe. El valor predeterminado es `false`. Debe establecer este valor de parámetro en `true` si no existe el PDV de destino.

Ejemplos

- Copiar todos los datos en un PDV distinto del mismo cubo:
```
epmAutomate copyDataByPointOfView FY21_Jan_Actual_Working
FY22_Jan_Actual_Working ALL_DATA PCM_CLC PCM_CLC povDelimiter="_"
createDestPOV=true
```
- Copiar todos los datos en un PDV distinto en un cubo distinto:
```
epmAutomate copyDataByPointOfView FY21_Jan_Actual_Working
FY22_Jan_Actual_Working ALL_DATA PCM_CLC PCM_REP povDelimiter="_"
createDestPOV=true
```
- Copiar los datos de entrada en un PDV distinto del mismo cubo:
```
epmAutomate copyDataByPointOfView FY21_Jan_Actual_Working
FY22_Jan_Actual_Working INPUT PCM_CLC PCM_CLC povDelimiter="_"
createDestPOV=true
```
- Copiar los datos de entrada en un PDV distinto de un cubo distinto:
```
epmAutomate copyDataByPointOfView FY21_Jan_Actual_Working
FY22_Jan_Actual_Working INPUT PCM_CLC PCM_REP povDelimiter="_"
createDestPOV=true
```

copyDataByProfile

Copia datos para los elementos (por ejemplo, regiones) identificados en un perfil de borrado de datos.

Se aplica a

Financial Consolidation and Close y Tax Reporting

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate copyDataByProfile` *`PROFILE_NAME`* donde *`PROFILE_NAME`* es el nombre de un perfil de copia de datos definido en Financial Consolidation and Close y Tax Reporting.

Ejemplo

```
epmautomate copyDataByProfile copyDataProfile_01
```

copyFileFromInstance

Copia un archivo de un entorno de origen al entorno desde el que se ejecuta este comando.

Antes de ejecutar este comando, con EPM Automate, conéctese al entorno en el que desee copiar el archivo.

Se aplica a

Planning, Planning Modules, 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 y Sales Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado y el rol de aplicación Migraciones - Administrar (Oracle Enterprise Data Management Cloud y Profitability and Cost Management)

Sintaxis

`epmautomate copyFileFromInstance` *`SOURCE_FILE_NAME USERNAME PASSWORD_FILE URL TARGET_FILE_NAME`* donde:

- *`SOURCE_FILE__NAME`* es el nombre del archivo (incluida la extensión) que se quiere copiar desde el entorno origen.
- *`USERNAME`* es el nombre de usuario de un administrador del servicio del entorno de origen.
- *`PASSWORD_FILE`* es el nombre y la ubicación del archivo que contiene la contraseña cifrada del administrador de servicio del entorno de origen.
- *`URL`* es la URL del entorno de origen.
- *`TARGET_FILE_NAME`* es un nombre único para el archivo (incluida la extensión) en el entorno desde el que se ejecuta este comando.

Ejemplo

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

copyFromObjectStorage

Copia un archivo o una instantánea de copia de seguridad de un cubo de Oracle Object Storage en el entorno actual.

Si copia una instantánea de copia de seguridad, este comando la copia del cubo de Object Storage y extrae su contenido en Oracle Fusion Cloud Enterprise Performance Management.

Se aplica a

Planning, Planning Modules, 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 y Sales Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate copyFromObjectStorage` *`USERNAME PASSWORD URL TARGET_FILE_NAME`* donde:

- *`USERNAME`* es el ID de usuario de un usuario que tiene los derechos de acceso necesarios en Oracle Object Storage Cloud.
  Para los usuarios creados en un proveedor de identidad federado, especifique el nombre completo del usuario (por ejemplo, `exampleIdP/jdoe` o `exampleIdP/john.doe@example.com`, donde `exampleIdP` es el nombre del proveedor de identidad federado). Para otros usuarios, proporcione el ID de usuario.
- *`PASSWORD`* es la contraseña de Swift o el símbolo de autentificación asociado al usuario. Esta contraseña no es la misma que la que ha usado para iniciar sesión en la consola de Object Storage. El símbolo de autentificación es un símbolo generado por Oracle que usa para autentificarse en las API de terceros, por ejemplo, para autentificarse en un cliente de Swift. Para obtener instrucciones para crear este símbolo, consulte Para crear un símbolo de autentificación en *Documentación de Oracle Cloud Infrastructure* .
- *`URL`* es la URL del cubo de Oracle Object Storage Cloud, incluidos el nombre de cubo y el nombre del objeto que se va a copiar.
  Formato de URL:

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

  Componentes de esta URL:

  - `region_identifier` es una región de alojamiento de Oracle Cloud Infrastructure.
  - `namespace` es el contenedor de nivel superior para todos los cubos y objetos. A cada inquilino de Oracle Cloud Infrastructure se le asigna un nombre de espacio de nombre de Object Storage inmutable, generado por el sistema y único durante la creación de la cuenta. El nombre del espacio de nombres del arrendamiento, por ejemplo, `axaxnpcrorw5`, se aplica a todas las regiones.
  - `bucket_name` es el nombre de un contenedor lógico donde se almacenan los datos y archivos. Los cubos se organizan y mantienen en compartimentos. Un nombre de cubo generado por el sistema, por ejemplo, `bucket-20210301-1359` refleja el año, el mes, el día y la hora actuales.
  - `object_name` es el nombre de la instantánea o del archivo que desea copiar de Oracle Object Storage Cloud. Este valor debe coincidir exactamente con el nombre completo del objeto en Object Storage Cloud. No use una extensión como `.zip` a menos que el objeto la contenga.

  Para obtener más información, consulte estos temas en la *documentación de Oracle Cloud Infrastructure*

  - Regiones y dominios de disponibilidad
  - Descripción de los espacios de nombre de Object Storage
  - Gestión de cubos

- *TARGET_FILE_NAME* es un nombre único para el archivo o la instantánea en el entorno de Cloud EPM. Al copiar instantáneas, no especifique la extensión ZIP para que se pueda usar este nombre de archivo con el comando importSnapshot.
  Los archivos que superan los 100 MB se almacenan en Oracle Object Storage en un directorio lógico junto al archivo de manifiesto que identifica sus segmentos. Especifique el nombre del directorio lógico como *TARGET_FILE_NAME*.

Ejemplos

En estos ejemplos, reemplace *URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET* por una URL en funcionamiento con este formato: `https://swiftobjectstorage.`*`region_identifier`*`.oraclecloud.com/v1/`*`namespace/bucket_name/`*.

- Copie una instantánea denominada `backup_Snapshot_12_05_20.zip` del cubo de Oracle Object Storage en Cloud EPM y cámbiele el nombre:
  ```
  epmautomate copyFromObjectStorage oracleidentitycloudservice/jDoe example_pwd
  URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET/backup_Snapshot_12_05_20.zip
  snapshot_from_osc
  ```
- Copie una instantánea denominada `backup_Snapshot_12_05_20` del cubo de Oracle Object Storage en Cloud EPM y cámbiele el nombre:
  ```
  epmautomate copyFromObjectStorage oracleidentitycloudservice/jDoe example_pwd
  URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET/backup_Snapshot_12_05_20
  snapshot_from_osc
  ```
- Copie una instantánea denominada `backup_Snapshot_12_05_20` del cubo de Oracle Object Storage en Cloud EPM sin cambiarle el nombre:
  ```
  epmautomate copyFromObjectStorage oracleidentitycloudservice/jDoe example_pwd
  URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET/backup_snapshot_12_05_20
  backup_snapshot_12_05_20
  ```
- Copie un archivo en Cloud EPM del cubo de Oracle Object Storage:
  ```
  epmautomate copyFromObjectStorage oracleidentitycloudservice/jDoe example_pwd
  URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET/example_file.txt copied_from_osc.txt
  ```

copyFromSFTP

Copia un archivo directamente de un servidor SFTP en un entorno OCI (Gen 2). Este comando resulta útil para copiar un archivo de datos directamente de un servidor SFTP seguro en un entorno de Oracle Fusion Cloud Enterprise Performance Management. Con este comando, puede copiar un archivo directamente de un servidor SFTP sin descargarlo primero en otro servidor utilizando un cliente SFTP y, a continuación, utilizando EPM Automate o la API de REST para cargarlo en Cloud EPM.

Si la lista de direcciones IP permitidas está activada para el servidor SFTP, debe agregar la dirección IP de salida de la región de OCI que aloja los entornos de Cloud EPM a la lista de IP permitidas del servidor SFTP. Consulte Direcciones IP de salida de los centros de datos y las regiones de Cloud EPM en la *Guía de operaciones* para conocer las direcciones IP de salida de la regiones de OCI.

> **Note:**
>
> Este comando soporta un servidor SFTP que se ejecute solo en el puerto 22.

Se aplica a

Planning, Planning Modules, 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 y Sales Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate copyFromSftp SFTP_SERVER_URL FILE_NAME username=USERNAME password=PASSWORD`, donde:

- *SFTP_SERVER_URL* es la URL del servidor SFTP que incluye la ubicación y el nombre del archivo que desea copiar del servidor.
- *FILE_NAME* es el nombre del archivo en el entorno de Cloud EPM. Especifique la ruta, por ejemplo, si no está copiando en la ubicación de carga predeterminada.
- *USERNAME* es el ID de un usuario que tiene los derechos de acceso necesarios en el servidor SFTP.
- *PASSWORD* es la contraseña de usuario SFTP.

Ejemplo

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

copyOwnershipDataToNextYear

Copia los datos de propiedad del último período de un año en el primer período del año siguiente.

La configuración predeterminada inicial y de sustitución de propiedad se traslada automáticamente de un periodo a otro dentro del mismo año, pero no a los años siguientes. Para trasladar la configuración de propiedad más actual del último periodo de un año al primer periodo del año siguiente, debe copiar la configuración de propiedad del último periodo del año en el PDV y en el primer periodo del año siguiente.

Se aplica a

Financial Consolidation and Close y Tax Reporting.

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario

Sintaxis

`epmautomate copyOwnershipDataToNextYear` *`Scenario Year`* donde:

- `Scenario` es el nombre del escenario del que se copian los datos de propiedad.
- `Year` es el año cuyos datos de propiedad se copian en el primer periodo del año siguiente.

Ejemplo

```
epmautomate copyOwnershipDataToNextYear FCCS_total_Actual FY18
```

copyPOV

Copia los artefactos del modelo y los datos del cubo de Oracle Essbase del PDV de origen en un PDV de destino.

Se aplica a

Profitability and Cost Management.

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

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

- *`APPLICATION_NAME`* es el nombre de la aplicación de Profitability and Cost Management que contiene el PDV de origen.
- *`SOURCE_POV_NAME`* es el nombre del PDV de origen en la aplicación especificada
- *`NOMBRE_PDV_DESTINO`* es el nombre de un PDV de destino válido con el estado `Draft`
- `PARAMETER=`*`VALUE`* indica los parámetros de tiempo de ejecución y sus valores para copiar el PDV. Especifique tantos pares de parámetro y valor como requiera el proceso. Parámetros válidos y sus valores:
  - `isManageRule=true|false` especifica si se van a copiar las reglas.
  - `isInputData=true | isAllData=true | isAllInputData=true` especifica opcionalmente cómo copiar los datos. Para estos parámetros, el valor predeterminado es false. Especifique solo uno de estos elementos como true:
    - especifique `isInputData=true` para copiar los datos de entrada en el PDV de destino.
    - especifique `isAllData=true` para copiar todos los datos de entrada y calculados en el PDV de destino.
    - `AllInputData=true` para copiar todos los datos de entrada, incluidos los datos de inductor, en el PDV de destino.
  - `modelViewName=`*`NAME`* especifica el nombre del segmento de datos que se va a copiar desde el PDV de origen al PDV de destino.

  - `createDestPOV=true|false` especifica si se va a crear el PDV de destino si no existe.
  - `nonEmptyTupleEnabled=true|false` especifica si se va a activar la tupla no vacía (NET) de manera que el comando solo tenga en cuenta las intersecciones que incluyan datos. El valor predeterminado es `true` que, en pocas ocasiones, puede provocar que el comando no se ejecute bien al copiar datos de Essbase. En esos casos, sobrescriba el valor predeterminado con `nonEmptyTupleEnabled=false` para mejorar el rendimiento.

> **Nota:**
>
> Los valores de parámetros (`true` o `false`) deben ir en minúscula.

- `stringDelimiter="DELIMITER"` especifica el delimitador utilizado en los valores de PDV. El delimitador debe ir entre comillas dobles.

Ejemplos

- `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

Copia la instantánea actual de un entorno de origen al entorno (destino) desde el que se ejecuta este comando.

Este comando se utiliza principalmente como primer paso para migrar un entorno copiando la instantánea actual de otro entorno; por ejemplo, de un entorno de prueba a un entorno de producción. Utilice el comando importSnapshot para completar el proceso de migración.

Antes de ejecutar este comando, inicie una sesión en EPM Automate y en el entorno de destino.

Si este comando se ejecuta para copiar la instantánea actual mientras se está generando la instantánea del entorno de origen; por ejemplo, durante el mantenimiento diario, verá el error `File not found`.

Se aplica a

Planning, Planning Modules, 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 y Sales Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado y el rol de aplicación Migraciones - Administrar (Oracle Enterprise Data Management Cloud y Profitability and Cost Management)

Sintaxis

`epmautomate copySnapshotFromInstance SNAPSHOT_NAME USERNAME PASSWORD_FILE URL` donde:

- *SNAPSHOT_NAME* es el nombre de una instantánea existente en el entorno de origen.
- *USERNAME* es el nombre de usuario de un administrador de servicio del entorno de origen.
- *PASSWORD_FILE* es el nombre y la ubicación del archivo que contiene la contraseña cifrada del administrador de servicio del entorno de origen.
- *URL* es la URL del entorno de origen.

Ejemplo

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

copyToObjectStorage

Copia un archivo o una instantánea del entorno actual a un cubo de Oracle Object Storage Cloud.

Si copia una instantánea, este comando comprime en un archivo ZIP el contenido de la instantánea antes de copiarla en Oracle Object Storage.

Para facilitar la copia rápida de archivos, este comando divide los archivos grandes (que superan los 100 MB) en segmentos de 10 MB (denominados *FILE_NAME*/*FILE_NAME*`_object_store_bytes_seg_0` por *FILE_NAME*/*FILE_NAME*`_object_store_bytes_seg_`*n*) y crea un archivo de manifiesto (denominado *FILE_NAME*/*FILE_NAME*`.manifest`). Los segmentos de archivo se almacenan en Oracle Object Storage junto al archivo de manifiesto. En la consola de Object Storage, el archivo se muestra como un directorio lógico que contiene los segmentos de archivo y el archivo de manifiesto.

Los archivos inferiores a los 100 MB no están segmentados y se almacenan con el nombre de archivo original.

Se aplica a

Planning, Planning Modules, 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 y Sales Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate copyToObjectStorage SOURCE_FILE_NAME USERNAME PASSWORD URL` donde:

- *SOURCE_FILE_NAME* es el nombre del archivo o la instantánea en Oracle Fusion Cloud Enterprise Performance Management. Si copia una instantánea, no especifique la extensión ZIP.
- *USERNAME* es el ID de usuario de un usuario que tiene los derechos de acceso necesarios para escribir en Oracle Object Storage Cloud.

Para los usuarios creados en un proveedor de identidad federado, especifique el nombre completo del usuario (por ejemplo, `exampleIdP/jdoe` o `exampleIdP/john.doe@example.com`, donde `exampleIdP` es el nombre del proveedor de identidad federado). Para otros usuarios, proporcione el ID de usuario.

- *`PASSWORD`* es la contraseña de Swift o el símbolo de autentificación asociado al usuario. Esta contraseña no es la misma que la que ha usado para iniciar sesión en la consola de Object Storage. El símbolo de autentificación es un símbolo generado por Oracle que usa para autentificarse en las API de terceros, por ejemplo, para autentificarse en un cliente de Swift. Para obtener instrucciones para crear este símbolo, consulte Para crear un símbolo de autentificación en *Documentación de Oracle Cloud Infrastructure* .
- *`URL`* es la URL del cubo de Oracle Object Storage Cloud con un nombre de objeto opcional al final.
  El formato de URL sin nombre de objeto:

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

  El formato de URL con nombre de objeto:

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

  Componentes de esta URL:

  - `region_identifier` es una región de alojamiento de Oracle Cloud Infrastructure.
  - `namespace` es el contenedor de nivel superior para todos los cubos y objetos. A cada inquilino de Oracle Cloud Infrastructure se le asigna un nombre de espacio de nombre de Object Storage inmutable, generado por el sistema y único durante la creación de la cuenta. El nombre del espacio de nombres del arrendamiento, por ejemplo, `axaxnpcrorw5`, se aplica a todas las regiones.
  - `bucket_name` es el nombre de un contenedor lógico donde se almacenan los datos y archivos. Los cubos se organizan y mantienen en compartimentos. Un nombre de cubo generado por el sistema, por ejemplo, `bucket-20210301-1359` refleja el año, el mes, el día y la hora actuales.
  - `object_name`, opcionalmente, es el nombre que desea usar para el archivo en Oracle Object Storage Cloud. Si no se especifica un nombre de objeto, el archivo se copiará con su nombre original.

  Para obtener más información, consulte estos temas en la *documentación de Oracle Cloud Infrastructure*

  - Regiones y dominios de disponibilidad
  - Descripción de los espacios de nombre de Object Storage
  - Gestión de cubos

Ejemplos

En estos ejemplos, reemplace *`URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET`* por una URL en funcionamiento con este formato: `https://swiftobjectstorage.`*`region_identifier`*`.oraclecloud.com/v1/`*`namespace/bucket_name/`*.

- Copie una instantánea en un cubo de Object Storage y cámbiele el nombre:
  ```
  epmautomate copyToObjectStorage "Artifact Snapshot"
  oracleidentitycloudservice/jDoe example_pwd
  URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET/Snapshot_04_30_21
  ```

- Copie un archivo en un cubo de Object Storage:
  `epmautomate copyToObjectStorage example_file.txt oracleidentitycloudservice/jDoe example_pwd URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET`
- Copie un archivo en un cubo de Object Storage y cámbiele el nombre:
  `epmautomate copyToObjectStorage example_file.txt eoracleidentitycloudservice/jDoe example_pwd URL_OF_THE_ORACLE_OBJECT_STORAGE_BUCKET/epm_text_file.txt`

copyToSFTP

Copia un archivo de una ubicación de un entorno OCI (Gen 2) en una ubicación de un servidor FTP seguro. Con este comando, puede copiar un archivo directamente de un entorno de Oracle Fusion Cloud Enterprise Performance Management en un servidor SFTP sin descargarlo primero en otro servidor utilizando EPM Automate o la API de REST y, a continuación, utilizar un cliente SFTP para cargarlo en el servidor SFTP.

Si la lista de direcciones IP permitidas está activada para el servidor SFTP, debe agregar la dirección IP de salida de la región de OCI que aloja los entornos de Cloud EPM a la lista de IP permitidas del servidor SFTP. Consulte Direcciones IP de salida de los centros de datos y las regiones de Cloud EPM en la *Guía de operaciones* para conocer las direcciones IP de salida de la regiones de OCI.

> **Note:**
>
> Este comando soporta un servidor SFTP que se ejecute solo en el puerto 22.

Se aplica a

Planning, Planning Modules, 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 y Sales Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate copyToSftp SFTP_SERVER_URL EPM_FILE_NAME username=USERNAME password=PASSWORD`, donde:

- *SFTP_SERVER_URL* es la URL del servidor SFTP que incluye el directorio en el que se va a copiar el archivo y el nombre del archivo.
- *EPM_FILE_NAME* es el nombre y la ubicación (si no es del buzón de salida predeterminado) del archivo que desea copiar del entorno de Cloud EPM . No especifique la extensión .ZIP si está copiando una instantánea.
- *USERNAME* es el ID de un usuario que tiene los derechos de acceso necesarios en el servidor SFTP.
- *PASSWORD* es la contraseña de usuario SFTP.

Ejemplo

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

createGroups

Agrega grupos a Control de acceso mediante el uso de un archivo CSV con codificación ANSI o UTF-8 que se haya cargado en el entorno.

Un administrador del servicio utiliza el comando uploadFile para cargar archivos en un entorno. El formato del archivo es el siguiente:

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

Los nombres de grupos no distinguen mayúsculas y minúsculas. Cuando finalice, el comando imprime información sobre cada una de las entradas con fallos en la consola. Consulte esta información para saber el motivo por el que se ha producido un fallo en la ejecución del comando en algunas entradas del archivo CSV.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Gestionar

Sintaxis

`epmautomate createGroups` *`FILE_NAME`* donde *`FILE_NAME`* es el nombre del archivo CSV que contiene los nombres de grupos y las descripciones.

Ejemplo

```
epmautomate createGroups group_file.CSV
```

createNRSnapshot

Cree una instantánea bajo demanda, denominada `EPRCS_Backup.tar.gz`, de un entorno de Narrative Reporting.

Puede descargar `EPRCS_Backup.tar.gz` y un archivo de error en un equipo local con el comando downloadFile o copiarlo en otro entorno con el comando copyFileFromInstance.

Los datos de aplicación de `EPRCS_Backup.tar.gz` son de la fecha del último mantenimiento diario. Si necesita realizar una copia de seguridad de los datos más recientes, use la función de exportación de datos de Narrative Reporting.

Se aplica a

Narrative Reporting

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate createNRSnapshot [errorFile=Error_File.txt]`, donde `errorFile`, opcionalmente, identifica el nombre de un archivo de texto único para registrar los errores, si el comando encuentra alguno.

Ejemplo

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

createReconciliations

Copia los perfiles en el periodo especificado.

Se aplica a

Account Reconciliation.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido (el rol Visor requiere el rol de aplicación Reconciliación - Preparador)

Sintaxis

`epmautomate createreconciliations` *`PERIOD SAVED_FILTER`* donde:

- *`PERIODO`* es el nombre de un periodo
- *`SAVED_FILTER`* es el nombre de un filtro público guardado. Si no especifica ningún filtro guardado, el comando copia todos los perfiles aplicables

Ejemplos

- Copie todos los perfiles para el período: `epmautomate createReconciliations "January 2015"`
- Copie los perfiles de un filtro específico: `epmautomate createReconciliations "January 2015" "Corporate Recs"`

deleteFile

Suprime un archivo o una instantánea de la ubicación de carga predeterminada, el buzón o la bandeja de salida, una carpeta Data Management o de profitinbox/profitoutbox.

Para suprimir un archivo de una ubicación diferente a la ubicación de carga predeterminada, debe especificar la ubicación del archivo.

Si este comando se ejecuta para suprimir una instantánea que se esté generando o archivando, verá uno de estos errores:

- `File not found` si la instantánea se está generando
- `Archieve process is in progress. Unable to Rename or Delete` si la instantánea se está archivando

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado y el rol de aplicación Migraciones - Administrar (Oracle Enterprise Data Management Cloud y Profitability and Cost Management)

Sintaxis

```
epmautomate deleteFile FILE_NAME
```

> **Nota:**
>
> Debe especificar el nombre del archivo, incluida la extensión; por ejemplo, `data.csv`, `data.zip`, si procede. Puede suprimir una instantánea sin especificar la extensión del archivo (.zip). Sin embargo, este uso no está aprobado. Debe especificar la ubicación del archivo si no se encuentra en la ubicación predeterminada. Para obtener información detallada, consulte Ubicaciones de archivo predeterminadas. Entre las ubicaciones soportadas se incluyen `inbox`, `profitinbox`, `outbox`, `profitoutbox`, `to_be_imported` y `inbox/directory_name`.

Ejemplos

- Suprima un archivo de la ubicación de carga predeterminada:
  `epmautomate deleteFile data.csv`
- Suprima un archivo del buzón:
  `epmautomate deleteFile inbox/data.csv`
- Suprima de la bandeja de salida:
  `epmautomate deleteFile outbox/data.csv`
- Suprima una instantánea que haya creado mediante la migración:
  - `epmautomate deleteFile "Backup 18-06-12.zip"` o
  - `epmautomate deleteFile "Backup 18-06-12"` (no aprobado)
- Suprima de profitinbox (Profitability and Cost Management):
  `epmautomate deleteFile profitinbox/data.csv`
- Suprima de profitoutbox (Profitability and Cost Management):
  `epmautomate deleteFile profitoutbox/data.csv`
- Suprima de una carpeta de carga de Data Management:

```
epmautomate deleteFile inbox/dm_data/data.csv
```

- Suprima de una carpeta de Data Management:
  `epmautomate deleteFile outbox/dm_data/data.csv`

deleteGroups

Elimina grupos de Control de acceso según la información disponible en un archivo CSV con codificación ANSI o UTF-8 que se haya cargado en el entorno.

Un administrador del servicio utiliza el comando uploadFile para cargar archivos en un entorno. El formato del archivo es el siguiente:

```
Group Name
Example_grp1
Example_grp2
```

Los valores de nombres de grupos del archivo no distinguen mayúsculas y minúsculas. Cuando finalice, el comando imprime información sobre cada una de las entradas con fallos en la consola. Consulte esta información para saber el motivo por el que se ha producido un fallo en la ejecución del comando en algunas entradas del archivo CSV.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Gestionar

Sintaxis

`epmautomate deleteGroups` *`FILE_NAME`* donde *`FILE_NAME`* es el nombre de un archivo CSV que contiene los nombres de los grupos que se van a eliminar de Control de acceso.

Ejemplo

```
epmautomate deleteGroups group_file.CSV
```

deletePointOfView

Suprime los artefactos y los datos del cubo de Oracle Essbase de un PDV en una aplicación de Enterprise Profitability and Cost Management.

Se aplica a

Enterprise Profitability and Cost Management

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate deletePointOfView POV_NAME [povDelimiter="DELIMITER"]`, donde:

- *POV_NAME* identifica el nombre del PDV que se va a suprimir.
- `povDelimiter` es el delimitador utilizado en los valores de PDV. El valor predeterminado es `::` (dos puntos dobles). Este valor se debe escribir entre comillas dobles. Ejemplo: `povDelimiter="_"`.
  Además de los valores predeterminados, solo están soportados estos delimitadores: _ (carácter de subrayado), # (almohadilla), & (ampersand), ~ (virgulilla), % (porcentaje), ; (punto y coma), : (dos puntos), - (guion).

Ejemplo

- Supresión de un PDV que use un delimitador de PDV personalizado
  `epmAutomate deletePointOfView FY21_Jan_Actual_Working povDelimiter="_"`
- Supresión de un PDV que use el delimitador de PDV predeterminado
  `epmAutomate deletePointOfView FY21::Jan::Actual::Working`

deletePOV

Suprime los artefactos del modelo y los datos del cubo de Oracle Essbase de un PDV en Profitability and Cost Management.

Se aplica a

Profitability and Cost Management

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

`epmautomate deletePOV APPLICATION_NAME POV_NAME stringDelimiter="DELIMITER"`, donde:

- *APPLICATION_NAME* es el nombre de la aplicación de Profitability and Cost Management que contiene el PDV que se va a suprimir.
- *POV_NAME* es el nombre del PDV que se va a suprimir. Este valor es necesario.
- `stringDelimiter="DELIMITER"` especifica el delimitador utilizado en los valores de PDV. El delimitador debe ir entre comillas dobles.

Ejemplo

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

deployCube

Despliega o vuelve a desplegar el cubo de cálculo de una aplicación de Profitability and Cost Management.

Se aplica a

Profitability and Cost Management

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

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

- *APPLICATION_NAME* es el nombre de una aplicación de Profitability and Cost Management
- `PARAMETER=VALUE` indica los parámetros de tiempo de ejecución y sus valores para desplegar el cubo. Especifique tantos pares de parámetro y valor como requiera el proceso. Parámetros válidos y sus valores:

  > **Nota:**
  >
  > Los valores de parámetros (`true` o `false`) deben ir en minúscula.

  - `isKeepData=true|false`

    especifica si se conservan los datos existentes, si los hay
  - `isReplaceCube=true|false` especifica si se debe reemplazar el cubo existente

    > **Nota:**
    >
    > Los valores de `isKeepData` e `isReplaceCube` no se pueden definir ambos en `true`.

  - `isRunNow=true|false` especifica si se va a ejecutar el proceso inmediatamente
- `comment` es un comentario opcional entre comillas dobles

Ejemplo

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

deployEJTemplates

Despliega plantillas de asientos de Enterprise finalizadas en periodos abiertos de Financial Consolidation and Close. El despliegue de plantillas de Asientos de Enterprise crea asientos

recurrentes asociados a la plantilla para el período seleccionado. También le permite crear asientos ad hoc mediante las plantillas desplegadas.

Este comando es una alternativa a usar las pantallas de Financial Consolidation and Close para desplegar nuevas plantillas de Enterprise Journal al empezar el mes.

Se aplica a

Financial Consolidation and Close

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

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

- `Year` es el año del asiento.
- `Period` es el periodo del asiento. Este valor se puede especificar solo si se especifica el año.
- `Template=`*`TEMPLATE_NAME`* identifica el nombre de los asientos que se van a desplegar. Para desplegar más de un asiento, proporcione un nombre de plantilla único con el formato `Template=`*`TEMPLATE_NAME`*, por ejemplo, `Template="Loan Details" Template="Housing Details" Template="Repayment Details"`.
  Si este valor de parámetro no está especificado, el comando despliega todas las plantillas para la combinación de año y período especificada.
- `ResetJournals`, opcionalmente, indica si se deben restablecer todos los asientos a la primera etapa tras volver a desplegar las plantillas. El valor predeterminado es `false`. Financial Consolidation and Close valida este valor internamente en función de los cambios de las plantillas, y puede anular el valor que especifique, si es necesario.

Ejemplo

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

deployFormTemplates

Despliega plantillas de formulario finalizadas en nuevos periodos de recopilación de datos para crear formularios de datos adicionales, lo cual garantiza un proceso de recopilación de datos repetible y consistente.

Se aplica a

Financial Consolidation and Close, Tax Reporting.

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

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

- `COLLECTION_INTERVAL` es el nombre del intervalo de recopilación en el que se va a desplegar la plantilla.
- Opcionalmente, `DIMENSION` especifica las dimensiones de frecuencia del proceso de recopilación de datos con el formato *`DIMENSION=MEMBER_NAME`*. Especifica tantas dimensiones como se hayan definido en el intervalo de recopilación (un máximo de cuatro, incluidos el año y el período; por ejemplo, `"Year=2020" "Period=July" "Product=Oracle EPM" "Consolidation=entity Input"`. No se utiliza ningún valor predeterminado si este valor de parámetro no está especificado.
- Opcionalmente, `Template` identifica los nombres únicos de las plantillas de formulario que se van a desplegar con el formato `Template=`*`TEMPLATE_NAME`*. Puede especificar cualquier número de nombres únicos (tantos como sean necesarios) con este formato. Por ejemplo, `Template="Loan Details Template" Template="Housing Details Template" Template="Repayment Detals Template"`.
  Si este valor de propiedad no está especificado, el comando despliega todas las plantillas para el intervalo especificado.
- Opcionalmente, `resetWorkFlows` indica si se van a restablecer todos los formularios a la primera etapa después de volver a desplegarlos. El valor predeterminado es `false`.

Ejemplo

```
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
```

deployTaskManagerTemplate

Despliega tareas de una plantilla del Gestor de tareas en un programa de tareas, lo cual garantiza una ejecución consistente de procesos de negocio repetitivos.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

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

- *`TEMPLATE_NAME`* es el nombre de la plantilla del Gestor de tareas que desplegar.
- *`SCHEDULE_NAME`* es el nombre del programa que crear a partir de la plantilla.
- *`YEAR`* es el miembro de dimensión Year donde se va a desplegar la plantilla.
- *`PERIOD`* es el miembro de dimensión Period donde se va a desplegar la plantilla.

- *DAY_ZERO_DATE* es la fecha de día cero, en formato válido, que se va a usar para crear el formato.
- `dateFormat`, opcionalmente, es el formato de fecha para la fecha de día cero. El formato predeterminado es YYYY-MM-DD.
- `orgUnit`, opcionalmente, es el nombre de la unidad organizativa. Si no se especifica un valor, se creará el programa con la asignación de fecha estándar. No se usarán reglas de vacaciones.

Ejemplo

- Despliegue la plantilla del Gestor de tareas para la unidad organizativa *Ind* con el formato de fecha predeterminado (AAAA-MM-DD) para la fecha cero:
```
epmautomate deployTaskManagerTemplate "Vision Monthly Close" "Qtr 2 Close"
2021 July 2021-07-10 orgUnit=Ind
```
- Despliegue la plantilla del Gestor de tareas para la unidad organizativa Ind con dd/mm/yyyy como formato de fecha para la fecha cero:
```
epmautomate deployTaskManagerTemplate "Vision Monthly Close" "Qtr 2 Close"
2021 July 02/07/2021 dateFormat=dd/MM/yyyy orgUnit=Ind
```

dismissIPMInsights

Automatiza el descarte de los datos de Intelligent Performance Management (IPM) Insight antes de ejecutar los nuevos trabajos de IPM Insight. Al descartar los datos se cierran todos los análisis abiertos en los que no tenga previsto llevar a cabo alguna acción. Este comando es una alternativa al descarte manual de los datos con el panel de IPM Insight.

Se aplica a

Planning, Planning Modules, Strategic Workforce Planning, Sales Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate dismissIPMInsights [comment="comment"]`, donde comment, opcionalmente, es una justificación para el descarte de los análisis abiertos.

Ejemplo

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

downloadFile

Descarga un archivo de un entorno en el equipo local.

Utilice este comando para descargar datos, metadatos y copias de seguridad de instantáneas para el almacenamiento local. Este archivo se descarga en la carpeta desde la que ejecuta EPM Automate.

Si este comando se ejecuta para descargar la instantánea actual mientras se está generando la instantánea del entorno; por ejemplo, durante el mantenimiento diario, verá el error `File not found`.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado y el rol de aplicación Migraciones - Administrar (Oracle Enterprise Data Management Cloud y Profitability and Cost Management)

Sintaxis

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

Ejemplos

- Descarga de instantánea de mantenimiento: `epmautomate downloadFile "Artifact Snapshot"`
- Descarga de una instantánea personalizada: `epmautomate downloadFile "mySnapshot.zip"`
- Descarga de instantánea de mantenimiento de Narrative Reporting: `epmautomate downloadFile "EPRCS_Backup.tar.gz"`
- Descarga de un archivo de la ubicación de descarga predeterminada: `epmautomate downloadFile data.csv`
- Descarga de una carpeta de Data Management: `epmautomate downloadfile outbox/dm_data/data.csv`
- Descarga de profitoutbox: `epmautomate downloadFile profitOutbox/data.csv`

enableApp

Activa una aplicación.

Se aplica a

Profitability and Cost Management

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

`epmautomate enableapp` *`APPLICATION_NAME`* donde *`APPLICATION_NAME`* es el nombre de la aplicación de Profitability and Cost Management que desea activar.

Ejemplo

```
epmautomate enableApp BksML12
```

enableQueryTracking

Activa el seguimiento de consultas en cubos de ASO para iniciar la captura de los patrones de recuperación de datos de usuario (consultas).

Use los patrones de recuperación de datos capturados para optimizar la agregación de cubos de ASO, que se inicia con el comando executeAggregationProcess.

Se aplica a

Planning, Planning Modules, FreeForm, Enterprise Profitability and Cost Management, Strategic Workforce Planning y Sales Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate enableQueryTracking` *`ASO_CUBE_NAME`*, donde *`ASO_CUBE_NAME`* es el nombre del cubo de ASO en el que se va a activar el seguimiento de las consultas.

Ejemplo

```
epmautomate enableQueryTracking VISION_ASO
```

encrypt

Usa el estándar de cifrado avanzado (`AES/CBC/PKCS5Padding(256)`) para cifrar la contraseña de Oracle Fusion Cloud Enterprise Performance Management (o el símbolo de refrescamiento y el ID de cliente de OAuth2.0 para acceder a entornos de OCI (Gen 2)) y, opcionalmente, la contraseña del servidor proxy de Internet usada para la conexión a los entornos de Oracle Fusion Cloud EPM y la almacena en un archivo de contraseñas.

El cifrado de los secretos permite a los administradores del servicio compartir su archivo de contraseñas cifradas con los desarrolladores que escriben scripts de EPM Automate para que puedan ejecutar los scripts. De esta forma se elimina la necesidad de compartir la contraseña del administrador del servicio o de crear una cuenta de Cloud EPM genérica y compartida específicamente para ejecutar los scripts.

Su contraseña solo se cifrará una vez.

> **Nota:**
>
> Consulte Manejo de caracteres especiales para obtener información sobre el cifrado de contraseñas que contengan caracteres especiales.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Sintaxis

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

- *PASSWORD|REFRESH_TOKEN PASSWORD* es la contraseña o el símbolo de refrescamiento de OAuth que desea cifrar. No puede utilizar credenciales corporativas con EPM Automate.
- *KEY* es la clave privada que se utilizará para cifrar la contraseña.
- *PASSWORD_FILE* es el nombre y la ubicación del archivo en el que se almacena la contraseña cifrada o el símbolo de refrescamiento. El archivo de contraseñas debe llevar la extensión `.epw`.
- `ClientID`, opcionalmente, es el identificador de cliente creado durante la configuración de OAuth 2.0. Este valor se debe especificar al cifrar un símbolo de refrescamiento de OAuth 2.0. No especifique este valor al cifrar una contraseña.
- `ProxyServerPassword` es la contraseña con la que se autentifica el usuario en el servidor proxy HTTP. Necesaria solo si está activada la autentificación en el servidor proxy para la red.

Ejemplos

- **Cifrado solo de la contraseña de Cloud EPM:** `epmautomate encrypt P@ssword1 myKey C:\mySecuredir\password.epw`
- **Cifrado de las contraseñas de Cloud EPM y del servidor proxy de Internet:** `epmautomate encrypt E@xample1 myKey C:\mySecuredir\password.epw ProxyServerPassword=Proxy_Pwd1`
- **Cifrado del símbolo de refrescamiento y el identificador de cliente:** `epmautomate encrypt AAyyilYBAWD4....FVkxefd8kjoJr6HJPA= myEncyprtion42Key C:\mySecuredir\oauthfile1.epw ClientID=6fdf2e72fd343430ABR22394C`

essbaseBlockAnalysisReport

Crea el informe de análisis de bloques de Essbase, que le ayuda a analizar los datos de Oracle Essbase para soportar el ajuste de cubos de Opción de almacenamiento de bloques (BSO), por lo general, usados para cálculos, en su aplicación.

El informe de análisis de bloques de Essbase resulta útil para resolver los problemas de rendimiento derivados de patrones de datos, como los números repetidos en cubos BSO de Essbase. El informe proporciona información sobre estas tres áreas:

1. **Porcentaje de bloques solo con cero**: muestra los bloques que contienen solo ceros como porcentaje de todos los bloques incluidos en el archivo de exportación.
2. **10 principales valores de celda numérica repetidos por porcentaje de celdas numéricas**: en esta tabla se muestran los 10 principales valores repetidos como porcentaje de todos los valores del archivo de exportación.

3. **100 principales combinaciones de miembros densos con valores repetidos**: en esta tabla se muestran las 100 principales combinaciones de miembros densos con valores repetidos en el cubo. En la columna **Valor de celda** se muestra un valor para cada miembro para que aparezca en la jerarquía como una columna diferente. Por ejemplo, si el periodo está en la columna, habrá una columna diferente para enero, febrero, etc. Otras dimensiones densas aparecen en las filas. Esto debería ayudarle a identificar las ubicaciones de los valores repetidos.

Antes de ejecutar este informe, utilice el comando exportEssbaseData para exportar los datos del cubo para el que desea crear el informe de análisis de bloques en un archivo zip. Puede exportar los datos de nivel 0 o de todos los niveles, según sea necesario. Ejecute este comando para crear el informe de análisis de bloques para este archivo zip. El informe se crea en el buzón de salida; puede utilizar el comando downloadFile para descargarlo en un equipo local o el comando sendMail para enviarlo por correo electrónico.

Se aplica a

Financial Consolidation and Close, Planning, Planning Modules, FreeForm, Strategic Workforce Planning, Sales Planning y Tax Reporting.

Roles necesarios

Administrador del servicio

Sintaxis

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

- `EXPORT_DATA_FILE.zip` es el nombre del archivo zip que contiene los datos de Essbase exportados anteriormente de un cubo BSO mediante el comando exportEssbaseData .
- `REPORT_FILE` es el nombre del archivo de informe de análisis de bloques con formato HTML.

Ejemplo

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

executeAggregationProcess

Inicia el proceso de agregación, usando opcionalmente estadísticas de seguimiento de consultas, para mejorar el rendimiento de los cubos de ASO. Este es un paso importante a la hora de optimizar los cubos de ASO.

Antes de ejecutar este comando:

- Use el comando enableQueryTracking para capturar las estadísticas de recuperación de datos para optimizar la agregación de ASO.
- Deje suficiente tiempo para permitir que el proceso de negocio capture los patrones de recuperación de los datos (consultas) que se pueden utilizar para crear vistas de agregado.

Se aplica a

Planning, Planning Modules, FreeForm, Enterprise Profitability and Cost Management, Strategic Workforce Planning y Sales Planning.

Roles necesarios

Administrador del servicio

Sintaxis

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

- `useQueryData` para usar los datos de consulta registrados y seleccionar el conjunto más adecuado de vistas de agregado. El valor predeterminado es `false`.
- `includeAlternateRollups` para incluir las jerarquías secundarias (con un uso de nivel predeterminado) en el proceso de selección de vistas. El valor predeterminado es `disable`.
- `growthSizeRatio`, opcionalmente, es el ratio de aumento máximo del cubo para agregar las vistas que seleccione el servidor. El aumento del cubo se detendrá cuando el aumento máximo alcance el ratio que especifique. La configuración predetermina permite que el tamaño del cubo aumente sin ningún límite de ratio de aumento.

> **Note:**
>
> Para crear vistas de agregado predeterminadas, ejecute este comando sin especificar parámetros opcionales.

Ejemplos

- Cree una vista de agregado basada en los datos de consultas capturados con el comando enableQueryTracking:
  ```
  epmautomate executeAggregationProcess VISION_ASO useQueryData=true
  includeAlternateRollups=enable
  ```
- Cree una vista de agregado predeterminada:
  ```
  epmautomate executeAggregationProcess Vis1ASO
  ```

executeBurstDefinition

Ejecuta una definición de repartición que especifica los artefactos, puntos de vista y otras opciones necesarias para ejecutar informes o libros para más de un miembro de una sola dimensión para un origen de datos.

Se aplica a

Narrative Reporting

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

A los usuarios con los roles Usuario avanzado, Usuario y Visor se les debe asignar seguridad adicional mediante una ACL

Sintaxis

`epmAutomate executeBurstDefinition` *`ARTIFACT_NAME`*, donde *`ARTIFACT_NAME`* es el nombre y la ruta de definición de repartición.

Ejemplo

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

executeReportBurstingDefinition

Con una definición de repartición, ejecuta la repartición de un único informe o un libro para más de un miembro de una única dimensión, y publica un PDF o una salida estática (que no se puede refrescar en Oracle Smart View for Office) para cada miembro.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmAutomate executeReportBurstingDefinition` *`BURST_DEFINITION_NAME`* `[jobName=`*`JOB_NAME`*`]`, donde:

- *`BURST_DEFINITION_NAME`* es la ruta y el nombre de una definición de repartición.
- *`JOB_NAME`*, opcionalmente, es el nombre del trabajo que se debe usar para ejecutar la definición de repartición. El valor predeterminado es `Execute Bursting Definition`.

Ejemplo

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

exportAccessControl

Exporta el informe de detalles del usuario, que contiene información sobre los usuarios que tienen roles predefinidos en el entorno y muestra los atributos de cada uno de los usuarios (como el nombre y el correo electrónico), así como información sobre su acceso (como una asignación a grupos, equipos y organizaciones), a un archivo CSV o XLS.

Un informe de ejemplo:

	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			

Puede descargar este informe con el comando downloadFile.

Se aplica a

Account Reconciliation

Roles necesarios

Administrador del servicio

Sintaxis

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

- *`REPORT_NAME`* es el nombre del archivo de exportación que contendrá el informe.
- `reportFormat`, opcionalmente, es el formato de archivo. Los valores válidos son XLS y CSV (valor predeterminado).

Ejemplo

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

exportAppAudit

Exporta registros de auditoría de datos en un archivo ZIP, que puede descargar y archivar en un equipo local. La información de auditoría está disponible durante 365 días en el entorno. Puede generar este informe para un número de días o un rango de fechas.

El primer carácter del archivo CSV de salida es el carácter de marca de orden de bytes (BOM) `\ufeff` seguido de un identificador cifrado entre comillas dobles. La cabecera del archivo CSV sigue al identificador de aplicación.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

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

- *`EXPORT_FILE_NAME`* es el nombre del archivo ZIP en el que se almacenarás los datos de auditoría exportados. Utilice el comando downloadFile para descargar archivos de un entorno.
- `userNames`, opcionalmente, es una lista separada por comas de nombres de usuario. Si se especifica, solo se exportarán los datos de auditoría creados por estos usuarios. No especifique este valor si desea exportar los datos de auditoría para todos los usuarios.
- `nDays`, opcionalmente, identifica el número de días para el que se van a exportar los registros de auditoría. El valor predeterminado es siete días. Los valores posibles son: `all` para exportar todos los datos de auditoría disponibles de los últimos 365 días, `1`, `2`, `7`, `30`, `60` y `180`. No especifique este valor si está generando el informe para un rango de fechas.

- `startDate` y `enddate`, opcionalmente, identifican las fechas de inicio y fin para las que se van a exportar los registros de auditoría. Estos valores se deben especificar con el formato *AAAA-MM-DD* .
  - La fecha de inicio debe ser anterior a la fecha de finalización. Para generar un informe, se deben especificar tanto la fecha de inicio como la de finalización.
  - Estas fechas no pueden ser fechas en el futuro.
  - Estas fechas se ignoran si se especifica el valor opcional `nDays`.
- Con `excludeApplicationId`, opcionalmente, se identifica si el identificador de la aplicación se va escribir en el archivo de exportación. El valor predeterminado es `false`.

> **Nota:**
>
> Los datos de archivos exportados que no contengan el identificador de aplicación no se pueden importar en entornos de Oracle Fusion Cloud Enterprise Performance Management.

Ejemplos

- Exportar los datos de auditoría de los usuarios especificados de los últimos 30 días con el identificador de aplicación:
  ```
  epmautomate exportAppAudit auditData userNames=johnDoe,jane.doe@example.com
  ndays=30
  ```
- Exportar los datos de auditoría de los usuarios especificados de los últimos 30 días sin el identificador de aplicación:
  ```
  epmautomate exportAppAudit auditData userNames=johnDoe,jane.doe@example.com
  ndays=30 excludeApplicationId=true
  ```
- Exportar los datos de auditoría de agosto de 2024 de todos los usuarios con el identificador de aplicación:
  ```
  epmautomate exportAppAudit auditData startDate=2024-08-01 endDate=2024-08-31
  ```

exportAppSecurity

Exporta las asignaciones de acceso a nivel de artefacto (ACL) a un archivo CSV, que puede descargar para almacenamiento local.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate exportAppSecurity` *`EXPORT_FILE_NAME`*`.CSV` donde *`EXPORT_FILE_NAME`* es el nombre del archivo en el que se almacenarán los datos de seguridad exportados. Este archivo se creará en el buzón de salida, desde donde puede descargarlo en su computadora.

Ejemplo

```
epmautomate exportAppSecurity app_security.CSV
```

exportARApplicationProperties

Exporta la configuración de la aplicación de Account Reconciliation (relacionada con la experiencia Redwood, el tema, la notificación por correo electrónico y el nombre de proceso de negocio), la imagen de fondo y la imagen del logotipo a un archivo JSON para que pueda importarla en el mismo entorno o en otro.

Este comando resulta útil al importar una aplicación del entorno de producción al de prueba. Si la configuración de la aplicación varía del entorno de producción al de prueba, puede exportarla del entorno de prueba antes de importar la aplicación desde el entorno de producción y, a continuación, importar la configuración en el entorno de prueba para mantener los ajustes originales.

Se aplica a

Account Reconciliation

Roles necesarios

Administrador del servicio

Sintaxis

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

- *FILE_NAME* es el nombre del archivo JSON en el que se almacenarán los valores de propiedad exportados.
  Puede descargar el archivo de exportación con el comando downloadFile. Utilice el comando uploadFile para cargarlo en el entorno de destino y, a continuación, ejecute el comandoimportARApplicationProperties para restaurar esta configuración en el entorno de destino.
- `Properties`, opcionalmente, es una lista de propiedades que exportar separadas por comas. Puede exportar algunas o todas las propiedades siguientes. Si se omite esta propiedad, se exportan todas estas propiedades:
  - `Theme`: exporta el tema de visualización usado en el entorno.
  - `EmailNotification`: exporta la notificación por correo electrónico definida en el entorno.
  - `DisplayBusinessProcessName`: exporta si se va a mostrar el nombre del proceso de negocio en la página del entorno.
  - `RedwoodExperience`: exporta el valor Experiencia Redwood del entorno.
  - `BackgroundImage`: exporta la imagen de fondo que se usa en el entorno
  - `LogoImage`: exporta la imagen de registro usada en el entorno.

Ejemplo

Solo exporta la configuración de experiencia de Redwood y notificación por correo electrónico, así como la imagen del logotipo desde un entorno:

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

exportBackgroundImage

Exporta la imagen de fondo usada en un entorno de Account Reconciliation a un archivo JPG para que pueda importarla en otro entorno.

Se aplica a

Account Reconciliation

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate exportBackgroundImage` *`IMAGE_NAME`*`.jpg`, donde *`IMAGE_NAME`* es el nombre del archivo de imagen de fondo.
Puede descargar el archivo de imagen con el comando downloadFile. Utilice el comando uploadFile para cargarlo en el entorno de destino y, a continuación, ejecute el comando importBackgroundImage para importarlo.

Ejemplo

```
epmautomate exportBackgroundImage corpImage.jpg
```

exportCellLevelSecurity

Exporta la configuración de seguridad de nivel de celda del proceso de negocio en un archivo ZIP, que puede descargar en un equipo local con el comando downloadFile.

Se aplica a

Planning, Planning Modules, FreeForm, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate exportCellLevelSecurity` *`FILE_NAME`*`.ZIP [names=`*`SECURITY_RECORD_NAMES`*`]`, donde:

- `FILE_NAME` es el nombre del archivo ZIP que se creará para incluir el archivo Excel que contiene la información de seguridad a nivel de celda.
- `names`, opcionalmente, identifica una lista de definiciones de seguridad a nivel de celda separadas por comas en la aplicación. Si no se proporciona esta opción, se exportan todas las definiciones de seguridad de nivel de celda en la aplicación.

Ejemplos

- **Exportar definiciones de seguridad específicas a nivel de celda**
  ```
  epmautomate exportCellLevelSecurity ExportCLSDRecordsFile.zip
  names=CLSDAccountPeriod,CLSDEntityPeriod,CLSDProductPeriod
  ```

- **Exportar todas las definiciones de seguridad a nivel de celda**
  `epmautomate exportCellLevelSecurity ExportCLSDRecordsFile.zip`

exportConsolidationJournals

Exporta los asientos de consolidación utilizando un trabajo definido en Financial Consolidation and Close.

Se aplica a

Financial Consolidation and Close

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate exportConsolidationJournals jobName [fileName=FILE_NAME]` donde

- `jobName` es el nombre de un trabajo de exportación de asientos creado en Financial Consolidation and Close.
- Opcionalmente, `fileName`, es el nombre de un archivo .JLF al que se exportarán los asientos. Utilice el comando downloadFile para descargar este archivo en un equipo local.

Ejemplo

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

exportData

Exporta datos de la aplicación a un archivo ZIP utilizando la configuración de exportación de datos, incluido el nombre de archivo, especificada en un trabajo de tipo `export data`.

El archivo de datos exportado se almacena en la ubicación de descarga predeterminada desde donde lo puede descargar en su equipo. Utilice el explorador de bandeja de entrada/buzón de salida para ver los detalles del archivo exportado.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate exportData JOB_NAME [FILE_NAME]` donde *JOB_NAME* es el nombre de un trabajo definido en la aplicación y *FILE_NAME* es el nombre del archivo ZIP (opcional) al que se exportarán los datos.

Ejemplo

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

exportDataManagement

Exporta registros de Data Management de un entorno en un archivo ZIP.

Con este comando se exporta un conjunto completo de datos de configuración y temporales, incluidas las columnas ID a un archivo ZIP para que los datos se puedan importar sin perder la integridad referencial.

El archivo exportado, por ejemplo `dataFile.zip`, está almacenado en la bandeja de salida. Puede descargar el archivo exportado con el comando downloadFile, por ejemplo, `epmAutomate downloadFile outbox/dataFile.zip`. Puede usar este archivo ZIP para importar los datos con el comando importDataManagement.

Se aplica a

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

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

`epmautomate exportDataManagement` *`FILE_NAME`*`.zip`, donde *`FILE_NAME`* es el nombre del archivo ZIP al que se exportan los datos.

Ejemplo

```
epmautomate exportDataManagement dataFile.zip
```

exportDimension

Exporta una dimensión de una aplicación de Oracle Fusion Cloud Enterprise Data Management a un archivo en el área temporal u, opcionalmente, a un entorno de destino definido en una conexión.

Se aplica a

Oracle Enterprise Data Management Cloud

Roles necesarios

- Administrador del servicio
- Usuario (necesita el permiso de gestor de datos)

Sintaxis

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

- `APPLICATION` es el nombre de una aplicación de Oracle Enterprise Data Management Cloud
- `DIMENSION` es el nombre de una dimensión de aplicación

- `FILE_NAME` es el nombre del archivo (CSV para exportar a un archivo o ZIP para exportar a Oracle Financials Cloud) para almacenar los datos exportados. Si el valor del parámetro connection no está establecido, este archivo se crea en el área temporal. Puede descargarlo en un equipo local mediante el comando downloadFile o copiarlo en otro entorno de Oracle Fusion Cloud Enterprise Performance Management mediante el comando copyFileFromInstance.
- Opcionalmente, `connection=NAME` identifica un nombre de conexión (ubicación de instancia) definido en Oracle Enterprise Data Management Cloud. Si se ha especificado, el archivo de exportación se carga en el entorno de destino (la bandeja de entrada para Cloud EPM y la ubicación de carga predeterminada para Oracle Financials Cloud).

> **Nota:**
>
> Las credenciales especificadas en la definición de conexión deben tener derechos de acceso para escribir en el entorno de destino.

Ejemplos

- Exporte al área temporal de Oracle Enterprise Data Management Cloud: `epmautomate exportDimension USOperations Entity EntityData.CSV`
- Exporte y cargue en Oracle Financials Cloud: `epmautomate exportDimension USOperations Entity EntityData.zip Connection=ora_fusion_gl`
- Exporte y cargue en la bandeja de entrada de Cloud EPM de destino: `epmautomate exportDimension USOperations Entity EntityData.CSV Connection=EPM_cloud_pln`

exportDimensionMapping

Exporta las reglas de asignación de una dimensión de Oracle Fusion Cloud Enterprise Data Management específica para una ubicación a fin de crear un archivo de reglas de asignación y, opcionalmente, carga el archivo exportado en la bandeja de entrada de Data Management de otro entorno de Oracle Fusion Cloud Enterprise Performance Management.

Se aplica a

Oracle Enterprise Data Management Cloud

Roles necesarios

- Administrador del servicio
- Usuario (con permiso de gestor de datos)

Sintaxis

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

- `APPLICATION` es el nombre de una aplicación de Oracle Enterprise Data Management Cloud
- `DIMENSION` es el nombre de una dimensión de aplicación
- `LOCATION` es la ubicación específica de para la que se deben exportar las reglas de asignación.

- `FILE_NAME` es el nombre del archivo CSV para almacenar las asignaciones exportadas. Este archivo se crea en el área temporal si no se ha establecido el parámetro `connection`; puede descargarlo en un equipo local mediante el comando downloadFile o utilizar el comando copyFileFromInstance para copiar el archivo en otro entorno de Cloud EPM.
- Opcionalmente, `connection=NAME` identifica un nombre de conexión (ubicación de instancia) definido en Oracle Enterprise Data Management Cloud. Si se ha especificado, el comando carga el archivo exportado en la ubicación de carga predeterminada del entorno de destino.

> **Nota:**
>
> Las credenciales especificadas en la conexión deben tener derechos de acceso para escribir en el entorno de destino.

Ejemplos

- Exporta al área temporal: `epmautomate exportDimensionMapping USOperations Entity Loc1 Loc1Mappings.CSV`
- Exporta y carga en la bandeja de entrada del entorno de Cloud EPM de destino: `epmautomate exportDimensionMapping USOperations Entity Loc1 Loc1Mappings.CSV Connection=EPM_cloud_pln`

exportEJJournals

Exporta asientos de Enterprise Journals que estén listos para contabilizarse desde Financial Consolidation and Close en un archivo ZIP. A continuación, este archivo se puede usar para importar datos de asientos en un sistema ERP.

Después de exportar los asientos a un archivo de exportación, con este comando se actualiza el estado de contabilización de cada asiento exportado de `Ready To Post` a `Post In Progress`.

Se aplica a

Financial Consolidation and Close

Roles necesarios

Administrador del servicio

Sintaxis

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

- *`FILE_NAME`* identifica un archivo ZIP en el que se van a archivar los archivos CSV de la exportación de asientos. El comando genera un archivo CSV (el formato de nombre es `YEAR_PERIOD_JOURNALID_YYYYDDMMHHMMSS.csv`) para cada asiento y, a continuación, los comprime para crear este archivo ZIP.
- *`YEAR`*, opcionalmente, es el año de recopilación de datos desde el que se exportan los datos de asientos. Si no se especifica, se exportan los datos de todos los años.

- *PERIOD*, opcionalmente, es el periodo de recopilación de datos desde el que se exportan los datos de asientos. Solo se puede establecer si se especifica un año de recopilación de datos. Si no se especifica un valor, se exportan los datos de todos los periodos.

> **Note:**
>
> Si no se especifican *YEAR* y *PERIOD*, con este comando se exportan todos los asientos que tengan un estado de contabilización `Ready To Post` en todos los años y periodos.

- `Operation`, opcionalmente, es la operación que se va a realizar. Los valores posibles son `posting` y `validation`. El valor predeterminado es `posting`.
  `operation=posting` proporciona un archivo CSV que contiene los resultados de la contabilización de asientos con estas columnas: Year, Period, Journal ID, Posting Status (Posted/Failed), Message, Error Code y Error Message. Las columnas Message, Error Code y Error Message son opcionales. El estado de contabilización de los asientos se actualiza en función del estado proporcionado en el archivo. Solo se actualizarán los asientos que tengan un estado de contabilización *Contabilización en curso* .

  `operation = validation` genera un archivo CSV que contiene los resultados de la validación de asientos con estas columnas: Year, Period, Journal ID, Validation Status (Valid/Failed), Message, Error Code y Error Message. Las columnas Message, Error Code y Error Message son opcionales. El estado de validación de los asientos se actualiza en función del estado proporcionado en el archivo. Solo se actualizarán los asientos que tengan un estado de validación *Validación en curso* .

Ejemplos

- Exporte los datos de asientos de todos los años y periodos:
  `epmautomate exportEJJournals Journal_Export.zip`
- Exporte los datos de asientos de un año específico.
  `epmautomate exportEJJournals Journal_Export.zip year=2020`
- Exporte los datos de Journal para una combinación específica de año y periodo:
  `epmautomate exportEJJournals Journal_Export.zip year=2024 period=March`
- Exportar los datos de Journal para una combinación específica de año y periodo, realizando una operación de validación:
  `epmautomate exportEJJournals Journal_Export.zip year=2024 period=March Operation=validation`

exportEssbaseData

Exporta datos de un cubo de aplicación a un archivo. Puede exportar solo los datos de nivel 0 (cubos ASO y BSO) o todos los datos en el cubo (cubos BSO).

Utilice el archivo exportado para analizar los datos de Oracle Essbase para patrones; por ejemplo, para ayudar a mejorar el rendimiento.

> **Note:**
>
> - Los archivos de datos incluidos en el archivo tienen la extensión `.txt`.
> - Los objetos de texto de celda y de listas inteligentes se exportan como UUID.

La ejecución de una exportación de datos pone el cubo en modo de solo lectura e impide cualquier actividad de escritura mientras la operación de exportación está en curso.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

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

- *`CUBE_NAME`* identifica el cubo desde el que se van a exportar los datos.
- *`FILE_NAME`* es el nombre del archivo ZIP que contendrá los datos exportados. Puede descargar este archivo ejecutando el comando downloadFile.
- `level`, opcionalmente, identifica el nivel de datos que se va exportar. El valor predeterminado es `0`.
  - **Cubos ASO:** especifique `0` para exportar datos de nivel 0. No puede usar la opción `Todo`.
  - **Cubos BSO:** especifique `0` para exportar datos de nivel 0 o `Todo` para exportar todos los datos.

Ejemplos

- Exportar todos los datos desde un cubo BSO:
  ```
  epmautomate exportEssbaseData Report1 Report1_all_data.zip level=All
  ```
- Exportar datos de nivel 0 desde un cubo:
  ```
  epmautomate exportEssbaseData Plan1 Plan1_lvl0_data.zip
  ```

exportJobConsole

Exporta los registros de la consola de trabajos a un archivo CSV y crea un archivo ZIP de exportación.

El primer carácter del archivo CSV de salida es el carácter de marca de orden de bytes (BOM) `\ufeff` seguido de un identificador cifrado entre comillas dobles. La cabecera del archivo CSV sigue al identificador de aplicación.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

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

- *`FILE_NAME`* es el nombre del archivo ZIP en el que se almacenarán los registros de la consola de trabajos exportados. Utilice el comando downloadFile para descargar este archivo de un entorno.
- Con `nDays`, opcionalmente, se identifica el número de días para el que se van a exportar los registros de la consola de trabajos. Los valores posibles son: `all` (todo en minúsculas) para exportar todos los registros de la consola de trabajos disponibles, `1`, `2`, `7`, `30` y `60`. El valor predeterminado es `7`.
- `jobTypes`, opcionalmente, es una lista separada por comas de códigos de trabajo para los que se deben exportar los registros de la consola. El valor predeterminado es `Rules`. Los valores válidos son:
  - `all` (todo en minúscula)
  - `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`, opcionalmente, es una lista separada por comas de códigos de estado de trabajo para los que se van a exportar los registros. El valor predeterminado es `2` (Completado correctamente). Los valores posibles son los siguientes:
  - `all` (todo en minúscula) para todos los trabajos de cualquier estado
  - `1`: Procesando
  - `2`: Completado correctamente
  - `3`: Fallo con errores
  - `4`: Completado con estado desconocido
  - `5`: Completado con una violación de umbral
  - `6`: Cancelación pendiente
  - `7`: Cancelado
  - `8`: Completado con errores
  - `9`: Completado con advertencias
- *`exportErrorDetails`*, opcionalmente, exporta los detalles de los trabajos con fallos o que notificaron un error en archivos log si este valor se establece en `true`. Este archivo log de error se incluye en el archivo ZIP de salida. El valor predeterminado es `false`. Si este valor se establece en `true`, se exportan los detalles de estado de los trabajos con el siguiente estado.
  - Fallo con errores
  - Completado con estado desconocido
  - Completado con una violación de umbral
  - Completado con errores
  - Completado con advertencias
- Con `excludeApplicationId`, opcionalmente, se identifica si el identificador de la aplicación se va escribir en el archivo de exportación. El valor predeterminado es `false`.

> **Note:**
>
> Los datos de archivos exportados que no contengan el identificador de aplicación no se pueden importar en entornos de Oracle Fusion Cloud Enterprise Performance Management.

Ejemplos

- Exporte todos los registros de la consola de trabajos disponibles:
  ```
  epmautomate exportJobConsole jobs.zip nDays=all jobTypes=all
  jobStatusCodes=all
  ```

- Exporte todos los registros de la consola de trabajos de reglas disponibles:
  `epmautomate exportJobConsole jobs.zip nDays=all jobStatusCodes=all`
- Exporte todos los registros de la consola de trabajos de reglas disponibles sin identificador de aplicación:
  `epmautomate exportJobConsole jobs.zip nDays=all jobStatusCodes=all excludeApplicationId=true`
- Exporte solo los registros del trabajo Reglas que haya finalizado correctamente en los últimos 14 días:
  `epmautomate exportJobConsole jobs.zip nDays=14`
- Exporte los registros de la consola y los errores de los trabajos de importación de metadatos y borrado de cubos con los estados de fallo Con errores o Completado con errores y que se hayan ejecutado en los últimos siete días:
  `epmautomate exportJobConsole jobs.zip jobtypes=IMPORT_METADATA,CLEAR_CUBE jobStatusCodes=3,8 exportErrorDetails=true`

exportLibraryArtifact

Exporta artefactos de biblioteca de Narrative Reporting. Opcionalmente, solo para artefactos de informes, este comando puede convertir la exportación en un archivo LCM que puede importar en Financial Consolidation and Close, Planning, Planning Modules o Tax Reporting.

Al finalizar la exportación, utilice el comando downloadFile para descargar los archivos de exportación y de errores en un equipo local.

Se aplica a

Narrative Reporting

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

A los usuarios con los roles Usuario avanzado, Usuario y Visor se les debe otorgar seguridad adicional mediante una ACL

Sintaxis

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

- *`ARTIFACT_PATH`* es la ubicación del artefacto en la biblioteca de Narrative Reporting.
- *`EXPORT_FILE`* es un nombre único para el archivo al que se exportarán los artefactos.
- `exportFormat`, opcionalmente, es uno de los siguientes:
  - Con `Native` se exportan los artefactos como archivo zip que se puede usar con otros entornos de Narrative Reporting. Este es el valor predeterminado.
  - Con `File` se exportan los archivos en el formato binario original (PDF, DOCX, Zip, JPEG, etc.) en que estén disponibles en Narrative Reporting. Este parámetro se puede usar para exportar solo archivos binarios; no se debe usar con artefactos de informes.

  - `LCM` convierte los informes en el formato que usa Migración y los exporta en un archivo ZIP que se puede importar en entornos de Financial Consolidation and Close, Planning, Planning Modules o Tax Reporting.

- `applicationName`, opcionalmente, es el nombre de la aplicación de destino en la que tiene previsto importar los informes. Este valor solo es necesario si está usando `LCM` como valor del parámetro `exportFormat`.
- Opcionalmente, *`errorFile`* es un nombre único para el archivo de texto en el que se almacenarán los errores relacionados con la exportación.

Ejemplos

- Exporte un informe con su formato nativo para que se pueda importar en otro entorno de Narrative Reporting:
```
epmautomate exportLibraryArtifact "Library/Samples/Sample Report 1"
exp_SampleReport1.doc errorFile=export_errors.txt
```
- Exporte una hoja de cálculo en su formato binario original:
```
epmautomate exportLibraryArtifact "Library/Spreadsheets/Sheet1.xlsx"
exp_Sheet1.xlsx exportFormat=File errorFile=export_errors.txt
```
- Exporte informes y aplíqueles formato para importarlos en Financial Consolidation and Close, Planning, Planning Modules o Tax Reporting:
```
epmautomate exportLibraryArtifact "Library/Samples/Sample Report 1"
exp_SampleReport1.zip exportFormat=LCM applicationName=Vision
errorFile=report_exp_errors.txt
```

exportLibraryDocument

Exporta los documentos disponibles en la biblioteca Informes a un archivo.

Puede descargar el archivo exportado en un equipo local mediante el comando downloadFile.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

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

- *`ARTIFACT_PATH`* es la ubicación o el identificador único universal (UUID) del documento en la biblioteca Informes. Para exportar varios documentos, especifique una lista separada por comas de UUID o ubicaciones del documento.
- De manera opcional, *`jobName`* es el nombre del trabajo de exportación de documentos de biblioteca que se utilizará para exportar el documento. El nombre predeterminado del trabajo es `Copy Artifact From Library`.
- Opcionalmente, *`EXPORT_FILE`* es un nombre único para el archivo de exportación. Si utiliza `exportFormat=ZIP`, el archivo ZIP resultante utiliza este nombre. Asegúrese de utilizar la extensión .zip en este valor.

Si utiliza `exportFormat=file`:

  - Para exportar un documento: el archivo de exportación se crea con este nombre si la extensión de archivo especificada coincide con la del documento.
  - Para exportar varios documentos: el comando ignora este valor y exporta cada documento con el nombre original.

- `exportFormat`, opcionalmente, es uno de los siguientes:
  - `File` exporta los documentos en el formato binario original (PDF, DOCX, Zip, JPEG, etc.) en el que están disponibles en la biblioteca. Cada archivo exportado se crea en el buzón de salida con el nombre original. Este es el valor predeterminado.
  - `zip` crea un archivo ZIP que contiene todos los documentos exportados en su formato binario original. Si no se especifica ningún valor para el parámetro opcional `exportFile`, el nombre de archivo zip se genera automáticamente con el nombre de usuario del usuario que ejecuta el comando y el registro de hora. El formato de nombre de archivo es *nombreusuario_exported_artifacts_aaaaMMddHHmm.zip*, por ejemplo, *JDoe_exported_artifacts_202410201559.zip*.
- De manera opcional, *`errorFile`* es un nombre único para el archivo en el que se almacenarán los errores relacionados con la exportación. No se creará ningún archivo de error si no especifica este valor.
- De manera opcional, `overwrite` controla si se debe sobrescribir un archivo con el mismo nombre situado actualmente en la ubicación de descarga. El valor predeterminado es `false`, que significa que el comando fallará si existe un archivo con el mismo nombre en el buzón de salida.

Ejemplos

- Exportar un documento:
```
epmautomate exportLibraryDocument Library/folder1/WeeklySales.html
jobName="Copy Weekly Sales" exportFile=WeeklySales.zip
errorFile=WeeklySalesError.log overWrite=true exportFormat=zip
```
- Exportar varios documentos:
```
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

Exporta el logotipo corporativo usado en un proceso de negocio de Account Reconciliation a un archivo JPG para que pueda importarlo en otro entorno.

Se aplica a

Account Reconciliation

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate exportLogoImage` *`IMAGE_NAME`*`.jpg`, donde *`IMAGE_NAME`* es el nombre del archivo de imagen del logotipo.

Puede descargar el archivo de logotipo exportado con el comando downloadFile. Utilice el comando uploadFile para cargarlo en el entorno de destino y, a continuación, ejecute el comando importLogoImage.

Ejemplo

```
epmautomate exportLogoImage corpLogo.jpg
```

exportMapping

Exporta las reglas de asignación de una dimensión o ubicación específicas para crear un archivo de reglas de asignación. Debe especificar el nombre de archivo y una ubicación dentro del buzón de entrada (por ejemplo, `inbox/exportedAccountMap.txt` o `inbox/france sales/exportedAccountMap.txt`) para exportar las asignaciones.

Utilice el comando downloadFile para descargar el archivo de asignación exportado en un equipo local.

Se aplica a

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

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

`epmautomate exportMapping` *`DIMENSION_NAME|ALL FILE_NAME LOCATION`* donde:

- *`DIMENSION_NAME|ALL`* es la dimensión de origen desde la que se van a exportar las asignaciones. Especifique el nombre de la dimensión desde la que se van a exportar las asignaciones u `ALL` para exportar asignaciones desde todas las dimensiones de una ubicación.
- *`FILE_NAME`* es un nombre único para el archivo de asignación y una ubicación dentro del buzón de salida.
- *`LOCATION`* es la ubicación de Data Management para la que se deben exportar las reglas de asignación.

Ejemplos

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

exportMetadata

Exporta metadatos a un archivo utilizando la configuración especificada en un trabajo de tipo `export metadata`. El archivo que contiene los datos exportados se almacena en la ubicación de descarga predeterminada desde donde lo puede descargar en un equipo local.

Opcionalmente, puede especificar un nombre de archivo para los datos exportados, lo que sustituirá el nombre de archivo predeterminado (nombre del trabajo que se utiliza para exportar metadatos). Los metadatos se exportarán solo como un archivo ZIP.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate exportMetadata` *`JOB_NAME`* `[`*`FILE_NAME`*`]` donde *`JOB_NAME`* es el nombre de un trabajo definido en la aplicación y *`FILE_NAME`* es el nombre del archivo ZIP al que se van a exportar los metadatos.
Utilice el comando downloadFile para descargar este archivo en un servidor local.

Ejemplo

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

exportOwnershipData

Exporta datos de propiedad desde una entidad a un archivo CSV delimitado por comas.

Los datos de propiedad predeterminados que haya rellenado Financial Consolidation and Close no se incluyen en el archivo de exportación. Solo los datos que haya introducido el usuario para reemplazar la configuración predeterminada se incluyen en el archivo de exportación.

Se aplica a

Financial Consolidation and Close y Tax Reporting.

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario

Sintaxis

`epmautomate exportOwnershipData` *`Entity Scenario Year Period FILE_NAME`* donde:

- `Entity` es el nombre de la entidad de la que se van a exportar los datos.
- `Scenario` es el escenario del que se van a exportar los datos.
- `Year` es el año del que se van a exportar los datos.
- `Period` es el periodo del año del que se van a exportar los datos.
- `FILE_NAME` es el nombre de un archivo CSV al que se van a exportar los datos. Utilice el comando downloadFile para descargar este archivo en un servidor local.

Ejemplo

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

exportQueryResults

Ejecuta una consulta definida en una aplicación y exporta los resultados a un archivo de texto.

El archivo de resultados de la consulta se almacena en profitoutbox; puede descargarlo mediante el comando downloadFile o mediante el explorador de archivos de Profitability and Cost Management.

Se aplica a

Profitability and Cost Management

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Sintaxis

`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]`, donde:

- *`APPLICATION_NAME`* es el nombre de la aplicación de Profitability and Cost Management para la que desea ejecutar la consulta.
- `fileName` es el nombre del archivo que almacenará los resultados de la consulta. Se necesita este valor de parámetro si no se está especificado el valor del parámetro `queryName`. Es opcional si se especifica el valor de parámetro `queryName`, en cuyo caso, se usa el nombre de consulta como nombre del archivo de resultados de la consulta.
  El formato de datos que especifique determina el formato del archivo de salida. Si usa `dataFormat=NATIVE` (valor predeterminado), el proceso de exportación crea un archivo de texto. Si usa `dataFormat=COLUMNAR`, el proceso de exportación crea varios archivos de texto con numeración secuencial y los comprime en un archivo Zip.
- `fileOutputOptions`, opcionalmente, identifica el formato de salida del archivo de resultado de la consulta. El valor predeterminado es `ZIP_ONLY`, que crea `fileName`.ZIP o `queryName`.ZIP, en función de si se ha especificado un valor para el parámetro `fileName`. Otras opciones son `TEXT_ONLY` para crear el archivo de salida como un archivo de texto y `ZIP_AND_TEXT` para generar un archivo de texto y un archivo zip.
- `queryName` es un parámetro opcional que identifica una consulta que se define en la aplicación. Los nombres de las consultas que contienen los caracteres de espacio deben ir entre comillas dobles.
  No especifique un nombre de consulta si desea exportar todos los datos de Oracle Essbase que pertenecen a la aplicación.

  Las siguientes condiciones pueden provocar que este comando cree un archivo de datos vacío:

  - Una consulta con un formato incorrecto que no recupera ningún dato.
  - Una consulta que genera demasiados datos. En este caso, puede reducir el ámbito de la consulta de forma que recupere menos datos o divida la consulta en varias consultas más pequeñas.
    Consulte Gestión de consultas de Oracle Profitability and Cost Management Cloud en *Administración de Profitability and Cost Management*.
- Con `exportOnlyLevel0Flg`, opcionalmente, se especifica si la consulta solo debe recuperar datos de Level0. Especifique este valor de parámetro en minúscula.
  Este parámetro se ignora si va a exportar todos los datos de la aplicación omitiendo el nombre de la consulta.
- `roundingPrecision`, de forma opcional, especifica el número de posiciones decimales (con precisión de redondeo) que se va a utilizar al exportar los resultados de la consulta. Solo es aplicable si se ha especificado `queryName`. El valor predeterminado es 2.
- Con `dataFormat`, opcionalmente, se identifica el formato de salida. Los valores válidos son:
  - `NATIVE`, que mantiene el resultado de la consulta como datos con formato nativo de Essbase. Este es el valor predeterminado.
  - `COLUMNAR`, que convierte los datos con formato nativo de Essbase y los ordena en columnas para que sea más fácil su interpretación e importación en otras aplicaciones.
    Con esta opción se exportan todos los datos de Essbase y se ignora el valor del parámetro `queryName`. Puede filtrar los datos estableciendo el valor del parámetro `memberFilters`.

> **Nota:**
>
> El comando tiene en cuenta los siguientes parámetros opcionales solo si se especifica `dataFormat` como `COLUMNAR`.

- `memberFilters`, opcionalmente, acepta una cadena con formato JSON para filtrar por dimensión y miembros de Level0. Ejemplo, `"{\"Dim1\":[\"Mem1\"],\"Dim2\":[\"Mem21\",\"Mem22\"]}"`
- `includeHeader`, opcionalmente, agregar nombres de dimensión como cabeceras de columna. Establezca este valor en `false` para excluir la cabecera de la columna. El valor predeterminado es `true`.
- Con `delimiter`, opcionalmente, se identifica el delimitador que se va a usar para separar los miembros de dimensión en el archivo de resultado de la consulta. El delimitador debe ir entre comillas dobles. El valor predeterminado es un espacio (" ").
- Con `keepDuplicateMemberFormat`, opcionalmente, se especifica si se va a imprimir el formato de miembro en formato de miembro duplicado de Essbase, por ejemplo, `[Account]@[Accoun1]`. Establezca este valor en `false` para imprimir solo el nombre del miembro. El valor predeterminado es `true`.

Ejemplos

- Exportar todos los datos de la aplicación:
  ```
  epmautomate exportQueryResults BksML12 fileName="BksML12_MyQuery1.txt"
  fileOutputOptions=TEXT_ONLY
  ```
- Exportar resultados de una consulta específica:

```
epmautomate exportQueryResults BksML12 queryName="My Product Query"
roundingPrecision=3
```

- Exporte datos de Level0 en formato de datos NATIVO:
```
epmautomate exportQueryResults BksML30 fileName="BksML30_ExportLevel0-Data"
fileOutputOptions=ZIP_AND_TEXT exportOnlyLevel0Flg=true
```
- Exporte los datos de Level0 con el formato de datos COLUMNAR con un filtro de una dimensión y un miembro:
```
epmautomate exportQueryResults BksML30 fileName="BksML30_Level0-Data"
dataFormat="COLUMNAR" memberFilters="{\"Period\":[\"December\"]}"
includeHeader="true" delimiter="," roundingPrecision="3"
```
- Exporte los datos de Level0 con el formato de datos COLUMNAR con filtros de una dimensión y varios miembros:
```
epmautomate exportQueryResults BksML30 fileName="BksML30_Level0-Data"
dataFormat="COLUMNAR" memberFilters="{\"Period\":[\"November\",\"December\"]}"
includeHeader="true" delimiter="," roundingPrecision="3"
```
- Exporte los datos de Level0 con el formato de datos COLUMNAR con filtros de varias dimensiones y varios miembros:
```
epmautomate exportQueryResults BksML30 fileName="BksML30_Level0-Data"
dataFormat="COLUMNAR" memberFilters="{\"Year\":[\"2016\"],\"Period\":
[\"November\",\"December\"]}" includeHeader="true" delimiter="="
roundingPrecision="3"
```

exportSnapshot

Repite una operación de exportación realizada anteriormente para crear una instantánea del contenido de Migración.

Mediante Migración, seleccione y exporte los artefactos que desee a una instantánea; por ejemplo, `January16FullApp`. Utilice el nombre de la instantánea con este comando para repetir posteriormente la operación de exportación, que exportará solo los artefactos que se seleccionaron durante la operación de exportación original. Consulte Exportación de artefactos y aplicaciones en *Administración de la migración*.

- Los siguientes elementos no forman parte de las instantáneas de aplicación de Planning, Planning Modules y FreeForm:
  - Datos de auditoría
  - Datos de la consola de trabajos

  Use el comando cloneEnvironment o la función Clonar entorno si desea copiar los datos de auditoría y de la consola de trabajos en el entorno de destino.
- Las instantáneas no contienen los datos de la tabla temporal de Data Management. Para importar estos datos, use los comandos exportDataManagement y importDataManagement o la interfaz de scripts de mantenimientos del sistema de Data Management. Puede usar el comando cloneEnvironment o la función Clonar entorno para crear una copia idéntica del entorno, incluidos los datos de la tabla temporal de Data Management.

Puede descargar la instantánea exportada de la ubicación predeterminada mediante el comando downloadFile.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado y el rol de aplicación Migraciones - Administrar (Oracle Enterprise Data Management Cloud y Profitability and Cost Management)

Sintaxis

`epmautomate exportSnapshot` *`SNAPSHOT_NAME`* donde *`SNAPSHOT_NAME`* es el nombre de una instantánea existente en Migración. Esta instantánea se reemplazará por la nueva instantánea.

Ejemplo

```
epmautomate exportSnapshot January16FullApp
```

exportTemplate

Exporta una aplicación como una plantilla a un archivo .ZIP. El archivo exportado se almacena en profitoutbox.

Puede descargar el archivo exportado en un equipo local mediante el comando downloadFile.

Se aplica a

Profitability and Cost Management

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

`epmautomate exportTemplate` *`APPLICATION_NAME File_Name`* donde:

- *`APPLICATION_NAME`* es el nombre de la aplicación de Profitability and Cost Management que desea exportar como plantilla
- `nombre_archivo` es el nombre del archivo de plantilla

Ejemplo

```
epmautomate exportTemplate BksML12 template1
```

exportTaskManagerAccessControl

Exporta el informe de detalles de usuario para las asignaciones de usuario Gestor de tareas, Datos adicionales y Asiento de Enterprise. El informe contiene información sobre los usuarios que tienen roles predefinidos en el entorno y muestra los atributos de cada uno de los usuarios (como el nombre y el correo electrónico), así como su estado, los equipos, los roles predefinidos, los roles de flujo de trabajo, las organizaciones, los grupos y los registros de hora de la última conexión en un archivo de Excel o CSV.

Un informe de control de acceso del gestor de tareas de ejemplo:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Name	User Login	Status	Teams	Email	Role	Workflov	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

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmAutomate exportTaskManagerAccessControl` *`REPORT_NAME`* donde *`REPORT_NAME`* es el nombre del archivo de exportación (incluida una extensión (CSV o XLS) válida) que contendrá el informe.

Este informe se puede generar con formato CSV o XSL. Puede descargarlo con el comando downloadFile.

Ejemplos

- `epmAutomate exportTaskManagerAccessControl aclreport.csv`
- `epmAutomate exportTaskManagerAccessControl aclreport.xls`

exportValidIntersections

Exporta los grupos de intersecciones válidos del proceso de negocio en un archivo ZIP, que puede descargar en un equipo local con el comando downloadFile. Las intersecciones válidas son interacciones de celdas que se filtran en función de reglas, denominadas reglas de intersección válidas, que puede definir. Estas reglas filtran determinadas intersecciones de celda para los usuarios cuando introducen datos o seleccionan peticiones de datos en tiempo de ejecución.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate exportValidIntersections` *`FILE_NAME`*`.zip [names=`*`INTERSECTION_NAMES`*`]`, donde:

- *`FILE_NAME`* es un nombre del archivo ZIP de exportación. Todas las intersecciones válidas identificadas en el comando se exportan a un archivo de Microsoft Excel y, a continuación, se comprimen para crear este archivo.

- `names`, opcionalmente, identifica una lista de intersecciones válidas separadas por comas que desea exportar. Si no se especifica este valor de parámetro, el comando exporta todas las intersecciones válidas en la aplicación.

Ejemplos

- **Exportar intersecciones válidas específicas**
  ```
  epmautomate exportValidIntersections VI_export_File.zip
  names=VIAccountPeriod,VIEntityPeriod,VIProductPeriod
  ```
- **Exportar todas las intersecciones válidas**
  ```
  epmautomate exportValidIntersections VI_export_File.zip
  ```

extractDimension

Extrae una dimensión de Oracle Fusion Cloud Enterprise Data Management a un archivo o a una conexión global.

Se aplica a

Oracle Enterprise Data Management Cloud

Roles necesarios

- Administrador del servicio
- Usuario (con permiso de gestor de datos)

Sintaxis

`epmautomate extractDimension` *`APPLICATION DIMENSION EXTRACT_PROFILE FILE_NAME`* `[connection=`*`NAME`*`]`, donde:

- `APPLICATION` es el nombre de una aplicación de Oracle Enterprise Data Management Cloud
- `DIMENSION` es el nombre de la dimensión que se va a extraer.
- `EXTRACT_PROFILE` es el nombre de un perfil de extracción definido en la aplicación. Este perfil se usa para extraer la dimensión.
- `FILE_NAME` es el nombre de un archivo (CSV para exportar a un archivo o ZIP para exportar a Oracle Financials Cloud) para almacenar los datos extraídos. Si el valor del parámetro connection no está establecido, este archivo se crea en el área temporal. Puede descargarlo en un equipo local mediante el comando downloadFile o copiarlo en otro entorno de Oracle Enterprise Data Management Cloud mediante el comando copyFileFromInstance.
- Opcionalmente, `connection=`*`NAME`* identifica un nombre de conexión global (ubicación de instancia) definido en Oracle Enterprise Data Management Cloud como la ubicación del archivo. Si se ha especificado, el archivo de extracción se carga en el entorno de destino (la bandeja de entrada para Oracle Fusion Cloud Enterprise Performance Management y la cuenta del documento especificada para Oracle ERP).

> **Note:**
>
> Las credenciales especificadas en la conexión global deben tener derechos de acceso para escribir en el entorno de destino.

Ejemplos

- Extraiga al área temporal de Oracle Enterprise Data Management Cloud: `epmautomate extractDimension USOperations Entity EntityExtProfile EntityData.CSV`
- Realice la extracción y carga en Oracle ERP: `epmautomate extractDimension USOperations Entity EntityExtProfile EntityData.zip Connection=ora_fusion_gl`
- Realice la extracción y carga en la bandeja de entrada de Cloud EPM de destino: `epmautomate extractDimension USOperations Entity EntityExtProfile EntityData.CSV Connection=EPM_cloud_pln`

extractPackage

Ejecuta un paquete de extracciones formado por varias extracciones para una aplicación con una sola operación. Los resultados de cada extracción en el paquete se agregan a un archivo ZIP, que se puede escribir en el área temporal o en una conexión global.

Se aplica a

Oracle Fusion Cloud Enterprise Data Management

Roles necesarios

- Administrador del servicio
- Usuario (con permiso de gestor de datos)

Sintaxis

`epmautomate extractPackage` *`APPLICATION_NAME EXTRACT_PACKAGE_NAME FILE_NAME`*`.zip [connection=`*`CONNECTION_NAME`*`]`, donde:

- *`APPLICATION_NAME`* es el nombre de una aplicación de Oracle Enterprise Data Management Cloud.
- *`EXTRACT_PACKAGE_NAME`* es el nombre del paquete definido en la aplicación de Oracle Enterprise Data Management Cloud.
- *`FILE_NAME`* es el nombre del archivo ZIP de extracción que contendrá los resultados de la extracción. Este archivo se crea en el área temporal o se escribe en una conexión global si se especifica el nombre de conexión.
  Un paquete puede contener uno o varios perfiles de extracción, cada uno con un nombre diferente. El archivo ZIP del paquete de extracciones contendrá los archivos de salida de las extracciones en el paquete. El número de archivos del archivo ZIP lo determina la configuración del paquete.
- Opcionalmente, *`CONNECTION_NAME`* es el nombre de la conexión global (ubicación de instancia) definido en Oracle Enterprise Data Management Cloud. Identifica la ubicación en la que se va a cargar el archivo ZIP que contiene los resultados de la extracción.
  Si se ha especificado un nombre de conexión, el archivo ZIP que contiene los datos extraídos se carga en el entorno de destino (la bandeja de entrada para un entorno de Oracle Fusion Cloud Enterprise Performance Management o la cuenta del documento especificada para Oracle ERP). Si no se especifica ningún valor, el archivo de exportación se crea en el buzón de salida predeterminado (área temporal) del entorno de Oracle Enterprise Data Management Cloud.

Ejemplos

- Extraer el paquete COARedesignMaps en un archivo en el área temporal de Oracle Enterprise Data Management Cloud:

```
epmautomate extractPackage FinancialsCloud COARedesignMaps COARedesign.zip
```

- Extraer el paquete COARedesignMaps en un archivo y cargarlo en una bandeja de entrada de Oracle Fusion Cloud EPM con una conexión.

```
epmautomate extractPackage CorporateAccount COARedesignMaps
COARedesign.zip Connection=EPM_cloud_pln
```

feedback

Envía comentarios a Oracle y a los administradores del servicio del entorno y carga de forma automática todos los archivos de registro de EPM Automate creados en las últimas 24 horas desde el directorio actual.

Opcionalmente, puede cargar los archivos adicionales (por ejemplo, archivos de rastreo de Fiddler) que desee que utilicen los Servicios de Soporte Oracle para diagnosticar por qué se produce la incidencia actual.

Este comando, que imita la función Proporcionar comentarios del servicio, resulta de especial utilidad para hacer comentarios a Oracle en los casos en que la interfaz de usuario no responde o se produce un problema al ejecutar EPM Automate.

Para obtener más información sobre la función Proporcionar comentarios, consulte Contribución a la recopilación de información de diagnóstico de Oracle mediante la utilidad de proporción de comentarios en *Guía de introducción para administradores*.

Este comando devuelve un mensaje parecido al siguiente en el que se informa de que el comentario no crea una solicitud de servicio. Si necesita ayuda de Oracle para resolver un problema, debe registrar una solicitud de servicio. El comando muestra un número de referencia UDR que debe incluir en la solicitud de servicio que registre.

```
Thank you for providing feedback. If you need Oracle's assistance with this issue please log into My Oracle Support and
log a service request.
Make a note of the feedback reference below as you will be asked to provide this information during the SR submission process.
Reference is UDR_502367689_example@example.com_2022_10_17_06_29_41
feedback completed successfully

c:\Oracle\EPM Automate\bin>
```

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario

- Visor

Sintaxis

`epmautomate feedback "Comment" [Screenshot="FILE_PATH"] [File="FILE_PATH"]`, donde:

- `Comment` es un texto que describe la incidencia sobre la que se envía este comentario. Los comentarios deben escribirse entrecomillados.
- `Screenshot`, de forma opcional, identifica el nombre de un archivo de gráfico que ilustra el problema para el que se está enviando este comentario. Puede enviar varias instantáneas mediante la repetición de este parámetro y este valor según sea necesario.
- Opcionalmente, `File` identifica el nombre de un archivo que desee que utilicen los Servicios de Soporte Oracle para resolver la incidencia actual. Utilice este parámetro para enviar los rastreos de Fiddler u otros archivos a Oracle. Puede enviar varios archivos mediante la repetición de este parámetro y este valor según sea necesario.

Ejemplos

- 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

Comprueba si la aplicación está en modo de administración con accesos limitado solo a los administradores del servicio.

Este comando, que devuelve `true` si la aplicación está en modo de administración y `false` en caso contrario, resulta útil para comprobar el estado de la aplicación antes de ejecutar los scripts de automatización. Por ejemplo, el comando refreshCube exige que la aplicación esté en modo de administración. Puede usar este comando en el script de automatización de la siguiente forma si la aplicación está en modo de administración.

```
adminMode = 'epmautomate.sh getApplicationAdminMode'
        if ["$adminMode" == "true"]
              epmautomate.sh refreshCube
```

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate getApplicationAdminMode`

Ejemplo

```
epmautomate getApplicationAdminMode
```

getDailyMaintenanceStartTime

Muestra, en la consola, la hora universal coordinada (UTC) u, opcionalmente, la zona horaria, a la que está programado el inicio del mantenimiento diario del entorno.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio, usuario (solo Oracle Enterprise Data Management Cloud), cualquier rol predefinido y el rol de aplicación Migraciones - Administrar

Sintaxis

`epmautomate getDailyMaintenanceStartTime [timezone=true|false]`, donde `timezone=true`, opcionalmente, identifica si se va a mostrar la hora de inicio del mantenimiento diario en la zona horaria especificada al establecerse, por ejemplo, `America/Los_Angeles`. El valor predeterminado es `false`.

Ejemplos

- Muestre la hora de mantenimiento en la zona horaria especificada mientras se establece:
  `epmautomate getDailyMaintenanceStartTime timezone=true`
- Muestre la hora de mantenimiento en UTC:
  `epmautomate getDailyMaintenanceStartTime`

getEssbaseQryGovExecTime

Muestra la cantidad máxima de tiempo actual, en segundos, que puede utilizar una consulta de Oracle Essbase para recuperar y entregar información antes de la terminación de la consulta.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

```
epmautomate getEssbaseQryGovExecTime
```

Salida del comando de ejemplo:

```
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
```

Ejemplo

```
epmautomate getEssbaseQryGovExecTime
```

getIdleSessionTimeout

Muestra el tiempo de espera de sesión (en minutos) del entorno de Oracle Fusion Cloud Enterprise Performance Management. Después de que una sesión esté inactiva durante este tiempo, se redirecciona a los usuarios a la página de inicio de sesión.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

```
epmautomate getIdleSessionTimeout
```

Salida del comando de ejemplo:

```
c:\Oracle\EPM Automate\bin>epmautomate getIdleSessionTimeout
75
```

getIPAllowlist

Para entornos de OCI (Gen 2), muestra las direcciones IP y Classless Inter-Domain Routing (CIDR) incluidas en la lista de permitidos actual.

Este comando es útil para comprobar si una dirección IP o CIDR específica puede acceder actualmente a un entorno de OCI (Gen 2).

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

```
epmAutomate getIPAllowlist
```

> **Note:**
>
> Para escribir todas las direcciones IP y CIDR existentes en un archivo, redireccione la salida a un archivo de texto, que puede editar (eliminar algunas o todas las direcciones IP y CIDR), cargar en el entorno y, a continuación, usar el comando setIPAllowlist para eliminar las entradas en el archivo de la lista de permitidos. Ejemplo de ejecución de comando:
>
> ```
> epmAutomate getIPAllowlist > myRemoveList.txt
> epmAutomate uploadFile myRemoveList.txt
> epmAutomate setIPAllowlist remove myRemoveList.txt
> ```

Ejemplo

Mostrar las direcciones IP y CIDR incluidas en la lista de permitidos actual:

```
epmAutomate getIPAllowlist
```

getRestrictedDataAccess

Muestra si el entorno está configurado para impedir que los administradores del servicio consientan el envío de una instantánea de aplicación a Oracle al usar la utilidad Proporcionar comentarios.

Este comando muestra `true` si el entorno está configurado para impedir que los administradores del servicio consientan el envío de la instantánea de la aplicación y, de lo contrario, muestra `false`.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

Para cambiar el estado de la restricción de acceso a los datos actual, utilice setRestrictedDataAccess.

Ejemplo

```
epmautomate getRestrictedDataAccess
```

getSubstVar

Recupera los valores de las variables de sustitución y los muestra en pantalla.

El formato de visualización es *CUBE_NAME.SUBSTVAR*=value; por ejemplo, Plan2.CurYear=2016. Los valores de las variables de sustitución de nivel de aplicación se muestran con el formato *ALL.VARIABLE_SUSTITUCIÓN*=valor, por ejemplo, ALL.CurYear=2016

Se aplica a

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

Roles necesarios

Administrador del servicio, Usuario avanzado (con acceso de inicio de la regla)

Sintaxis

epmautomate getSubstVar *CUBE_NAME*|ALL [*name=VARIABLE_NAME*] donde:

- *CUBE_NAME* es el cubo (por ejemplo, Plan1, Plan2) del que desea recuperar la variable de sustitución. Utilice ALL para recuperar variables de sustitución en el nivel de aplicación.
- *name=NOMBRE_VARIABLE* identifica opcionalmente la variable de sustitución para la que desea recuperar el valor. Si no especifica un nombre de variable, el comando recupera el valor de todas las variables de sustitución.

Ejemplos

- Obtener el valor de todas las variables de sustitución en la aplicación y el nivel de cubo: epmautomate getSubstVar ALL
- Obtener el valor de una variable de sustitución específica en el nivel de aplicación: epmautomate getSubstVar ALL name=CurYear
- Obtener el valor de todas las variables de sustitución en el nivel de cubo: epmautomate getSubstVar Plan2
- Obtener el valor de una variable de sustitución específica en el nivel de cubo: epmautomate getSubstVar Plan2 name=CurYear

getVirusScanOnFileUploads

Comprueba si el entorno de OCI (Gen 2) está habilitado para explorar todos los archivos que se cargan para asegurarse de que no tienen virus.

Este comando comprueba si se ha aplicado la exploración de virus antes de que se carguen los archivos en el entorno.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate getVirusScanOnFileUploads`
Este comando imprime `true` si el entorno está activado para explorar los archivos cargados en busca de virus; de lo contrario, imprime `false`.

groupAssignmentAuditReport

Crea un informe en el que se muestran los usuarios y los grupos que se han asignado o cuya asignación se haya anulado (eliminado) de los grupos de Control de acceso para un rango de fechas especificado. Este informe se puede generar durante un máximo de 120 días.

Este informe, que se genera como archivo CSV, se puede usar para soportar operaciones de auditoría de seguridad. Cada fila del archivo CSV generado proporciona el usuario o grupo agregado o eliminado, el grupo al que se ha agregado el usuario o grupo o del que se ha eliminado, el administrador del servicio que ha realizado la acción, así como la fecha y hora en que se ha llevado a cabo la acción.

	A	B	C	D	E
1	**User/Group Name**	**Group**	**Action**	**Administrator**	**Date and Time**
2	testGroup1	exampleGroup1	Added	john.smith@example.com	May 10, 2022 23:13:20 UTC
3	johndoe@example.com	exampleGroup1	Added	joe.doe@example.com	May 11, 2022 23:14:20 UTC
4	JaneDoe@example.com	testGroup1	Added	joe.doe@example.com	May 11, 2022 23:14:50 UTC
5	jaDoe@example.com	testGroup1	Added	john.smith@example.com	May 12, 2022 23:25:20 UTC
6	JaneDoe@example.com	testGroup1	Removed	john.smith@example.com	May 13, 2022 18:13:20 UTC
7	jaDoe@example.com	testGroup1	Removed	john.smith@example.com	May 13, 2022 18:22:20 UTC
8	testGroup1	exampleGroup1	Removed	joe.doe@example.com	May 13, 2022 23:13:20 UTC

Se aplica a

Planning, Planning Modules, 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 y Sales Planning.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Gestionar
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Ver

Sintaxis

`epmAutomate groupAssignmentAuditReport` *`FROM_DATE TO_DATE REPORT_NAME`*, donde

- *`FROM_DATE`* es la fecha de inicio del periodo, con formato `YYYY-MM-DD`, para la que se va a generar el informe.
- *`TO_DATE`* es la fecha de finalización del periodo, con formato `YYYY-MM-DD`, para la que se va a generar el informe.

- *REPORT_NAME* es el nombre de un archivo CSV para el informe. Puede descargar el informe generado mediante el comando downloadFile.

Ejemplo

```
epmAutomate groupAssignmentAuditReport 2022-03-01 2022-05-01
GroupAssignmentReport.CSV
```

help

Muestra la ayuda para todos los comandos de EPM Automate o para un comando específico.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Sintaxis

`epmautomate [COMMAND_NAME] help`, donde *COMMAND_NAME*, opcionalmente, es el nombre del comando cuya ayuda desea mostrar en la consola.

Ejemplos

- Mostrar una página web desde la que puede acceder a ayuda sobre todos los comandos:
  `epmautomate help`
- Mostrar ayuda de los comandos en el comando encrypt en la consola:
  `epmautomate encrypt help`

importAppAudit

Importa registros de auditoría de datos de un archivo ZIP que haya creado exportando los datos de auditoría de un entorno.

Cree el archivo de importación con el comando exportAppAudit (`epmautomate exportAppAudit auditData ndays=All`). Use este comando para clonar registros de auditoría de un entorno a otro durante la migración o la clonación para la recuperación ante desastres.

Se aplica a

Planning, Planning Modules, FreeForm, Enterprise Profitability and Cost Management, Strategic Workforce Planning y Sales Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate importAppAudit FILE_NAME [logFilename=LOG_FILE_NAME]`, donde:

- `FILE_NAME` es el nombre de un archivo ZIP que contiene registros de auditoría de datos que desea importar en la aplicación. Antes de ejecutar este comando, utilice el comando uploadFile para cargar este archivo en el entorno.
- `logFileName`, opcionalmente, identifica el archivo log de errores en el que se registrarán los errores encontrados durante la importación. Si no se especifica este valor, el comando genera un archivo de error cuyo nombre se asigna según esta convención: `username_date_timestamp`. Puede descargar este archivo utilizando el comando downloadFile.

Ejemplo

```
epmautomate importAppaudit Audit_data.zip logFileName=auditImportLog
```

importAppSecurity

Carga permisos de acceso para usuarios o grupos de una aplicación de un archivo CSV disponible en la bandeja de entrada.

La importación de permisos de acceso sobrescribe las asignaciones existentes solo para miembros importados, formularios de datos, carpetas de formularios de datos, listas de tareas, reglas de negocio de Calculation Manager y carpetas de reglas de negocio. Todos los demás permisos de acceso existentes se mantienen intactos.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate importAppSecurity ACL_FILE_NAME ERROR_FILE [clearall=true|false]` donde:

- `NOMBRE_ARCHIVO_ACL` es el nombre de un archivo CSV que contiene permisos de acceso que desea importar en la aplicación. Antes de ejecutar este comando, utilice el comando uploadFile para cargar este archivo en el buzón de entrada. El contenido de un archivo de entrada de ejemplo puede ser similar al que se muestra en la siguiente imagen:

	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

Para obtener una descripción de las cabeceras de columna y los posibles valores, consulte Importación de seguridad en la guía *API de REST*.

- `ERROR_FILE` es el nombre de un archivo CSV, que EPM Automate crea en el buzón de salida para registrar los errores que se detecten durante esta operación. Puede descargar este archivo en una computadora local para analizar y corregir los informes indicados. El contenido de un archivo de error de ejemplo puede ser similar al que se muestra en la siguiente imagen. Las columnas de este archivo se corresponden con las columnas de cabecera del archivo de entrada:

	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`, opcionalmente, especifica si se van a suprimir los permisos de acceso existentes antes de cargar los permisos nuevos del archivo. El valor predeterminado es `false`.

Ejemplo

```
epmautomate importAppSecurity Acl_file.CSV Acl_import_error.CSV clearall=true
```

importARApplicationProperties

Importa la configuración de la aplicación (experiencia Redwood, tema, notificación por correo electrónico y nombre de proceso de negocio), el logotipo y las imágenes de fondo de un archivo JSON de exportación en un entorno de Account Reconciliation.

Se aplica a

Account Reconciliation

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate importARApplicationProperties` *`FILE_NAME`* donde *`FILE_NAME`* es el nombre del archivo JSON exportado desde un entorno.

Este archivo, exportado desde otro entorno con el comando exportARApplicationProperties, debe estar disponible en el entorno en el que esté restaurando la configuración de la aplicación.

Ejemplo

```
epmautomate importARApplicationProperties myProp.JSON
```

importBackgroundImage

Importa la imagen de fondo de un archivo de exportación a un entorno de Account Reconciliation.

Se aplica a

Account Reconciliation

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate importBackgroundImage FILE_NAME.jpg`, donde *FILE_NAME* es el nombre del archivo de imagen de fondo exportado desde otro entorno.

Ejemplo

```
epmautomate importBackgroundImage image_file.jpg
```

importBalances

Utiliza Data Management para importar datos de balances de una definición de carga de datos.

Se aplica a

Account Reconciliation.

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

A los usuarios con los roles Usuario avanzado, Usuario y Visor se les debe otorgar seguridad adicional mediante una ACL

Sintaxis

`epmautomate importBalances DL_DEFINITION PERIOD`, donde:

- *DL_DEFINITION* es una definición de carga de datos existente en Account Reconciliation.
- *PERIODO* es el nombre de un periodo.

Ejemplo

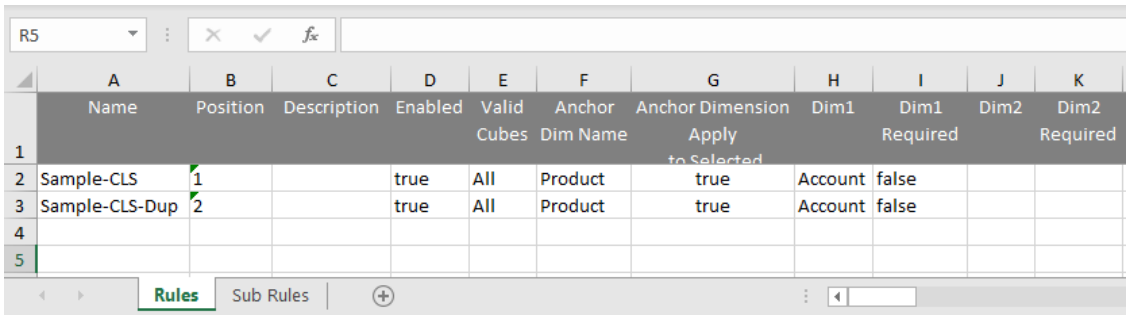
```
epmautomate importBalances DailyLoad "January 2020"
```

importCellLevelSecurity

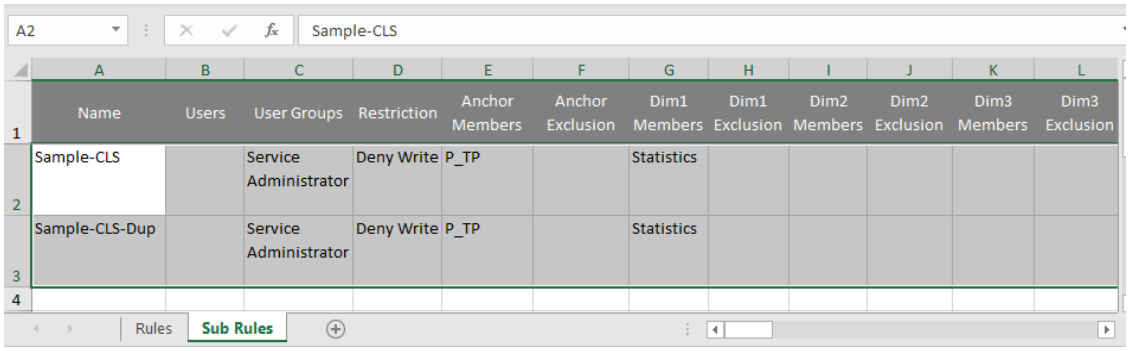
Importa la configuración de seguridad a nivel de celda de un archivo ZIP que contiene un archivo Excel con registros de seguridad de nivel de celda en el proceso de negocio. Antes de ejecutar este comando, utilice el comando uploadFile para cargar el archivo de importación en el entorno.

El archivo ZIP de importación debería contener un archivo de Excel con dos hojas de trabajo (Rules y Sub Rules) para importar correctamente la seguridad a nivel de celda. La hoja Rules debe contener definiciones de seguridad a nivel de celda, con dimensiones, y propiedades como Dimensiones válidas no especificadas y Dimensiones adicionales necesarias. La hoja Sub Rules debería contener las selecciones y exclusiones de miembros. El mejor método para

obtener la plantilla de formato del archivo de importación es exportar la seguridad a nivel de celda de la aplicación. En las siguientes ilustraciones se muestra un formato de ejemplo.



	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											



	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												

Se aplica a

Planning, Planning Modules, FreeForm, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate importCellLevelSecurity` *`FILE_NAME`*`.ZIP [ErrorFile=`*`FILE_NAME`*`.txt]`, donde:

- `FILE_NAME` es el nombre del archivo ZIP que contiene el archivo de Excel con información de seguridad de nivel de celda.
- `ErrorFile`, opcionalmente, identifica el nombre del archivo de texto en el que se escribirán los registros de error. Si no se especifica este valor de parámetro, el comando genera automáticamente un archivo de error; puede ver su nombre en la consola de trabajos. Utilice el comando downloadFile para descargar el archivo de error en un equipo local.

Ejemplo

```
epmautomate importCellLevelSecurity ImportCLSDRecordsFile.zip
ErrorFile=ImportCLSDRecords_errors.txt
```

importConsolidationJournals

Importa asientos de consolidación de un archivo .JLF en Financial Consolidation and Close.

- Utilice el comando exportConsolidationJournals para crear el archivo .JLF que se utiliza como entrada para este comando.
- Antes de ejecutar este comando, utilice el comando uploadFile para cargar el archivo de entrada en el entorno.

Se aplica a

Financial Consolidation and Close

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate importConsolidationJournals jobName [fileName=FILE_NAME] [errorFileName=ERROR_FILE_NAME]` donde

- `jobName` es el nombre de un trabajo de importación de asientos creado en Financial Consolidation and Close.
- Opcionalmente, `fileName`, es el nombre de un archivo .JLF desde el que se importarán las entradas de asiento.
- Opcionalmente, `errorFileName`, es el nombre del archivo de registro en el que se registrarán los mensajes generados durante el proceso de importación.

Ejemplo

```
epmautomate importConsolidationJournals "JIMPORT1" fileName="TestImport1.jlf"
errorFileName="TestImport1_error.log"
```

importData

Importa datos de un archivo a la aplicación utilizando la configuración de importación de datos especificada en un trabajo de tipo `import data`.

Utilice el comando uploadFile para cargar el archivo que contiene los datos de la aplicación en la ubicación de carga predeterminada.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate importData JOB_NAME [FILE_NAME] errorFile=ERROR_FILE.zip` donde:

- *JOB_NAME* es el nombre de un trabajo definido en la aplicación
- *FILE_NAME*, opcionalmente, especifica el nombre del archivo ZIP, CSV o TXT (archivo de datos de formato Essbase) desde el que se van a importar los datos. Si especifica un nombre de archivo, se ignorará el nombre del archivo de importación en el trabajo.
  Si el trabajo se define para importar los datos en formato de Essbase, el archivo ZIP debe incluir un archivo de formato TXT de Essbase. Para otros trabajos de importación, el

archivo ZIP puede contener uno o más archivos CSV que identifiquen la secuencia de importación en los nombres de archivo, por ejemplo, `data1-3.csv`, `data2-3.csv` y `data3-3.csv`.

- `errorFile`, opcionalmente, identifica el nombre de un archivo ZIP en el que se registrarán los registros rechazados, si los hubiera, durante las operaciones de importación. Se sobrescribirá el archivo ZIP con el mismo nombre en la bandeja de salida, si lo hay. Puede descargar este archivo utilizando el comando downloadFile.

Ejemplo

```
epmautomate importData dailydataload dailydata.zip errorFile=dataImport_error.zip
```

importDataManagement

Importa registros de Data Management de un archivo ZIP en un entorno.

Con este comando se importan los datos en las tablas de configuración y temporales de un archivo ZIP creado con el comando exportDataManagement. Un administrador del servicio utiliza el comando uploadFile, por ejemplo, `epmAutomate uploadFile "C:/datafile/datafile.zip" inbox` para cargar el archivo ZIP de la importación en la bandeja de entrada de Data Management o en una de sus carpetas.

> **Note:**
>
> Con este comando solo se pueden importar registros de Data Management exportados de otro entorno que ejecuten la misma actualización mensual. Por ejemplo, los registros exportados de un entorno 21.11 de Oracle Fusion Cloud Enterprise Performance Management solo se pueden importar en otro entorno 21.11.

Se aplica a

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

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

`epmautomate importDataManagement` *`FILE_NAME`*`.zip`, donde *`FILE_NAME`* es el nombre del archivo ZIP que contiene los datos de Data Management que se van a importar.

Ejemplos

- Importe del buzón de Data Management:
  ```
  epmautomate importDataManagement inbox/dataFile.zip
  ```
- Importe de una carpeta del buzón:
  ```
  epmautomate importDataManagement inbox/dm_data/dataFile.zip
  ```

importDimension

Importa una dimensión desde un archivo en una aplicación de Oracle Fusion Cloud Enterprise Data Management.

Este comando puede importar un archivo de entrada desde una conexión definida en Oracle Enterprise Data Management Cloud o el área temporal.

Si se va a importar el archivo desde el área temporal de Oracle Enterprise Data Management Cloud, un administrador del servicio utiliza el comando uploadFile para cargarlo en el entorno de Oracle Enterprise Data Management Cloud de destino. Un administrador del servicio también puede utilizar el comando copyFileFromInstance para copiar el archivo de otro entorno de Oracle Fusion Cloud Enterprise Performance Management.

Se aplica a

Oracle Enterprise Data Management Cloud.

Roles necesarios

- Administrador del servicio
- Usuario (con permiso de gestor de datos)

Sintaxis

`epmautomate importDimension` *`APPLICATION DIMENSION IMPORT_TYPE FILE_NAME`* `[connection=`*`NAME`*`]` donde:

- `APPLICATION` es el nombre de una aplicación de Oracle Enterprise Data Management Cloud
- `DIMENSION` es el nombre de la dimensión de aplicación que se está importando.
- `IMPORT_TYPE` indica cómo realizar la importación. Los tipos de importación válidos son los siguientes:
  - `ResetDimension` para suprimir todos los datos de dimensión existentes e importar los nuevos datos
  - `ReplaceNodes` para agregar o actualizar nodos y reemplazar las jerarquías existente durante la importación
  - `Merge` para procesar cambios incrementales en los nodos y las jerarquías mediante una solicitud de importación
- `FILE_NAME` es el nombre del archivo (CSV o ZIP) que contiene los datos de dimensión que se van a importar. El nombre de archivo debe finalizar con el nombre de dimensión precedido por _ (guion bajo); por ejemplo, `import_Entity.csv`. Si va a importar desde un archivo ZIP que contiene varios archivos de importación, este comando depende del nombre de archivo dentro del archivo ZIP para identificar el archivo de importación correcto.
  Si se ha especificado un valor para `connection`, debe importar la dimensión desde un archivo ZIP; por ejemplo, `importdata_Entity.zip`.
- Opcionalmente, `connection=`*`NAME`* identifica un nombre de conexión (ubicación de instancia) definido en Oracle Enterprise Data Management Cloud como la ubicación del archivo de importación. Si no se ha especificado, el proceso de importación buscará el archivo de importación en el área temporal local.

Ejemplos

- Importe desde un archivo cargado en el área temporal: `epmautomate importDimension USOperations Entity ReplaceNodes data_Entity.CSV`
- Importe desde el buzón de salida de otro entorno de Cloud EPM: `epmautomate importDimension USOperations Entity ReplaceNodes data_Entity.ZIP Connection=Cloud-EPM_pln`

importJobConsole

Clona registros de la consola de trabajos con un archivo ZIP que contenga registros de la consola de trabajos exportados de un entorno.

Importar los registros de la consola de trabajos con este comando es una tarea que solo se realiza una vez y que se debe llevar a cabo tras ejecutar el comando recreate. Si ya ha usado este comando para importar registros de la consola de trabajos, las posteriores llamadas del comando fallará hasta que vuelva a crear el entorno.

Use el comando exportJobConsole (`epmAutomate exportJobConsole FILE_NAME.zip nDays=All jobTypes=All jobStatusCode=All`) para crear el archivo ZIP que se usa como entrada para este comando.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate importJobConsole FILE_NAME.zip [logFileName=jobConsoleLog]`, donde:

- *`FILE_NAME`* es el nombre del archivo ZIP que contiene los registros de la consola de trabajos que desea importar. Utilice el comando uploadFile para cargar este archivo en el entorno
- `logFileName`, opcionalmente, identifica `jobConsoleLog` como el archivo de registro en el que se registrarán los errores encontrados durante la importación. Si no se especifica este valor, el comando genera un archivo de error cuyo nombre se asigna con esta convención: *`username`*`importLog_date_timestamp.zip`. Puede descargar este archivo utilizando el comando downloadFile.

Ejemplo

```
epmautomate importJobConsole jobConsole.zip jobConsoleLog
```

importLibraryArtifact

Importa artefactos de biblioteca desde un archivo en la biblioteca de Narrative Reporting.

Antes de ejecutar este comando, un administrador del servicio utiliza el comando de carga uploadFile para cargar el archivo de origen o el archivo en el entorno.

Se aplica a

Narrative Reporting

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

A los usuarios con los roles Usuario avanzado, Usuario y Visor se les debe otorgar seguridad adicional mediante una ACL

Sintaxis

`epmautomate importLibraryArtifact` *`SOURCE_FILE`* `[errorFile=`*`ERROR_FILE`*`.txt] [importFormat=Native|File] [importFolder=`*`FOLDER_PATH`*`] [ importPermission=true|false] [overwrite=true|false]`, donde:

- *`SOURCE_FILE`* es el nombre del archivo que contiene los artefactos que se van a importar en la biblioteca. Este archivo debe estar disponible en la bandeja de entrada.
- Opcionalmente, *`errorFile`* es un nombre único para el archivo de texto en el que se almacenarán los errores relacionados con la importación.
- `importFormat`, opcionalmente, es uno de estos valores:
  - Con `Native` se importan artefactos de un archivo zip creado con el comando exportLibraryArtifact con la opción `exportFormat=Native`. Este es el valor predeterminado.
  - Con `File` se importa un archivo binario.

> **Nota:**
>
> Use el comando importSnapshot para importar archivos zip de artefactos de biblioteca (creados con el comando exportLibraryArtifact con la opción `exportFormat=LCM`) en entornos de Financial Consolidation and Close, Planning, Planning Modules o Tax Reporting.

- Opcionalmente, `importFolder` es la ubicación de la biblioteca en la que se almacenarán los artefactos importados. Especifique esta ruta si esta ubicación es diferente de `Library` (la ubicación de importación predeterminada).
- *`importPermission`* indica si se van a importar los permisos de acceso definidos para los artefactos. El valor predeterminado es `False`.
- *`overwrite`* identifica si se van a sobrescribir los artefactos con nombres idénticos, si los hay, en la ubicación de biblioteca especificada. El valor predeterminado es `False`, que significa que el proceso no importará un artefacto si existe un artefacto con nombre idéntico en la ubicación de importación.

Al finalizar la importación, utilice el comando downloadFile para descargar los archivos de errores en un equipo local.

Ejemplos

- Importe un archivo en su formato binario:
  ```
  epmautomate importLibraryArtifact newReports.doc
  errorFile=report_imp_errors.txt importFormat=File importFolder="Library/My
  Reports" importPermission=true overwrite=true
  ```
- Importe los artefactos de un archivo zip exportados:
  ```
  epmautomate importLibraryArtifact newReports.zip
  errorFile=report_imp_errors.txt importFormat=Native importFolder="Library/My
  Reports" importPermission=true overwrite=true
  ```
- Importe los informes en un entorno de Financial Consolidation and Close, Planning, Planning Modules o Tax Reporting de un archivo zip exportado:
  ```
  epmautomate importSnapshot newReports.zip
  ```

importLogoImage

Importa el logotipo corporativo usado en un entorno de Account Reconciliation de un archivo de exportación en otro entorno.

Se aplica a

Account Reconciliation

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate importLogoImage` *`IMAGE_NAME`*`.jpg`, donde *`IMAGE_NAME`* es el nombre del archivo de imagen de logotipo.
Puede descargar la imagen exportada con el comando downloadFile. Cárguelo en el entorno de destino con el comando uploadFile y, a continuación, ejecute el comando importLogoImage para importarlo.

Ejemplo

```
epmautomate importLogoImage corpLogo.jpg
```

importMapping

Importa asignaciones desde un archivo de importación de asignaciones que se ha cargado previamente en el entorno.

Un administrador del servicio utiliza el comando uploadFile para cargar archivos en la bandeja de entrada de Data Management o en una de sus carpetas.

Se aplica a

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

Roles necesarios

- Administrador del servicio

- Usuario avanzado

Sintaxis

`epmautomate importMapping` *`DIMENSION_NAME|ALL FILE_NAME IMPORT_MODE VALIDATION_MODE LOCATION`* donde:

- *`DIMENSION_NAME|ALL`* indica el destinatario de la asignación. Especifique el nombre de la dimensión a la que se van a importar asignaciones u `ALL` para importar todas las asignaciones incluidas en el archivo a dimensiones adecuadas.
- *`FILE_NAME`* es el nombre y la ubicación del archivo de importación de asignaciones disponible en la bandeja de entrada de Data Management o en un directorio dentro de esta. Especifique el nombre de archivo (archivos TXT en formato de Data Management estándar) y su ruta (por ejemplo, `inbox/AccountMap.txt` o `inbox/pbcs_maps/AccountMap.txt`).
- *`IMPORT_MODE`* es `REPLACE` para borrar las reglas de asignación existentes antes de importar asignaciones o `MERGE` para agregar nuevas reglas de asignación a las reglas existentes.
- *`VALIDATION_MODE`* es `TRUE` para validar los miembros de destino con respecto a la aplicación o `FALSE` para cargar el archivo de asignación sin ejecutar las validaciones.
- *`LOCATION`* es la ubicación de Data Management para la que se deben cargar las reglas de asignación.

Ejemplos

- `epmautomate importMapping Account inbox/AccountMap.txt MERGE FALSE "France Sales"`
- `epmautomate importMapping ALL "inbox/France Sales/AllMaps.txt" MERGE FALSE "France Sales"` (carga las asignaciones del archivo de importación de asignaciones en todas las dimensiones asignadas en la ubicación France Sales)

importMetadata

Importa metadatos a la aplicación utilizando la configuración de importación especificada en un trabajo de tipo `import metadata`. También puede especificar el nombre del archivo ZIP desde el que se importarán los metadatos.

Utilice el comando uploadFile para cargar el archivo que contiene los metadatos en la ubicación de carga predeterminada.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate importMetadata` *`JOB_NAME [FILE_NAME]`* `errorFile=`*`ERROR_FILE`*`.zip` donde:

- *`JOB_NAME`* es el nombre de un trabajo definido en la aplicación
- *`FILE_NAME`*, opcionalmente, identifica el nombre del archivo ZIP desde el cual se importarán los metadatos. Si se especifica, el contenido de este archivo ZIP tiene prioridad

sobre los nombres de archivo definidos en el trabajo. El archivo ZIP puede contener uno o más archivos CSV. Los nombres de archivo que contienen metadatos para dimensiones deben coincidir con los nombres de archivos de importación definidos en el trabajo o bien terminar por `_DIMENSIONNAME.csv`. Por ejemplo: `metadata_Entity.csv`, `metadata_HSP_Smart Lists.csv` y `metadata_Exchange Rates.csv`.

- `errorFile`, opcionalmente, identifica el nombre de un archivo ZIP en el que se registrarán los registros rechazados, si los hubiera, durante las operaciones de importación. Se sobrescribirá el archivo ZIP con el mismo nombre en la bandeja de salida, si lo hay. Puede descargar este archivo utilizando el comando downloadFile.

Nota:

- No puede cambiar el nombre de los miembros mediante la ejecución de un trabajo de importación de metadatos con un archivo de carga en el que se hayan modificado las propiedades `old_name` o `unique_name`. Se ignorará el cambio de nombre de los miembros.
- No puede suprimir dimensiones de atributos mientras importa metadatos con este comando.
- Solo se importarán los metadatos para las dimensiones para los que está configurada la importación de metadatos en el trabajo. Si están incluidas en el archivo ZIP, se ignorarán los metadatos para otras dimensiones.

  Se crea una situación de importación ambigua si se cumplen las dos condiciones siguientes para el archivo ZIP:

  – El ZIP contiene un archivo de metadatos cuyo nombre coincide con el nombre de archivo definido en el trabajo.

  – El ZIP contiene uno o más archivos de metadatos cuyos nombres terminan por `_DIMENSIONNAME.CSV` o `_DIMENSIONNAME.TXT`, donde *DIMENSIONNAME* es el nombre de la dimensión en la que se importan los metadatos.

Oracle recomienda que el archivo ZIP contenga un archivo de metadatos que tenga un nombre idéntico al que se hace referencia en el trabajo, o un archivo cuyo nombre termine en `_NOMBREDIMENSIÓN.CSV` (o `_NOMBREDIMENSIÓN.TXT`), pero no las dos cosas a la vez. Por ejemplo, si va a cargar un trabajo que hace referencia al archivo de metadatos `Empleados_A-Z.CSV` en la dimensión `Empleados`, el archivo ZIP puede incluir `Empleados_A-Z.CSV` o `Nuevos_Empleados.CSV`, pero no los dos. Si el ZIP contiene `Empleados_A-Z.CSV` y `Nuevos_Empleados.CSV`, EPM Automate puede seleccionar cualquiera de los dos archivos para la importación, dependiendo del orden de los archivos en el ZIP. El archivo `Empleados_A-Z.CSV` es apto para la importación porque su nombre coincide con el nombre de archivo al que se hace referencia en el trabajo; `Nuevos_Empleados.CSV` también es apto porque su nombre coincide con el patrón `_NOMBREDIMENSIÓN.CSV`.

Ejemplo

```
epmautomate importMetadata importAccount importAccount.zip
errorFile=metadataImport_error.zip
```

importOwnershipData

Importa datos de propiedad de un archivo CSV disponible en el entorno a un período.

Antes de ejecutar este comando, un administrador del servicio utiliza el comando uploadFile para cargar el archivo CSV de origen de importación en el entorno. La cabecera de este archivo CSV es similar a esta:

```
Scenario, Year, Period, Entity, Parent, POwn, Control, Method
```

Los valores `POwn`, `Control` y `Method` son opcionales.

Los datos de propiedad importados se fusionan con los datos existentes, lo que puede generar entradas de propiedad no válidas. Si una entidad está presente en más de una rama o jerarquía, los datos de propiedad importados pueden provocar que el % de propiedad combinado de la entidad supere el 100%. Debe corregir manualmente el % de propiedad para garantizar que no supere el 100%.

Se aplica a

Financial Consolidation and Close y Tax Reporting.

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario con acceso de escritura para la entidad.

Sintaxis

`epmautomate importOwnershipData` *`Scenario Year Period FILE_NAME`* donde:

- `Scenario` es el escenario al que se van a importar los datos de propiedad.
- `Year` es el año al que se van a importar los datos.
- `Period` es el periodo del año al que se van a importar los datos de propiedad.
- `NOMBRE_ARCHIVO` es el nombre del archivo CSV del que se deben importar los datos.

Ejemplo

```
epmautomate importOwnershipData FCCS_TotalActual FY19 Jan importfile.csv
```

importPreMappedBalances

Importa datos de balance preasignados de un archivo en el repositorio de Account Reconciliation.

Se aplica a

Account Reconciliation

Roles necesarios

- Administrador del servicio
- Usuario avanzado

- Usuario
- Visor

Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

`epmautomate importPreMappedBalances` *`PERIOD FILE_NAME BALANCE_TYPE CURRENCY_BUCKET`* donde:

- *`PERIODO`* es el nombre de un periodo
- *`FILE_NAME`* es el nombre del archivo CSV que contiene los datos que se van a importar
- *`TIPO_BALANCE`* es `SRC` o `SUB`
- *`DEPÓSITO_MONEDAS`* es `Entered`, `Functional` o `Reporting`

Ejemplo

```
epmautomate importPreMappedBalances "January 2015" dailydata.csv SRC Reporting
```

importPreMappedTransactions

Importa transacciones preasignadas de un archivo CSV en el repositorio de Account Reconciliation.

Se aplica a

Account Reconciliation

Roles necesarios

Administrador del servicio, Usuario avanzado, Usuario, Visor
Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

`epmautomate importPreMappedTransactions` *`PERIOD TRANSACTION_TYPE FILE_NAME DATE_FORMAT`* donde:

- *`PERIODO`* es el nombre de un periodo
- *`TRANSACTION_TYPE`* es una de las siguientes opciones:
  - `BEX` para cargar las explicaciones de balance
  - `SRC` para cargar los ajustes del sistema de origen
  - `SUB` para cargar los ajustes del subsistema
  - `VEX` para cargar las explicaciones del análisis de varianza
- *`FILE_NAME`* es el nombre del archivo CSV del que se van a importar los datos
- *`DATE_FORMAT`* es una cadena de texto con formato de fecha; por ejemplo, MMM d, yyyy.

Ejemplo

```
epmautomate importPreMappedTransactions "January 2015" "BEX" transactions.csv
"MMM d, yyyy"
```

importProfiles

Importa nuevas definiciones de perfil de un archivo CSV en el repositorio de Account Reconciliation.

Se aplica a

Account Reconciliation

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

`epmautomate importProfiles` *`FILE_NAME PROFILE_TYPE METHOD DATE_FORMAT`* donde:

- *`FILE_NAME`* es el nombre del archivo CSV del que se van a importar los datos
- *`TIPO_PERFIL`* es `profiles` o `children`
- *`MÉTODO`* es `Replace` o `Update`
- *`DATE_FORMAT`* es una cadena de texto con formato de fecha; por ejemplo, MMM d, yyyy

Ejemplo

```
epmautomate importProfiles NewRecProfiles.csv Profiles Replace "MMM d, yyyy"
```

importRates

Importa tipos de cambio de un archivo CSV en el repositorio de Account Reconciliation.

Se aplica a

Account Reconciliation

Roles necesarios

Administrador del servicio, Usuario avanzado, Usuario, Visor
Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

`epmautomate importRates` *`PERIOD RATE_TYPE REPLACE_MODE FILE_NAME`* donde:

- *`PERIODO`* es el nombre de un periodo
- *`RATE_TYPE`* es un tipo de cambio predefinido
- *`MODO_REEMPLAZO`* es `Replace` o `ReplaceAll`

- *FILE_NAME* es el nombre del archivo CSV del que se van a importar los tipos

Ejemplo

```
epmautomate importRates "January 2015" Actual ReplaceAll avgrates.csv
```

importRCAttributeValues

Importa valores de atributo en la lista de Reconciliation Compliance o los atributos de grupo de Account Reconciliation.

Se aplica a

Account Reconciliation

Roles necesarios

- Administrador del servicio
- Usuario avanzado (requiere seguridad adicional proporcionada mediante ACL)

Sintaxis

`epmautomate importRCAttributeValues` *`ATTRIBUTE_NAME FILE_NAME`* `[METHOD=REPLACE|REPLACE ALL|UPDATE][DATEFORMAT=DD/MM/YYYY|DD-MMM-YYYY|MMM d,yyyy|All]`, donde:

- *`ATTRIBUTE_NAME`* es el nombre de una lista o un atributo de grupo en el que se importarán los valores.
- *`FILE_NAME`* es un archivo de importación CSV del que se importarán los valores. Utilice el comando uploadFile para cargar este archivo en el entorno antes de ejecutar este comando.
- De manera opcional, `METHOD` determina cómo se importarán los valores. Valores válidos:
  - `Replace` para agregar todos los valores del archivo de importación como el valor de atributo en Reconciliation Compliance. Los valores de atributo existentes se reemplazarán por los del archivo de importación; se agregarán los valores que no existan en el atributo pero estén presentes en el archivo de importación. No se modificarán los valores que existan en el atributo que no estén en el archivo de importación. Tenga en cuenta que todos los datos de atributo de un valor de clave concreto se reemplazarán por el contenido del archivo o se borrarán. Los nuevos valores se agregarán en la parte inferior en el orden en el que aparecen en el archivo. Este tipo de importación resulta muy útil si solo mueve los cambios más recientes desde un sistema de origen; por ejemplo, al agregar nuevos datos de almacenamiento desde una adquisición para reemplazar solo los valores de atributo especificados, si están presentes, por los valores del archivo de importación. Es la opción predeterminada.
  - `Replace All` para reemplazar el valor de atributo existente por los valores de la importación. Se suprimirán los valores existentes en el atributo que no estén presentes en el archivo de importación.
    Este tipo de importación resulta muy útil para replicar los valores de un sistema de origen con una actualización completa; por ejemplo, para completar actualizaciones semanales que se sincronicen con los datos de la tienda del sistema de origen.
  - `Update` para reemplazar o agregar todos los valores del archivo de importación en el atributo. Los valores de atributo existentes se reemplazarán por los del archivo de importación. Se agregarán los valores que estén en el archivo de importación que no estén presentes en el atributo. No se modificarán los valores que existan en el atributo

que no estén presentes en el archivo de importación. Solo los datos de atributo de un valor de clave concreto se reemplazarán por el contenido del archivo; los datos de atributos no disponibles en el archivo no se modificarán. Cualquier clave presente en el archivo de importación, que no esté en el atributo, causará un error.
Este tipo de importación resulta muy útil para actualizar algunos atributos en todos los valores de atributo; por ejemplo, mientras se actualizan los gerentes de tienda después de una reorganización sin que afecte a los datos restantes de la tienda.

- De manera opcional, `Dateformat` especifica los formatos de fecha válidos (por ejemplo, `DD/MM/YYYY`, `DD-MMM-YYYY` (valor predeterminado), `MMM d,yyyy` y `All`) para el análisis. Puede especificar varios valores de formato de fecha separados por puntos y comas.

Ejemplo

```
epmautomate importRCAttributeValues Stores StoreData.csv METHOD=Replace
DATEFORMAT="All"
```

importReconciliationAttributes

Importa los atributos de conciliaciones en las conciliaciones existentes de un archivo que un administrador del servicio haya cargado en el entorno de Account Reconciliation con el comando uploadFile.

Se aplica a

Account Reconciliation

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

`epmautomate importReconciliationAttributes` *`FILE`*`.CSV Period [Rules=`*`RULE_NAME`*`] [Reopen=true|false] [Dateformat=`*`DATE_FORMAT`*`]`, donde:

- `FILE` es el nombre del archivo CSV que contiene los atributos de conciliación que desea importar en las conciliaciones.
- `Period` identifica el periodo al que pertenecen las conciliaciones.
- `Rules`, opcionalmente, identifica las reglas que se van a ejecutar en las conciliaciones afectadas tras importar los atributos. Use las comas para separar los distintos nombres de reglas. Los valores válidos son:
  - `None`: no ejecuta ninguna regla en las conciliaciones afectadas. Este es el valor predeterminado; no se debe combinar con otros valores.
  - `ALL`: ejecuta todas las reglas definidas para las conciliaciones para el periodo especificado. Este valor se debe usar solo; no se puede combinar con otros nombres de reglas.
  - `SET_ATTR_VAL`: ejecuta la regla predefinida para establecer el valor de atributo.

  - `CRT_ALT`: ejecuta la regla predefinida para crear la alerta.
  - `AUTO_APP`: ejecuta la regla predefinida para aprobar automáticamente la conciliación.
  - `AUTO_SUB`: ejecuta la regla predefinida para enviar automáticamente la conciliación.
  - `EMAIL_ON_SAVE`: ejecuta la regla predefinida para enviar automáticamente el correo electrónico tras actualizar la conciliación.
- `Reopen`, opcionalmente, especifica si se va a reabrir las conciliaciones cambiadas al finalizar la operación de importación. El valor predeterminado es `false`.
- `Dateformat`, opcionalmente, especifica los formatos de fecha válidos (por ejemplo, `MM-dd-yyyy`, `dd-MMMM-yy`, `MMM d` y `yyyy` que analizar. Puede especificar varios valores de formato de fecha separados por puntos y comas.

Ejemplos

- **Importación de valores de atributos para un periodo y ejecución de varias reglas con varios formatos de fecha:**
  ```
  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"
  ```
- **Importación de valores de atributos para un periodo sin ejecutar reglas:**
  ```
  epmAutomate importReconciliationAttributes Reconciliations.csv "July 2020"
  ```
- **Importación de valores de atributos para un periodo, ejecutando todas las reglas aplicables y reabriendo las conciliaciones afectadas:**
  ```
  epmAutomate importReconciliationAttributes Reconciliations.csv "July 2020"
  Rules=ALL Reopen=true
  ```

importSnapshot

Importa el contenido de una instantánea en el entorno de servicio. La instantánea que importe debe estar disponible en la ubicación de carga predeterminada.

Un administrador del servicio utiliza el comando uploadFile para cargar una instantánea o el comando copySnapshotFromInstance para copiarla de otra instancia.

- Los siguientes elementos no forman parte de las instantáneas de aplicación de Planning, Planning Modules y FreeForm:
  - Datos de auditoría
  - Datos de la consola de trabajos

  Use el comando cloneEnvironment o la función Clonar entorno si desea copiar los datos de auditoría y de la consola de trabajos en el entorno de destino.
  La importación de la instantánea puede fallar si el proceso de negocio de Planning contiene un miembro de período definido cuyo nombre se ha cambiado que se ha suplantado por un miembro de período personalizado. Por ejemplo, si ha cambiado el nombre del miembro de período *YearTotal* definido a *unused_YearTotal* y, a continuación, ha agregado, un miembro de período de tipo alternativo con el nombre de miembro definido original (*YearTotal* en este ejemplo). En este escenario, la importación de la instantánea puede fallar.
- Las instantáneas no contienen los datos de la tabla temporal de Data Management. Para importar estos datos, use los comandos exportDataManagement y importDataManagement o la interfaz de scripts de mantenimientos del sistema de Data Management. Puede usar el comando cloneEnvironment o la función Clonar entorno para

crear una copia idéntica del entorno, incluidos los datos de la tabla temporal de Data Management.

Las actividades que puede realizar con este comando dependen de su rol.

- Los administradores de servicio solo pueden importar artefactos de aplicación en un entorno.
- Necesita los roles de administrador del servicio y administrador del dominio de identidad para importar contenido de la aplicación en el entorno de servicio, y artefactos de dominio de identidad (usuarios y sus asignaciones de roles predefinidas) en el dominio de identidad del entorno.
  Si se hace referencia a un usuario que no está en el dominio de identidad en la instantánea que se está importando, este comando crea un usuario en el dominio de identidad y asigna la contraseña predeterminada que se especifica en el comando, o bien una contraseña única temporal a cada usuario si no especifica una contraseña en el comando. Por defecto, será necesario que el usuario restablezca la contraseña durante el primer inicio de sesión.

> **Nota:**
>
> - En el caso de procesos de negocio que no sean de Account Reconciliation, Profitability and Cost Management ni de Oracle Fusion Cloud Enterprise Data Management: al cargar los metadatos, Oracle Fusion Cloud Enterprise Performance Management puede hacer varias pasadas de carga si en el intento anterior se han rechazado registros porque los miembros compartidos aparecían antes que los miembros base en el esquema. Estos intentos pueden aumentar el tiempo de procesamiento del comando.
> - A los usuarios que sean miembros de grupos en Control de acceso se les debe asignar un rol predeterminado. No están permitidos los intentos de asignar a un grupo un usuario que no esté previamente asignado a un rol predefinido.

Se aplica a

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

Roles necesarios

- Administrador del servicio
- Usuario avanzado y el rol de aplicación Migraciones - Administrar (Oracle Enterprise Data Management Cloud y Profitability and Cost Management)

Se necesita el rol Administrador de dominio de identidad para importar usuarios y asignaciones de roles predefinidas.

Sintaxis

`epmautomate importSnapshot` *`SNAPSHOT_NAME`* `[importUsers=true|false] [userPassword=`*`DEFAULT_PASSWORD`*`] [resetPassword=true|false]`, donde:

- *`SNAPSHOT_NAME`* es el nombre de una instantánea de la ubicación de carga predeterminada.

- `importUsers`, opcionalmente, especifica si se van a importar los usuarios y sus asignaciones de roles predefinidos de la instantánea. El valor predeterminado es `false`. Use `importUsers=true` para importar usuarios y asignaciones de roles predefinidos en el dominio de identidad si la instantánea de origen contiene datos sobre los usuarios nuevos o si se han asignado nuevos roles a los usuarios actuales.
  Los valores de User Login no distinguen mayúsculas y minúsculas. Por ejemplo, el valor de inicio de sesión del usuario `jane.doe@example.com` se considera que es idéntico a `Jane.Doe@Example.com` o a cualquier variación de sus mayúsculas o minúsculas. Si cualquiera de las variaciones encuentra una coincidencia con una conexión de usuario existente en el dominio de identidad, con este comando no se importa el usuario de la instantánea.

  > **Nota:**
  >
  > – La importación de usuarios y sus roles predefinidos falla si un usuario que no es un Administrador de dominio de identidad realiza la operación de importación. Se registrará el siguiente error en el Informe de estado de la migración: `Error al importar el artefacto del directorio externo ARTIFACT_NAME. El usuario USER_NAME no tiene autorización para realizar esta operación. El usuario debe tener el rol de administrador de dominio de identidad para realizar esta operación.`
  >
  > – Si no importa los usuarios y un usuario de la instantánea de origen no está asignado a un rol predefinido en el entorno de destino, se muestra un error (`EPMIE-00070: Error al buscar el usuario durante la importación de las funciones asignadas`).

  – Los cambios en los roles predefinidos del usuario se actualizarán en función de los roles asignados en la instantánea de origen. Sin embargo, las asignaciones de roles del destino no se eliminarán para que coincidan con las de la instantánea de origen. Por ejemplo, asuma que a `jdoe` se le ha asignado el rol predefinido Usuario avanzado en el entorno de destino, pero solo tiene el rol Usuario en la instantánea de origen. En esta situación, este comando asigna a `jdoe` el rol Usuario y no elimina la asignación de rol Usuario avanzado en el entorno de destino.

  – Con este comando no se suprimen los usuarios existentes del entorno de destino si no existen en la instantánea de origen. Por ejemplo, `jdoe` tiene una cuenta en el entorno de destino, pero esta cuenta no está en la instantánea de origen. En esta situación, la cuenta de `jdoe` en el entorno de destino no se suprime.

  – Con este comando se agregan usuarios que no existan en el entorno de destino; no se actualizan las propiedades del usuario actual en el entorno de destino, incluso aunque sean distintos en la instantánea de origen. Por ejemplo, si el apellido de `jdoe` en la instantánea de origen está escrito de forma distinta en el entorno de destino, el cambio no se realizará en el entorno de destino.

  – Con este comando no se cambian las contraseñas de los usuarios existentes en el entorno de destino, incluso aunque sean diferentes en las instantáneas de origen.

- `userPassword`, opcionalmente, indica la contraseña predeterminada que asignar a los nuevos usuarios que se crean en el dominio de identidad. La contraseña que especifique debe cumplir los requisitos mínimos para contraseñas. Si no especifica un valor para este parámetro, se asigna una contraseña temporal única a cada usuario.

- *resetPassword*, opcionalmente, indica si el nuevo usuario debe cambiar la contraseña la primera vez que inicia sesión. El valor predeterminado es `true`, que exige que los usuarios nuevos cambien la contraseña la primera vez que inicien sesión. Si este valor se establece en `true`, los usuarios nuevos recibirán mensajes de correo electrónico de activación de la cuenta en que se le solicite que cambien las contraseñas.

Ejemplos

- Importe solo los artefactos de aplicación: `epmautomate importSnapshot April16FullApp`
- Importe artefactos de dominio de identidad y de la aplicación (se necesitan roles de administrador del servicio y de administrador de dominio de identidad):
  - Asigne una contraseña temporal única a cada usuario nuevo y fuerce a los usuarios a restablecerla tras iniciar sesión la primera vez:
    `epmautomate importSnapshot April16FullApp importUsers=true`
  - Asigne una contraseña específica y permita a los usuarios no cambiarla si así lo deciden. No se recomienda para importaciones en entornos de producción:
    `epmautomate importSnapshot April16FullApp importUsers=true userPassword=P@ssw0rd1 resetPassword=false`

importSupplementalCollectionData

Importa datos de recopilación adicionales de un archivo a la aplicación.

Utilice el comando uploadFile para cargar el archivo que contiene los datos en la ubicación de carga predeterminada. El formato del archivo de importación es el siguiente:

```
#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
```

Por ejemplo:

```
#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
```

Se aplica a

Financial Consolidation and Close, y Tax Reporting.

Roles necesarios

Administrador del servicio

Sintaxis

> **Nota:**
>
> Todos los parámetros de comando deben ir entre comillas dobles.

`epmautomate importSupplementalCollectionData "FILE_NAME""COLLECTION_NAME" "YEAR" "PERIOD" "[FREQUENCY_DIMENSION=MEMBER]"`, donde:

- `FILE_NAME` es el nombre de un archivo CSV, disponible en la ubicación de carga predeterminada, que contiene datos adicionales con un formato adecuado.
- `COLLECTION_NAME` es el nombre de la recopilación en la que se deben importar los datos adicionales del archivo.
- `YEAR` es el miembro de la dimensión Year que usar para la recopilación.
- `PERIOD` es el nombre de la dimensión de periodo que usar para la recopilación.
- *`FREQUENCY_DIMENSION`*, opcionalmente, es el nombre de la dimensión de frecuencia que usar para la recopilación. Puede especificar tantas dimensiones de frecuencia como sea necesario con el formato *`"FREQUENCY_DIMENSION1=MEMBER" "FREQUENCY_DIMENSION2=MEMBER"`*.

Ejemplo

```
epmautomate importSupplementalCollectionData "datafile.csv" "Journal Data
Collection" "FY20" "Jan" "Account=PAYROLL" "JournalID=LNR 113"
```

importSupplementalData

Importa datos adicionales de un archivo a la aplicación.

Utilice el comando uploadFile para cargar el archivo que contiene los datos en la ubicación de carga predeterminada.

Se aplica a

Financial Consolidation and Close y Tax Reporting.

Roles necesarios

Administrador del servicio

Sintaxis

> **Nota:**
>
> Todos los parámetros de comando deben ir entre comillas dobles.

`epmautomate importSupplementalData "FILE_NAME" "DATA_SET_NAME" "YEAR" "PERIOD_NAME" "SCENARIO_NAME"`, donde:

- `FILE_NAME` es el nombre de un archivo CSV, disponible en la ubicación de carga predeterminada, que contiene datos adicionales con un formato adecuado.
- `DATA_SET_NAME` es el nombre del juego de datos en el que se deben importar los datos adicionales del archivo.
- `YEAR` es el año en el que se despliega el juego de datos.
- `PERIOD_NAME` es el nombre del periodo en el que se despliega el juego de datos.
- `SCENARIO_NAME` es el nombre del escenario en el que se despliega el juego de datos.

Ejemplo

```
epmautomate importSupplementalData "DatasetImport.csv" "EmployeeDataSet" "FY17"
"Jan" "Actual"
```

importTemplate

Crea una estructura de aplicación mediante la importación de un archivo de plantilla que existe en profitinbox.

Un administrador del servicio carga un archivo de plantilla en profitinbox mediante el comando uploadFile.

Se aplica a

Profitability and Cost Management

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

`epmautomate importTemplate` *`APPLICATION_NAME File_Name`* `isApplicationOverwrite=true|false` donde:

- *`APPLICATION_NAME`* es el nombre de la aplicación de Profitability and Cost Management que desea crear mediante la importación de la plantilla.
- `nombre_archivo` es el nombre del archivo .ZIP que contiene la plantilla de aplicación. Este archivo debe existir en profitinbox.
- `isApplicationOverwrite` especifica si desea sobrescribir la aplicación existente, si la hay. Especifique este valor de parámetro en minúscula.

Ejemplo

```
epmautomate importTemplate BksML12 template1.zip isApplicationOverwrite=true
```

importTMAttributeValues

Importa valores en atributos de grupo de Transaction Matching de Account Reconciliation.

Un administrador del servicio utiliza el comando uploadFile para cargar el archivo de importación que contiene los valores de atributo de confrontación de transacciones en el entorno antes de ejecutar este comando.

Se aplica a

Account Reconciliation

Roles necesarios

- Administrador del servicio
- Usuario avanzado (requiere seguridad adicional proporcionada mediante ACL)

Sintaxis

`epmautomate importTMAttributeValues` *`ATTRIBUTE_NAME FILE_NAME`* `[METHOD=REPLACE|REPLACE ALL|UPDATE][DATEFORMAT=DD/MM/YYYY|DD-MMM-YYYY|MMM d,yyyy|All]`, donde:

- *`ATTRIBUTE_NAME`* es el nombre de un atributo de grupo en el que se importarán los valores.
- *`FILE_NAME`* es un archivo de importación CSV del que se importarán los valores en Transaction Matching.
- De manera opcional, `METHOD` determina cómo se importarán los valores. Valores válidos:
  - `Replace` para agregar todos los valores del archivo de importación en atributos de grupo de Transaction Matching. Los valores de atributo existentes se reemplazarán por los del archivo de importación; se agregarán los valores que no existan en el atributo pero estén presentes en el archivo de importación. No se modificarán los valores que existan en el atributo que no estén en el archivo de importación. Tenga en cuenta que todos los datos de atributo de un valor de clave concreto se reemplazarán por el contenido del archivo o se borrarán. Los nuevos valores se agregarán en la parte inferior en el orden en el que aparecen en el archivo.
    Este tipo de importación resulta muy útil si solo mueve los cambios más recientes desde un sistema de origen; por ejemplo, al agregar nuevos datos de almacenamiento desde una adquisición para reemplazar solo los valores de atributo especificados, si están presentes, por los valores del archivo de importación. Es la opción predeterminada.
  - `Replace All` para reemplazar el valor de atributo existente por los valores de la importación. Se suprimirán los valores existentes en el atributo que no estén presentes en el archivo de importación.
    Este tipo de importación resulta muy útil para replicar los valores de un sistema de origen con una actualización completa; por ejemplo, para completar actualizaciones semanales que se sincronicen con los datos de la tienda del sistema de origen.
  - `Update` para reemplazar o agregar todos los valores del archivo de importación en el atributo. Los valores de atributo existentes se reemplazarán por los del archivo de importación. Se agregarán los valores que estén en el archivo de importación que no estén presentes en el atributo. No se modificarán los valores que existan en el atributo que no estén presentes en el archivo de importación. Solo los datos de atributo de un valor de clave concreto se reemplazarán por el contenido del archivo; los datos de atributos no disponibles en el archivo no se modificarán. Cualquier clave presente en el archivo de importación, que no esté en el atributo, causará un error.
    Este tipo de importación resulta muy útil para actualizar algunos atributos en todos los valores de atributo; por ejemplo, mientras se actualizam los gerentes de tienda después de una reorganización sin que afecte a los datos restantes de la tienda.
- De manera opcional, `Dateformat` especifica los formatos de fecha válidos (por ejemplo, `DD/MM/YYYY`, `DD-MMM-YYYY` (valor predeterminado), `MMM d,yyyy` y `All`) para el análisis. Puede especificar varios valores de formato de fecha separados por puntos y comas.

Ejemplo

```
epmautomate importTMAttributeValues TMGA TMGA.csv METHOD=Replace DATEFORMAT="All"
```

importTmPremappedTransactions

Para un origen de datos específico, importa datos de transacciones previamente asignadas desde un archivo del repositorio de Account Reconciliation en la Confrontación de transacciones.

Un administrador del servicio utiliza el comando uploadFile para cargar el archivo de transacciones en el servicio.

Con este comando se muestra el estado de la importación y un archivo de registro de importación en la consola. Un administrador del servicio utiliza el comando downloadFile para descargar el archivo de registro en un equipo local.

Consulte Importación de datos en *Conciliación de cuentas con Account Reconciliation* para obtener información sobre requisitos de formato de archivos de importación y la importación de datos.

> **Nota:**
>
> - Puede importar transacciones solo para una coincidencia a la vez. Sin embargo, las importaciones paralelas se pueden ejecutar en distintos tipos de coincidencias.
> - A diferencia de la pantalla Trabajos, solo puede importar datos de transacciones previamente asignadas de un archivo a la vez.
> - Después de importar transacciones previamente asignadas para todos los orígenes de datos, ejecute el comando `runautomatch`.

Se aplica a

Account Reconciliation

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

`epmautomate importTmPremappedTransactions` *`MATCH_TYPE DATA_SOURCE FILE_NAME [DATE_FORMAT]`* donde:

- `MATCH_TYPE` es un tipo de coincidencia definido en Account Reconciliation.

- `DATA_SOURCE` es el identificador del origen de datos asociado al tipo de conciliación que ha especificado.
- `FILE_NAME` es el nombre del archivo CSV que contiene las transacciones que deben importarse. Este archivo debe estar disponible en el servicio.
- `DATE_FORMAT` es un parámetro opcional que indica el formato de los campos de fecha incluidos en el archivo de importación de transacciones. El valor predeterminado es `dd-MMM-YYYY`. Otros formatos de datos admitidos son los siguientes: `MM/dd/yyyy`, `dd/MM/yyyy`, `MM-dd-yyyy`, `d-M-yyyy` y `MMM d.yyyy`.

Ejemplo

```
epmautomate importTmPremappedTransactions "INTERCOMPANY" "AP" dailydata.csv d-M-yyyy
```

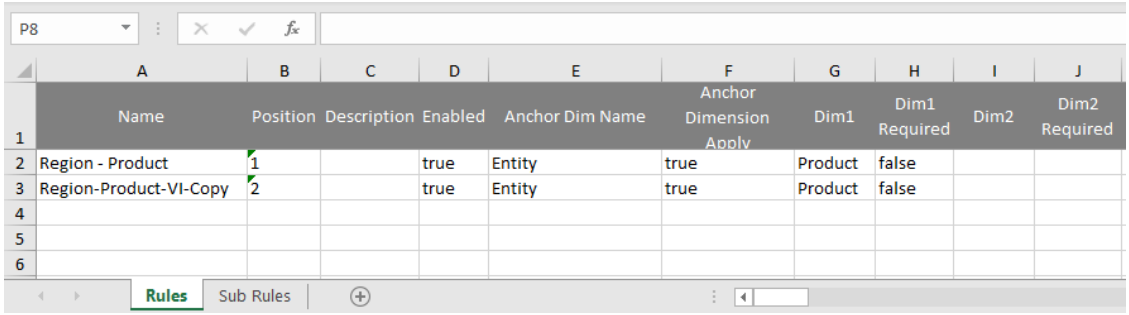
importValidIntersections

Importa grupos de intersecciones válidas de un archivo ZIP que contiene un archivo Excel con definiciones de intersecciones válidas en el proceso de negocio. Antes de ejecutar este comando, utilice el comando uploadFile para cargar el archivo de importación en el entorno.

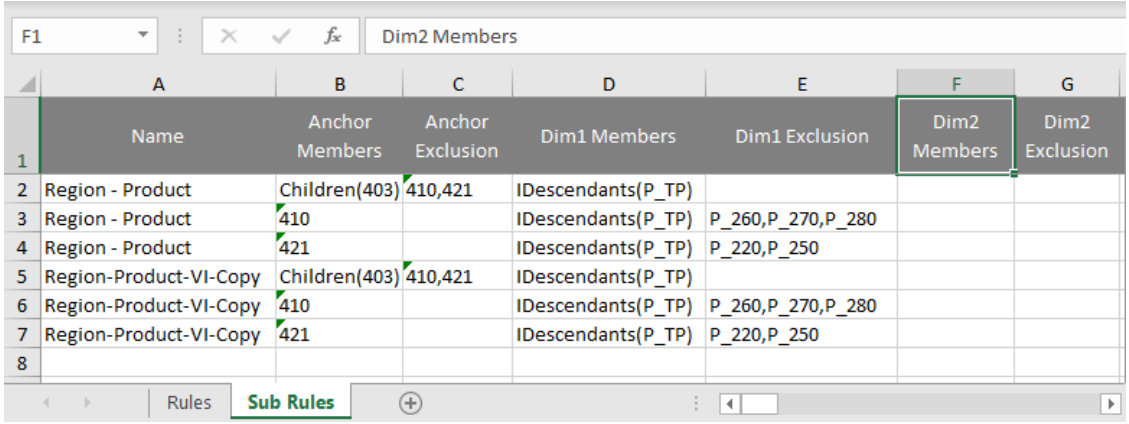
El archivo ZIP de importación debería contener un archivo de Excel con dos hojas de trabajo (Reglas y Subreglas) para importar correctamente las intersecciones válidas. La primera hoja, Reglas, debería definir el grupo de intersecciones, las dimensiones incluidas y propiedades como Valores no especificados válidos y Dimensiones adicionales necesarias. La segunda hoja, Subreglas, debería proporcionar selecciones y exclusiones de miembros. Para obtener más información, consulte estos temas en *Administración de Planning*.

- Dimensiones de anclaje y sin anclaje
- Ejemplos de intersecciones válidas

El mejor método para obtener la plantilla de formato del archivo de importación es exportar las intersecciones válidas de la aplicación. En las siguientes ilustraciones se muestra un formato de ejemplo.



	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										



F1 | Dim2 Members

	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							

Rules | Sub Rules

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate importValidIntersections` *`FILE_NAME`*`.zip [ErrorFile=`*`ERROR_FILE_NAME`*`.txt]`, donde:

- `FILE_NAME` es el nombre del archivo ZIP que contiene el archivo de Excel con la definición de intersección válida.
- `ErrorFile`, opcionalmente, identifica el nombre del archivo de texto en el que se escribirán los registros de error. Si no se especifica este valor de parámetro, el comando genera automáticamente un archivo de error; puede ver su nombre en la consola de trabajos.

Ejemplo

```
epmautomate importValidIntersections VI_Import_File.zip
ErrorFile=VI_Import_Log.txt
```

invalidLoginReport

En entornos de OCI (Gen 2), crea el informe de inicios de sesión no válidos, donde se enumeran los intentos fallidos iniciar sesión en el entorno durante el período de tiempo especificado que corresponde al período de retención de auditoría especificado para el entorno. El período de retención predeterminado es de 30 días. Puede ampliarlo a un máximo de 90 días cambiando el valor de **Período de retención de auditoría (días)** en la consola de Oracle Cloud Identity. Para retener los datos de auditoría durante un período superior a 90 días, descargue y archive este informe de forma periódica, así como el Informe de auditoría de asignación de roles.

El informe de inicios de sesión no válidos incluye información como la siguiente:

- Nombre del usuario que ha intentado iniciar la sesión
- Dirección IP remota desde la que el usuario ha intentado iniciar la sesión
- Registro de hora del intento de inicio de sesión

En este informe se muestran todos los intentos de inicio de sesión incorrectos a la instancia correspondiente de Identity Cloud Service. Es posible que no todo esto esté relacionado con una instancia de 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

Utilice el comando downloadFile para descargar el informe en un equipo local.

Se aplica a

Planning, Planning Modules, 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 y Sales Planning.

Roles necesarios

Administrador de dominio de identidad y cualquier rol predefinido

Sintaxis

`epmAutomate invalidLoginReport` *`FROM_DATE TO_DATE FILE_NAME`*`.CSV`, donde:

- *`FROM_DATE`* indica la fecha de inicio (con formato `YYYY-MM-DD`) del periodo para el que se va a generar el informe. Esta fecha debe estar dentro del período de retención de datos de auditoría especificado en la consola de Oracle Cloud Identity.
- *`TO_DATE`* indica la fecha de finalización (con formato `YYYY-MM-DD`) del periodo para el que se va a generar el informe.
- *`FILE_NAME`* es el nombre de un archivo CSV para el informe.

> **Note:**
>
> Este informe solo se puede generar para los últimos 90 días.

Ejemplo

```
epmAutomate invalidLoginReport 2021-06-01 2021-06-30 invalidLoginReport.CSV
```

listBackups

Muestra las instantáneas de copia de seguridad disponibles de entornos para determinar si hay disponible una copia de seguridad específica para que pueda archivarla o usarla para restaurar el entorno actual usted mismo.

Antes de intentar restaurar una copia de seguridad específica, utilice este comando para comprobar si la copia de seguridad necesaria está disponible en Oracle Object Storage. Si la copia de seguridad está disponible, puede restaurarla (copiarla en su entorno) ejecutando el comando restoreBackup. Después de duplicar la copia de seguridad, puede importarla con el

comando importSnapshot. Al llevar a cabo una restauración de autoservicio de un entorno conseguirá ahorrar tiempo de procesamiento.

En servicios que no sean Narrative Reporting, con este comando se muestran las instantáneas de copia de seguridad disponibles (creadas mediante el proceso de mantenimiento diario) con la convención de nomenclatura `YYYY-MM-DDTHH:MM:SS/Artifact_Snapshot.zip`; por ejemplo, `2022-02-16T21:00:02/Artifact_Snapshot.zip`. En Narrative Reporting, las instantáneas disponibles usan la convención de nomenclatura `YYYY-MM-DDTHH:MM:SS/EPRCS_Backup.tar.gz`; por ejemplo, `2022-02-16T21:00:02/EPRCS_Backup.tar.gz`. En ambos casos, el registro de hora refleja la hora UTC de creación de la instantánea. En la siguiente ilustración se muestra la salida de un comando de ejemplo.

```
c:\Oracle\EPM Automate\bin>epmautomate listbackups

 2022-03-04T06:37:51/Artifact_Snapshot.zip
 2022-03-08T06:32:01/Artifact_Snapshot.zip
 2022-03-09T12:08:05/Artifact_Snapshot.zip
 2022-03-10T06:37:48/Artifact_Snapshot.zip
 2022-03-15T06:21:28/Artifact_Snapshot.zip
 2022-03-16T06:20:52/Artifact_Snapshot.zip
 2022-03-16T12:13:56/Artifact Snapshot.zip

 Total 7
```

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado y el rol de aplicación Migraciones - Administrar (Oracle Enterprise Data Management Cloud y Profitability and Cost Management)

Sintaxis

```
epmAutomate listBackups
```

Ejemplo

```
epmAutomate listBackups
```

listFiles

Muestra los nombre de los archivos de la ubicación predeterminada, las carpetas de Data Management y profitinbox/profitoutbox (Profitability and Cost Management).

Con este comando también se muestran los archivos de exportación incrementales y de copia de seguridad, las instantáneas de migración, los registros de acceso y los informes de actividad. En esta ilustración se muestra una versión truncada de la salida del comando.

```
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
```

Con este comando no se mostrará la instantánea actual si este comando se ejecuta mientras se genera la instantánea del entorno; por ejemplo, durante el mantenimiento diario.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado y el rol de aplicación Migraciones - Administrar (Oracle Enterprise Data Management Cloud y Profitability and Cost Management)

Sintaxis

```
epmautomate listFiles
```

Ejemplo

```
epmautomate listFiles
```

loadData

Carga datos en un cubo de cálculo mediante un archivo disponible en profitinbox.

Un administrador del servicio utiliza el comando uploadFile para cargar archivos en profitinbox.

Se aplica a

Profitability and Cost Management

Roles necesarios

- Administrador del servicio

- Usuario avanzado

Sintaxis

`epmautomate loadData APPLICATION_NAME dataFileName=File_Name PARAMETER=VALUE`
donde:

- *APPLICATION_NAME* es el nombre de la aplicación de Profitability and Cost Management en la que desea cargar los datos.
- `dataFileName=File_Name` especifica un archivo de carga de datos disponible en profitinbox
  El nombre del archivo de datos debe ir entre comillas dobles.
- `PARAMETER=VALUE` indica los parámetros de tiempo de ejecución y sus valores para cargar datos. Especifique tantos pares de parámetro y valor como requiera el proceso. Parámetros válidos y sus valores:
  - `clearAllDataFlag=true|false` especifica si se van a borrar los datos existentes en el cubo de la aplicación
  - `dataLoadValue=OVERWRITE_EXISTING_VALUES|ADD_TO_EXISTING` especifica cómo manejar los datos existentes

Ejemplo

```
epmautomate loadData BksML12 dataFileName="data1.txt"clearAllDataFlag=true
dataLoadValue="OVERWRITE_EXISTING_VALUES"
```

loadDimensionViewpoint

Carga datos en el punto de vista no enlazado, enlazado o parcialmente enlazado (un subconjunto de nodos) de un archivo de carga con una carga definida.

El archivo de carga, un CSV, un archivo de Excel (XLSX) o un archivo ZIP que contiene un archivo CSV o XLSX, debe estar disponible en el entorno en el que carga el punto de vista. Puede cargar el archivo de carga en el entorno mediante el comando uploadFile o copyFileFromInstance.

Para cargar un punto de vista mediante la definición de carga predeterminada, utilice el comando loadViewpoint .

Se aplica a

Oracle Fusion Cloud Enterprise Data Management.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate loadDimensionViewpoint APPLICATION_NAME DIMENSION_NAME FILE_NAME LOAD_NAME [loadOption=ReplaceNodes|Merge] [purpose="PURPOSE"]`, donde:

- `APPLICATION_NAME` es el nombre de una aplicación de Oracle Enterprise Data Management Cloud.
- `DIMENSION_NAME` es el nombre de la dimensión que desea cargar.
- `FILE_NAME` es el nombre del archivo (incluida la extensión: CSV, XLSX o ZIP) desde el que se cargará el punto de vista.

- `LOAD_NAME` es el nombre de la carga definida que se va a utilizar para cargar el punto de vista.
- `loadOption`, opcionalmente, identifica cómo cargar el punto de vista. Las opciones de carga válidas son:
  - `ReplaceNodes` para borrar todas las relaciones de la jerarquía (incluidas las relaciones huérfanas y las utilizadas por otros puntos de vista que utilizan el mismo conjunto de jerarquías) que no sean las del archivo de carga. Este es el tipo de carga predeterminado.
  - `Merge` para conservar las relaciones existentes mediante el procesamiento solo de cambios incrementales.
- `purpose`, opcionalmente, es una cadena de texto, entre comillas dobles, que identifica el motivo para cargar el punto de vista.

Ejemplos

- Cargar el punto de vista reemplazando las jerarquías existentes: `epmautomate loadDimensionViewPoint "Data Warehouse" Citizen Data_Warehouse_Citizen_20241108.csv DW-Citizen`
- Cargar el punto de vista fusionando los cambios incrementales con la carga especificada: `epmautomate loadDimensionViewPoint"Data Warehouse" Citizen Data_Warehouse_Citizen_20241108.csv DW-Citizen loadOption=Merge purpose="Load Test"`

loadDimData

Carga metadatos de dimensión de uno o más archivos de profitinbox en una aplicación.

Un administrador del servicio utiliza el comando uploadFile para cargar archivos de metadatos en profitinbox.

Se aplica a

Profitability and Cost Management

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

`epmautomate loadDimData` *`APPLICATION_NAME`* `dataFileName=`*`File_Name`* `[stringDelimiter=`*`"DELIMITER"`*`] [acceptableDecreasePercentage=PERCENTAGE]`, donde:

- *`APPLICATION_NAME`* es el nombre de la aplicación de Profitability and Cost Management en la que desea cargar los metadatos de dimensión.
- `dataFileName` especifica un archivo de carga de metadatos de dimensión disponible en profitinbox. Para cargar metadatos de varios archivos, indique los nombres de los archivos separándolos con un delimitador.
- Opcionalmente, `stringDelimiter` especifica el delimitador utilizado para separar los nombres de los archivos de metadatos. El delimitador debe ir entre comillas dobles.
- Opcionalmente, `acceptableDecreasePercentage` especifica la diferencia de porcentaje (sin el símbolo %) en el recuento de miembros que será aceptable para la operación. Si el

nuevo recuento de miembros del archivo entrante es inferior al recuento de miembros actual, este valor representa la disminución de porcentaje aceptable. La carga de los datos de dimensión falla si la desviación del recuento de miembros supera el porcentaje.

Ejemplo

```
epmautomate loadDimData BksML12 dataFileName="dimdata1.txt#dimdata1.txt"
stringDelimiter="#" acceptableDecreasePercentage=5
```

loadViewpoint

Carga un punto de vista (un subconjunto de nodos) desde un archivo de carga en una aplicación de Oracle Fusion Cloud Enterprise Data Management.

Las cargas de punto de vista le permiten cargar datos en puntos de vista que están no enlazados, enlazados o parcialmente enlazados El archivo de carga de punto de vista, un CSV, un archivo de Excel (XLSX) o un archivo ZIP que contiene un archivo CSV o XLSX, debe estar disponible en el entorno en el que carga el punto de vista. Puede cargar el archivo de carga en el entorno mediante el comando uploadFile o copyFileFromInstance.

Se aplica a

Oracle Enterprise Data Management Cloud

Roles necesarios

Administrador del servicio

Sintaxis

```
epmautomate loadViewpoint VIEW VIEWPOINT PURPOSE FILE_NAME
[loadType=ReplaceNodes|Merge]
```

, donde:

- `VIEW` es el nombre de una vista de Oracle Enterprise Data Management Cloud.
- `VIEWPOINT` es el nombre del punto de vista que desea cargar.
- `PURPOSE` es una cadena de texto, entre comillas dobles, que indica por qué se carga el punto de vista.
- `FILE_NAME` es el nombre del archivo, con extensión, desde el que se cargará el punto de vista.
- `loadType`, opcionalmente, identifica cómo cargar el punto de vista. Los valores válidos son `Merge` y `ReplaceNodes`.
  - Utilice `Merge` para conservar las relaciones existentes mediante el procesamiento de cambios incrementales.
  - Utilice `ReplaceNodes` para borrar todas las relaciones de la jerarquía (incluidas las relaciones huérfanas y las relaciones utilizadas por otros puntos de vista que utilizan el mismo conjunto de jerarquías) que no sean las del archivo de carga. Este es el tipo de carga predeterminado.

Ejemplos

- **Fusionar cambios incrementales:** `epmautomate loadViewpoint USOperations Entity "Daily Upstream Load" data_Entity.CSV loadType=Merge`

- **Reemplazar las jerarquías existentes:** `epmautomate loadViewpoint USOperations Entity "Replace US Operations data" data_Entity.CSV`

login

Establece una conexión segura a un entorno. Este comando soporta la conexión a un entorno con una contraseña de texto sin formato, un archivo de contraseña cifrado que contenga la contraseña o un símbolo de refrescamiento de OAuth 2.0. Está soportado el inicio de sesión con el símbolo de refrescamiento de OAuth 2.0 solo para entornos de OCI (Gen 2).

Puede iniciar una sesión, que permanecerá activa hasta que se desconecte.

> **Nota:**
>
> - Este comando no está soportado para los usuarios configurados para la autenticación básica con autenticación de varios factores (MFA).
> - EPM Automate no admite el inicio de sesión con las credenciales de inicio de sesión único de su organización.
> - EPM Automate no funciona con el proxy SOCKS; solo funciona con el proxy HTTP/HTTPS.
> - Al utilizar este comando en los archivos por lotes para automatizar actividades, Oracle recomienda utilizar una contraseña cifrada o un símbolo de refrescamiento de OAuth 2.0 para evitar el registro de contraseñas de texto no cifrado en los archivos por lotes.
> - En equipos Windows, este comando identifica automáticamente el certificado de seguridad intermedio del servidor proxy que falta y que puede impedirle establecer una conexión y lo agrega al JRE instalado en `C:\Oracle\EPM Automate`. De esta forma se evitan los errores de conexión relacionados con los certificados de seguridad al usar servidores proxy para acceder a Internet.
> En equipos Linux, el comando login identifica el certificado de seguridad que falta del servidor proxy, lo descarga y muestra un error. Un usuario con acceso `root` puede instalar, a continuación, el certificado descargado en el JRE disponible en el `JAVA_HOME` identificado en las variables de entorno. Consulte las siguientes fuentes de información:
> - Java Runtime Environment y EPM Automate
> - Documentación de keytool de Java

Si está usando una versión anterior de esta, al iniciar sesión, aparece un mensaje para actualizar EPM Automate. Puede utilizar el comando upgrade para actualizar la instalación de forma silenciosa.

Si tiene previsto ejecutar el comando addUsers, removeUsers, assignRole o unassignRole, no se conecte con el símbolo de refrescamiento de OAuth. Con estos comandos debe usar la autentificación básica. El resto de comandos funcionan con OAuth 2.0 en entornos de OCI (Gen 2).

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Sintaxis

- **Con contraseñas cifradas:** `epmautomate login USERNAME PASSWORD URL [IDENTITYDOMAIN] [ProxyServerUserName=PROXY_USERNAME ProxyServerPassword=PROXY_PASSWORD ProxyServerDomain=PROXY_DOMAIN] ] [KeystorePassword=PASSWORD]`
- **Con archivo cifrado:** `epmautomate login USERNAME PASSWORD_FILE URL [IDENTITYDOMAIN] [ProxyServerUserName=PROXY_USERNAME] [ProxyServerPassword=PROXY_PASSWORD] [ProxyServerDomain=PROXY_DOMAIN] [KeystorePassword=KEYSTORE_PASSWORD]`

En estos comandos:

- *USERNAME* es el nombre de usuario del usuario.
- *PASSWORD* es la contraseña del usuario.
- *PASSWORD_FILE* es el nombre y la ubicación del archivo en el que se almacena la contraseña cifrada o el símbolo de refrescamiento de OAuth 2.0 del usuario. Consulte el comando encrypt.
- *URL* es la URL base del entorno al que conectarse. Puede usar una URL personalizada en lugar de la URL de Oracle Fusion Cloud Enterprise Performance Management. Consulte Uso de URL personalizadas en *Guía de introducción para administradores*

> **Nota:**
>
> Si utiliza una puerta de enlace de API o un proxy inverso, use su URL y el contexto definido para su entorno en lugar de la URL de Cloud EPM.

- *IDENTITYDOMAIN*, opcionalmente, es el dominio de identidad del entorno.
  Este valor se deriva automáticamente de la URL de Cloud EPM; se ignorará cualquier valor que se especifique.
- `ProxyServerUserName` es el nombre de usuario utilizado para autentificar una sesión segura con el servidor proxy HTTP que controla el acceso a Internet. Especifique el nombre de usuario sin agregar un prefijo al nombre de dominio. Necesaria solo si está activada la autentificación en el servidor proxy para la red.
- `ProxyServerPassword` es la contraseña con la que el usuario se autentifica en el servidor proxy. Necesaria solo si está activada la autentificación en el servidor proxy para la red. Esta contraseña se puede cifrar. Consulte el comando encrypt. Si esta contraseña está cifrada, se lee del *ARCHIVO_CONTRASEÑAS*.
- `ProxyServerDomain` es el nombre del dominio definido para el servidor proxy HTTP (no el nombre del servidor ni el nombre del host del servidor proxy). Necesaria solo si está

activada la autentificación para la red en el servidor proxy y si se ha configurado un dominio de servidor proxy.

- `KeystorePassword`, opcionalmente, es la contraseña del almacén de claves necesaria para importar el certificado de seguridad del servidor proxy. Use este parámetro solo en Windows y solo si aparecen los siguientes errores en entornos en los que se esté usando un servidor proxy para acceder a Internet:

```
EPMAT-7: Unable to connect as few SSL certificates are missing in the keystore

EPMAT-7: Unable to connect as above-mentioned SSL certificates are missing in the
keystore
```

> **Nota:**
>
> EPM Automate detecta y utiliza la configuración del proxy HTTP/HTTPS de su equipo.
> EPM Automate soporta los siguientes mecanismos de autentificación para conectarse al servidor proxy:
>
> - Autentificación básica
> - Autentificación de resumen
> - Autentificación Kerberos
> - Autentificación de proxy con negociación
> - Autentificación NTLM
>
> El método de autentificación disponible y su configuración depende del servidor proxy que esté usando.
>
> En los equipos Linux, si la configuración de proxy requiere la autentificación con el servidor proxy, deberá introducir el dominio, el nombre de usuario y la contraseña de servidor proxy como parámetros de este comando. Póngase en contacto con el administrador de red si necesita ayuda con el nombre de dominio y las credenciales del servidor proxy.

Ejemplos

- Con una contraseña de Cloud EPM no cifrada, sin autentificación proxy:
  `epmautomate login serviceAdmin P@ssword1 https://test-cloud-pln.pbcs.us1.oraclecloud.com`
- Con un archivo cifrado, sin autentificación proxy:
  `epmautomate login serviceAdmin C:\mySecuredir\password.epw https://test-cloud-pln.pbcs.us1.oraclecloud.com`
- Con un archivo cifrado, si está activada la autentificación en el servidor proxy con un dominio de servidor:
  `epmautomate login serviceAdmin C:\mySecuredir\password.epw https://test-cloud-pln.pbcs.us1.oraclecloud.com ProxyServerUserName=john.doe@example.com ProxyServerPassword=example ProxyServerDomain=example`
- Con un archivo cifrado, si está activada la autentificación en el servidor proxy sin un dominio de servidor:
  `epmautomate login serviceAdmin C:\mySecuredir\password.epw https://test-cloud-pln.pbcs.us1.oraclecloud.com ProxyServerUserName=john.doe@example.com ProxyServerPassword=ejemplo`

- Con una contraseña de servidor proxy y Cloud EPM cifrada, si la autentificación en el servidor proxy está activada con un dominio de servidor:
  ```
  epmautomate login serviceAdmin C:\mySecuredir\password.epw https://test-cloud-pln.pbcs.us1.oraclecloud.com ProxyServerUserName=john.doe@example.com ProxyServerDomain=example
  ```
- Con una contraseña de servidor proxy y Cloud EPM cifrada, si la autentificación en el servidor proxy está activada sin un dominio de servidor:
  ```
  epmautomate login serviceAdmin C:\mySecuredir\password.epw https://test-cloud-pln.pbcs.us1.oraclecloud.com ProxyServerUserName=john.doe@example.com
  ```
- Con un archivo cifrado con puerta de enlace de API de APIGEE:
  ```
  epmautomate login serviceAdmin C:\mySecuredir\password.epw https://exampleapigee.apigee.com/epm example_ID_DOM
  ```
- Uso de una URL personalizada:
  ```
  epmautomate login serviceAdmin C:\mySecuredir\password.epw https://rebrand.ly/Automate
  ```

logout

Finaliza la conexión actual a un entorno.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Sintaxis

```
epmautomate logout
```

Ejemplo

```
epmautomate logout
```

maskData

Enmascara los datos de aplicación para garantizar la privacidad de los datos. Utilice este comando solo en los entornos de prueba para ocultar datos confidenciales a los desarrolladores de aplicaciones.

ADVERTENCIA: No utilice este comando en entornos de producción, porque aleatoriza datos de aplicación actuales, dejándolos carentes de significado. No puede deshacer los efectos de este comando. Si enmascara los datos de un entorno de servicio por error, debe restaurar los datos a partir de una copia de seguridad o de la instantánea de mantenimiento.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate maskData [-f]` donde `-f` es una opción para forzar el inicio del proceso de enmascaramiento sin confirmación de usuario. Si no emplea la opción `-f`, el comando le pedirá que confirme la acción.

Ejemplo

```
epmautomate maskData [-f]
```

mergeDataSlices

Fusiona todos los segmentos de datos incrementales de un cubo de almacenamiento agregado en el segmento principal de la base de datos y, opcionalmente, se eliminan las celdas con el valor cero.

Se aplica a

Planning, Planning Modules, FreeForm, Enterprise Profitability and Cost Management, Sales Planning y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate mergeDataSlices` *`CUBE_NAME`* `[keepZeroCells=true|false]` donde:

- `NOMBRE_CUBO` identifica el cubo de almacenamiento agregado para el que se fusionan todos los segmentos de datos.
- `keepZeroCells`, opcionalmente, especifica si se van a eliminar las celdas que tengan un valor cero (borrar los datos de forma lógica de una región hace que la celda tenga un valor cero). El valor predeterminado es `true`

Ejemplo

```
epmautomate mergeDataSlices rep1 keepZeroCells=false
```

mergeSlices

Fusiona segmentos de datos incrementales en el cubo de base de datos principal y, de manera opcional, elimina las celdas de Oracle Essbase que contienen 0 (cero) como valor para compactar el cubo.

Eliminar las celdas que contienen 0 optimiza el rendimiento del cubo.

Se aplica a

Profitability and Cost Management

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

`epmautomate mergeSlices applicationName [removeZeroCells=true|false]` donde:

- `applicationName` es el nombre de una aplicación de Profitability and Cost Management.
- `removeZeroCells`, especifica, de manera opcional, si se eliminan las celdas que contienen 0. El valor predeterminado de este parámetro es `false`.

Ejemplos

- Fusionar segmentos sin eliminar celdas que contienen 0:
  - `epmautomate mergeSlices BksML30`
  - `epmautomate mergeSlices BksML30 removeZeroCells=false`
- Fusionar segmentos y eliminar las celdas que contienen 0: `epmautomate mergeSlices BksML30 removeZeroCells=true`

optimizeASOCube

Optimiza el rendimiento de consultas para seleccionar vistas de agregado para la extracción de datos desde cubos de ASO.

Este comando permite realizar las operaciones de optimización de consultas en cubos de ASO en aquellos casos en los que la agregación predeterminada no se considere suficiente para satisfacer las necesidades de extracción de datos o de creación de informes debido al gran tamaño de los datos. El proceso de optimización típico es similar a este:

- Borre las agregaciones predeterminadas y basadas en consultas.
- Inicie el seguimiento de consultas.
- Ejecute consultas de ejemplo desde el gestor de consultas de Profitability and Cost Management, Oracle Smart View for Office o Data Management, así como cualquier otra consulta MDX representativa del tipo de consultas que se desee optimizar para entrenar Oracle Essbase.
- Cree la agregación basada en consultas optimizadas o predeterminadas.

Se aplica a

Profitability and Cost Management

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

`epmautomate optimizeASOCube` *`APPLICATION_NAME OPTIMIZATION_TYPE`*, donde:

- *`APPLICATION_NAME`* es el nombre de la aplicación de Profitability and Cost Management a la que pertenece el cubo de ASO.
- *`OPTIMIZATION_TYPE`* es una operación de optimización de cubos. Los valores aceptables son los siguientes:
  - `clearAggregations`, que elimina las vistas predeterminada y basada en consultas.
  - `createAggregations`, que crea las vistas de agregado de Essbase predeterminadas. Use esta opción para realizar la agregación predeterminada en lugar de la agregación basada en consultas
  - `startQueryTracking`, que inicia el seguimiento de las consultas.
  - `stopQueryTracking` que detiene el seguimiento de consultas. Use esta opción para que Essbase deje de recopilar información de optimización. Essbase sigue recopilando la información de optimización hasta que detiene el seguimiento de las consultas o Essbase. Essbase puede cargar vistas de agregado basadas en los datos recopilados hasta que se detenga el seguimiento de consultas.
  - `createQBOAggregations`, que crea vistas de agregado de Essbase basadas en las consultas optimizadas que ejecute tras activar el seguimiento de consultas.

Ejemplos

- Borre las vistas de agregado predeterminada y basadas en consultas:
  ```
  epmautomate optimizeASOCube BksML12 clearAggregations
  ```
- Inicie el seguimiento de consultas.
  ```
  epmautomate optimizeASOCube BksML12 startQueryTracking
  ```
- Cree vistas de agregado de Essbase en función de las consultas optimizadas que ejecute tras activar el seguimiento de consultas:
  ```
  epmautomate optimizeASOCube BksML12 createQBOAggregations
  ```

programDocumentationReport

Crea el informe de documentación de programa que contiene la lógica de la aplicación de Profitability and Cost Management.

Puede descargar el informe en un equipo local mediante el comando downloadFile.

Se aplica a

Profitability and Cost Management

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Sintaxis

`epmautomate programDocumentationReport APPLICATION_NAME POV_NAME [fileName=FILE_NAME] [fileType=PDF|WORD|EXCEL|HTML] [useAlias=true|false] [skipFilters=true|false] stringDelimiter="DELIMITER"`, donde:

- *APPLICATION_NAME* es el nombre de la aplicación de Profitability and Cost Management para la que se va a crear el informe de documentación del programa.
- *POV_NAME* es el nombre del PDV de modelo de la aplicación para el que se va a generar el informe.
- Opcionalmente, `fileName` es un nombre único (incluida la extensión) para el archivo de informe. El nombre de archivo de informe predeterminado es `HPCMMLProgramDocumentationReport_APPLICATION_NAME_POV_NAME.pdf`.
- Opcionalmente, `fileType` es el formato de archivo de salida. El valor predeterminado es `PDF`.
- Opcionalmente, `useAlias` especifica si se van a imprimir alias en lugar de nombre de miembro. El valor predeterminado es `false`.
- `skipFilters`, opcionalmente, especifica si se van a ignorar los filtros para agilizar la generación de informes para modelos grandes que tengan muchas reglas en las que se usen filtros. El valor predeterminado es false.
  Al establecer este parámetro en `true`, se omite el proceso de resolución de cada filtro de regla para determinar los valores del recuento de orígenes estimado, del recuento de destinos estimado y del recuento de destinos estimados en el informe. En su lugar, el proceso utiliza los recuentos de miembros correspondientes de Level0. Si el valor de este parámetro se establece en `false` o no se especifica, el comando resuelve todos los filtros para generar recuentos más precisos.
- `stringDelimiter` es el delimitador utilizado en los valores de PDV. El delimitador debe ir entre comillas dobles.

Ejemplos

- Crear el informe después de resolver los filtros de reglas:
  ```
  epmautomate programDocumentationReport BksML30 2024_Feb_Actual fileName=Feb-Actual.xls fileType=Excel useAlias=true stringDelimiter="_"
  ```
- Crear el informe ignorando los filtros:
  ```
  epmautomate programDocumentationReport BksML30 2024_Feb_Actual fileName=Feb-Actual.xls fileType=Excel useAlias=true skipFilters=true stringDelimiter="_"
  ```

provisionReport

Genera un informe de asignación de roles (archivo .CSV) y lo almacena en la ubicación de descarga predeterminada.

El informe muestra los roles predefinidos (por ejemplo, usuario avanzado de un servicio) y de aplicación (como la asignación masiva, que es un rol de aplicación de Planning) asignados a los usuarios. Utilice el comando downloadFile para descargar el informe.

Se pueden generar dos versiones del informe: simplificada o clásica. El informe simplificado, que es idéntico al informe de asignaciones de roles que está disponible en la pantalla Control de acceso, no muestra los roles de aplicación incluidos en roles predefinidos ni los roles de componentes de roles de aplicación asignados al usuario. La versión clásica del informe muestra los roles de componente incluidos en los roles predefinidos a los que se asignan los

usuarios. Además, muestra los roles de aplicación asignados al usuario (directamente o mediante grupos).

La generación de este informe refresca la información de usuario y rol disponible en Control de acceso.

Solo para OCI (Gen 2): Oracle Fusion Cloud Enterprise Performance Management considera los usuarios desactivados como idénticos a los usuarios no asignados a ningún rol predefinido, incluso aunque dichos usuarios hayan podido tener roles predefinidos cuando se desactivaron. La información sobre los usuarios desactivados no se incluye en este informe.

> **Nota:**
>
> Este comando quedará obsoleto en una próxima versión. En lugar de este comando, use roleAssignmentReport, con el que se obtiene un informe equivalente.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Gestionar
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Ver

Sintaxis

`epmautomate provisionReport` *`REPORT_NAME`* `[format=classic|simplified] [userType=serviceUsers|IDAdmins]` donde:

- *`REPORT_NAME`* es un nombre para el informe.
- `format`, opcionalmente, identifica qué formato se va a aplicar al informe. Valores aceptables:
  - `simplified`, la opción predeterminada, crea un informe idéntico al informe de asignaciones de roles generado desde la pantalla Control de acceso.
  - `classic` crea un informe que muestra los roles de componente incluidos en los roles predefinidos a los que se asignan los usuarios. Además, muestra los roles de aplicación asignados al usuario (directamente o mediante grupos).
- `userType`, opcionalmente, identifica los usuarios que se van a incluir en el informe. Si no especifica un valor para este parámetro, se utiliza el valor predeterminado `serviceUsers`. Valores aceptables:
  - `serviceUsers` crea un informe que contiene información de todos los usuarios funcionales (no incluye los administradores de dominio de identidad si no se han asignado a un rol predefinido que otorgue acceso a la aplicación).
  - `IDAdmins` crea un informe que muestra solo los usuarios asignados al rol de administrador de dominio de identidad. El informe es idéntico en los formatos clásico y simplificado.

Ejemplos

- Cree un informe clásico: `epmautomate provisionReport myProvReport.CSV format=classic`
- Cree un informe simplificado:
  - `epmautomate provisionReport myProvReport.CSV format=simplified`
  - `epmautomate provisionReport myProvReport.CSV userType=serviceUsers`
- Cree un informe que muestre solo administradores de dominio de identidad:
  - `epmautomate provisionReport myProvReport.CSV userType=IDAdmins`
  - `epmautomate provisionReport myProvReport.CSV userType=IDAdmins format=classic`

purgeArchivedTmTransactions

Depura la transacciones coincidentes archivadas de la aplicación Account Reconciliation.

Utilice el comando archiveTmTransactions de forma periódica para archivar las transacciones coincidentes antiguas y ejecute a continuación este comando para eliminarlas de Account Reconciliation a fin de garantizar un tamaño de aplicación óptimo.

Se aplica a

Account Reconciliation

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

`epmautomate purgeArchivedTMTransactions JobID=`*`JOB_ID`* donde `JobID` es el identificador del trabajo de archivado de confrontación de transacciones que se ha ejecutado para archivar las transacciones coincidentes. El ID de trabajo aparece en la consola de EPM Automate cuando ejecuta el comando archiveTmTransactions . También puede encontrarlo en la consola de trabajos.

Ejemplo

```
epmautomate purgeArchivedTMTransactions JobID=100000002655003
```

purgeTmTransactions

Elimina la transacciones confrontadas de Account Reconciliation.

Se aplica a

Account Reconciliation

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

`epmautomate purgeTmTransactions matchType age [filterOperator=VALUE] [filterValue=VALUE] [logFilename=FILE_NAME]`, donde:

- `matchType` es el identificador (TextID) del tipo de confrontación cuyas transacciones confrontadas se deben suprimir.
- `age` identifica el número de días desde la confrontación de la transacción. Se suprimirán las transacciones confrontadas con una fecha anterior o igual a este valor.
- `filterOperator`, opcionalmente, es una de las siguientes condiciones de filtro para identificar las cuentas que contienen transacciones confrontadas para su supresión. Este valor se combina con `filterValue` para identificar las cuentas cuyas transacciones confrontadas se deban depurar:
  - `equals`
  - `not_equals`
  - `starts_with`
  - `ends_with`
  - `contains`
  - `not_contains`
- `filterValue`, opcionalmente, es un valor de filtro para identificar las transacciones que depurar. Si `filterOperator` tiene el valor `equals` o `not_equals`, puede usar una lista separada por comas para especificar varios valores; por ejemplo, `filterValue=101-120 filterValue=102-202`. Si se especifican varios valores, se seleccionan transacciones de cuentas que coincidan con cualquier combinación de valor de filtro y operador de filtro para su depuración.
- `logFilename`, opcionalmente, es el nombre de un archivo log para registrar la información sobre la actividad del comando. Si no se especifica un nombre de archivo, se genera automáticamente un archivo de registro denominado `PurgeTransactions_JOB_ID`.

> **Note:**
>
> Si no se especifican `filterOperator` y `filterValue`, se depurarán todas las transacciones confrontadas anteriores o iguales al valor `age` de todas las cuentas del valor `matchType` especificado.

Ejemplos

- Depure las transacciones confrontadas de `180` días o más para el tipo de confrontación `cashrecon`:
  ```
  epmautomate purgeTMTransactions cashrecon 180 logFileName=tmlogs.log
  ```
- Depure las transacciones confrontadas de `180` días o más para el tipo de transacción `cashrecon` para la cuenta `101-120` o `102-202`:
  ```
  epmautomate purgeTMTransactions cashrecon 180 filterOperator=equals
  filterValue=101-120 FilterValue=102-202
  ```
- Depure las transacciones confrontadas de `180` días o más para el tipo de transacción `cashrecon` para cualquier cuenta que contenga la cadena `11`:
  ```
  epmautomate purgeTMTransactions cashrecon 180 filterOperator=contains
  filterValue=11
  ```

recomputeOwnershipData

Vuelve a calcular los datos de propiedad

En estas situaciones, es necesario volver a calcular los datos de propiedad en Financial Consolidation and Close:

- Después de agregar o suprimir las reglas de sustitución para cuentas de Gestión de propiedad
- Tras cambiar la configuración del rango de métodos de consolidación
- Tras realizar un refrescamiento de la base de datos, con independencia de si la estructura de la entidad ha cambiado

Es necesario volver a calcular los datos de propiedad en Tax Reporting después de cada actualización de la base de datos, aunque no se haya modificado la estructura de la entidad.

Se aplica a

Financial Consolidation and Close y Tax Reporting.

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario

Sintaxis

`epmautomate recomputeOwnershipData` *`Scenario Year Period`* donde:

- `Scenario` es el nombre del escenario que se va a volver a calcular.
- `Year` es el año que se va a volver a calcular.

- `Period` es el primer periodo del año que se va a volver a calcular.
  Se volverán a calcular los periodos seleccionados y todos los periodos posteriores.

> **Nota:**
>
> Un PDV que se deba volver a calcular solo se puede consolidar una vez que se hayan vuelto a calcular los datos de propiedad.

Ejemplo

```
epmautomate recomputeOwnershipData FCCS_total_Actual FY19 Jan
```

recreate

Restaura un entorno a un estado limpio mediante la nueva creación.

Vuelva a crear el entorno para completar las siguientes tareas:

- Limpie un entorno antes de importar una instantánea completa.
- Cambiar el proceso de negocio que se puede desplegar en un entorno.

> **Atención:**
>
> - Este comando suprime la aplicación existente y, de forma opcional, todos los artefactos definidos por el usuario del entorno. Además, vuelve a crear la base de datos y elimina todos los datos existentes. Después de volver a crear el servicio, puede crear un nuevo proceso de negocio o importar uno mediante Migración o EPM Automate.
> - Este comando suprime el historial de migración. Como resultado, el informe sobre el estado de la migración disponible en Migración no contendrá información del historial.
> - Antes de utilizar este comando, realice una copia de seguridad completa del entorno. Puede crear una instantánea de copia de seguridad ejecutando el comando runDailyMaintenance.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate recreate [-f] [removeAll=true|false] [TempServiceType=Service_type]`, donde:

- `-f` fuerza que se inicie el proceso de nueva creación sin confirmación del usuario. Si no emplea la opción `-f`, el comando le pedirá que confirme la acción.
- `removeAll`, opcionalmente, elimina todas las instantáneas y el contenido de la bandeja de entrada (archivos cargados) y del buzón de salida (archivos exportados del entorno). El valor predeterminado es `false`, que mantiene las instantáneas y el contenido de la bandeja de entrada y el buzón de salida.
- `TempServiceType`, opcionalmente, convierte un entorno en un entorno de servicio diferente.
  Los procesos de negocio que puede desplegar en un entorno vienen determinados por el tipo de suscripción de que disponga. Por ejemplo, si tiene una suscripción a EPM Standard Cloud Service, no puede crear una aplicación de formato libre tras convertir el entorno de Account Reconciliation a Planning. Si dispone de una suscripción a EPM Enterprise Cloud Service, puede crear cualquier proceso de negocio en su entorno tras cambiar el tipo de servicio de la forma adecuada. Consulte Acerca de los nuevos servicios de Cloud EPM en la *Guía de introducción para administradores*

  El comportamiento de este parámetro depende de su suscripción.

  – **Suscripciones distintas de EPM Standard Cloud Service y EPM Enterprise Cloud Service:**
    Puede utilizar la opción `TempServiceType` para convertir temporalmente un entorno de Planning, Enterprise Planning, Tax Reporting o Financial Consolidation and Close en un entorno de Account Reconciliation, Oracle Enterprise Data Management Cloud o Profitability and Cost Management. Por ejemplo, si ha adquirido un entorno de Planning, puede convertirlo en un entorno de Account Reconciliation mediante la ejecución del siguiente comando:

    ```
    epmautomate recreate -f removeAll=true TempServiceType=ARCS
    ```

    Tras convertir el entorno en Account Reconciliation, puede convertirlo en un entorno de Oracle Enterprise Data Management Cloud o de Profitability and Cost Management usando el valor `TempServiceType` adecuado. Por ejemplo, para convertirlo en un entorno de Profitability and Cost Management, ejecute el siguiente comando:

    ```
    epmautomate recreate -f removeAll=true TempServiceType=PCMCS
    ```

    Para convertir el entorno de nuevo al tipo de servicio original, ejecute el siguiente comando:

    ```
    epmautomate recreate -f
    ```

    Profitability and Cost Management: puede convertir los entornos de Profitability and Cost Management en entornos de Planning, Enterprise Planning o Enterprise Profitability and Cost Management mediante el siguiente comando:

    ```
    epmautomate recreate -f removeAll=true TempServiceType=PBCS
    ```

    Para convertir el entorno de nuevo al entorno original de Profitability and Cost Management, ejecute el siguiente comando:

    ```
    epmautomate recreate -f TempServiceType=PCMCS
    ```

> **Nota:**
>
> Los entornos de Profitability and Cost Management no se pueden convertir en entornos de Account Reconciliation, Oracle Enterprise Data Management Cloud ni de Narrative Reporting.

– **Suscripciones de EPM Standard Cloud Service y EPM Enterprise Cloud Service:** Puede utilizar la opción `TempServiceType` para convertir un entorno de Oracle Fusion Cloud Enterprise Performance Management en cualquier otro entorno soportado.

  Las suscripciones de EPM Enterprise Cloud Service utilizan una plataforma de Cloud EPM común. Inicialmente, puede desplegar cualquier proceso de negocio de Cloud EPM soportado.

  Para cambiar de un proceso de negocio desplegado a otro, debe volver a crear el entorno mediante la especificación del nuevo tipo de servicio para el entorno. Por ejemplo, si ha creado un proceso de negocio de Account Reconciliation, pero ahora desea crear un entorno de Oracle Enterprise Data Management Cloud, ejecute el comando para volver a crear.

  ```
  epmautomate recreate -f removeAll=true TempServiceType=EDMCS
  ```

  Para convertir un proceso de negocio (por ejemplo, Account Reconciliation) a Planning, Tax Reporting o Financial Consolidation and Close, no especifique un valor `TempServiceType`. Por ejemplo, si ha creado un proceso de negocio de Account Reconciliation, pero ahora desea crear un entorno de Planning Modules, ejecute el comando para volver a crear, como se indica a continuación.

  ```
  epmautomate recreate -f removeAll=true
  ```

Valores de `TempServiceType` aceptables:

– `ARCS` para convertir un entorno en un entorno de Account Reconciliation

– `EDMCS` para convertir un entorno en un entorno de Oracle Enterprise Data Management Cloud

– `EPRCS` para convertir un entorno en un entorno de Narrative Reporting

– `PCMCS` para convertir un entorno en un entorno de Profitability and Cost Management

Ejemplos

- Vuelva a crear el entorno actual, restáurelo en el tipo de servicio original (si se ha emitido un recreate antes con un parámetro `TempServiceType`) sin eliminar instantáneas creadas por el usuario y el contenido de la bandeja de entrada y del buzón de salida:

  ```
  epmautomate recreate -f
  ```

- Vuelva a crear el entorno actual y restáurelo en el tipo de servicio original (si se ha emitido un recreate antes con un parámetro `TempServiceType`), elimine las instantáneas y el contenido de la bandeja de entrada y el buzón de salida:
  ```
  epmautomate recreate -f removeAll=true
  ```

- Vuelva a crear el entorno actual como entorno de Enterprise Data Management y elimine el contenido de la bandeja de entrada y del buzón de salida, así como las instantáneas existentes:
  ```
  epmautomate recreate -f removeAll=true TempServiceType=EDMCS
  ```

- Vuelva a crear el entorno actual de EPM Enterprise Cloud Service Account Reconciliation en un entorno de Financial Consolidation and Close y elimine el contenido de la bandeja de entrada y el buzón de salida, así como las instantáneas existentes:

```
epmautomate recreate -f removeAll=true
```

refreshCube

Refresca el cubo de la aplicación. Normalmente, puede refrescar el cubo después de importar los metadatos en la aplicación.

El tiempo necesario para terminar una operación de refrescamiento del cubo depende de los cambios realizados en la estructura de la aplicación y del impacto que tengan en el cubo. Por ejemplo, un refrescamiento después de actualizar un miembro del cubo de almacenamiento de bloques ligero no puede tardar mucho tiempo, mientras que el refrescamiento de un cubo después de actualizar un miembro del cubo de almacenamiento de bloques denso o un miembro del cubo de almacenamiento de agregado podría tardar una cantidad considerable de tiempo. Debe asegurarse de que la operación de refrescamiento del cubo termina antes de que se realice una copia de seguridad de la aplicación durante el próximo período de mantenimiento.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate refreshCube [JOB_NAME]` donde `JOB_NAME`, opcionalmente, es el nombre de un trabajo de refrescamiento de base de datos definido en la aplicación.
El estado de la operación se repite en la consola desde la que se ejecuta el comando. También puede ver el estado en la página **Actividad reciente** de la pantalla **Trabajos** en la aplicación.

Ejemplo

```
epmautomate refreshCube DaliyCubeRefresh
```

removeUserFromGroups

Elimina la afiliación de un usuario a grupos de Control de acceso identificados en un archivo CSV ANSI o UTF-8 codificado.

El formato del archivo es el siguiente:

```
Group Name
Group1
Group2
```

> **Nota:**
>
> Estos grupos deben existir en Control de acceso. Los valores de nombres de grupos no distinguen mayúsculas y minúsculas.

Un administrador del servicio utiliza el comando uploadFile para cargar el archivo en un entorno.

Cuando finalice, el comando imprime información sobre cada una de las entradas con fallos en la consola. Consulte esta información para saber el motivo por el que se ha producido un fallo en la ejecución del comando en algunas entradas del archivo CSV.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Gestionar

Sintaxis

`epmautomate removeUserFromGroups` *`FILE_NAME User_Login`* donde:

- `FILE_NAME` es el nombre del archivo CSV que contiene los nombres de los grupos de Control de acceso de los que se va a eliminar la afiliación del usuario.
- `User_Login` es el ID de inicio de sesión de un usuario de Oracle Fusion Cloud Enterprise Performance Management cuya afiliación se va a eliminar de los grupos de Control de acceso. Este identificador de inicio de sesión de usuario debe existir en el dominio de identidad que presta servicio al entorno y se debe asignar a un rol predefinido. Este valor no distingue mayúsculas y minúsculas.

Ejemplo

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

removeUsers

Suprime las cuentas identificadas en un archivo CSV codificado por ANSI o UTF-8 que se haya cargado en el entorno desde un dominio de identidad.

El formato del archivo es el siguiente:

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

Un administrador del servicio utiliza el comando uploadFile para cargar el archivo en un entorno. Los valores de User Login no distinguen mayúsculas y minúsculas. Por ejemplo, `jane.doe@example.com` se considera que es idéntico a `Jane.Doe@Example.com` o a cualquier variación de sus mayúsculas o minúsculas.

> **Nota:**
>
> - El archivo CSV no debe incluir la cuenta del usuario que ejecuta este comando.
> - Dado que las cuentas de usuario son comunes a todos los entornos de servicio que soporta un administrador de dominio de identidad, al suprimir una cuenta de un entorno se suprime para todos los entornos que comparten el administrador de dominio de identidad.

Cuando finalice, el comando imprime información sobre cada una de las entradas con fallos en la consola. Consulte esta información para saber el motivo por el que se ha producido un fallo en la ejecución del comando en algunas entradas del archivo CSV.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador de dominio de identidad y cualquier rol predefinido

Sintaxis

`epmautomate removeUsers` *`FILE_NAME`* donde *`FILE_NAME`* es el nombre del archivo CSV que contiene los identificadores de inicio de sesión de los usuarios que se van a eliminar del dominio de identidad.

Ejemplo

```
epmautomate removeUsers remove_users.CSV
```

removeUsersFromGroup

Elimina los usuarios mostrados en un archivo CSV con codificación ANSI o UTF-8 de un grupo mantenido en Control de acceso.

El formato del archivo es el siguiente:

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

Los valores de User Login no distinguen mayúsculas y minúsculas. Por ejemplo, `jane.doe@example.com` se considera que es idéntico a `Jane.Doe@Example.com` o a cualquier variación de sus mayúsculas o minúsculas. Un administrador del servicio utiliza el comando uploadFile para cargar el archivo que contiene inicios de sesión de usuario en el entorno.

Cuando finalice, el comando imprime información sobre cada una de las entradas con fallos en la consola. Consulte esta información para saber el motivo por el que se ha producido un fallo en la ejecución del comando en algunas entradas del archivo CSV.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Gestionar

Sintaxis

`epmautomate removeUsersFromGroup` *`FILE_NAME GROUP_NAME`* donde:

- *`NOMBRE_ARCHIVO`* es el nombre de un archivo CSV que contiene los nombres de inicio de sesión de los usuarios que desea eliminar de un grupo mantenido en Control de acceso.
- *`GROUP_NAME`*es el nombre del grupo Control de acceso del que desea eliminar usuarios. Este valor no distingue mayúsculas y minúsculas.

> **Nota:**
>
> El usuario se elimina de un grupo solo si se cumplen las condiciones siguientes:
>
> - Los inicios de sesión de usuario incluidos en el archivo existen en el dominio de identidad que presta servicio al entorno.
> - El usuario tiene asignado un rol predefinido en el dominio de identidad.

Ejemplo

```
epmautomate removeUsersFromGroup user_file.CSV example_group
```

removeUsersFromTeam

Elimina los usuarios de Oracle Fusion Cloud Enterprise Performance Management que aparecen en un archivo CSV de un equipo.

Si un usuario incluido en el archivo CSV no es miembro del equipo, con este comando se ignora al usuario. Los valores de este archivo no distinguen entre mayúsculas y minúsculas. El formato del archivo CSV es el siguiente:

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

Un administrador del servicio utiliza el comando uploadFile para cargar el archivo .CSV en el entorno.

Se aplica a

Financial Consolidation and Close, Tax Reporting y Account Reconciliation.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido y el rol de aplicación Equipos - Gestionar
- Cualquier rol predefinido y el rol de aplicación Usuarios - Gestionar

Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

`epmautomate removeUsersFromTeam FILE.CSV TEAM_NAME` donde:

- `FILE` identifica un archivo CSV con formato UTF-8 que muestra los identificadores de inicio de sesión de los usuarios que se van a eliminar del equipo.
- `NOMBRE_EQUIPO` identifica un nombre de equipo según la definición de Control de acceso. Este valor no distingue mayúsculas y minúsculas.

Ejemplo

```
epmautomate removeUsersFromTeam example_users.csv example_team
```

renameSnapshot

Cambia el nombre de una instantánea que ha cargado o creado en un entorno.

Si este comando se ejecuta para cambiar el nombre de una instantánea que se esté generando o archivando, verá uno de estos errores:

- `File not found` si la instantánea se está generando
- `Archive process is in progress. Unable to Rename or Delete` si la instantánea se está archivando

No cambie el nombre de una instantánea de mantenimiento en un entorno. Para mantener una copia de seguridad de la instantánea de mantenimiento, debe descargar `Artifact Snapshot` del entorno en un equipo local y, después, cambiarle el nombre según sea necesario. Consulte Descripción general de la instantánea de mantenimiento en *Guía de introducción para administradores*.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado y el rol de aplicación Migraciones - Administrar (Oracle Enterprise Data Management Cloud y Profitability and Cost Management)

Sintaxis

`epmautomate renameSnapshot SNAPSHOT_NAME NEW_SNAPSHOT_NAME` donde:

- *SNAPSHOT_NAME* es el nombre de una instantánea existente. Este valor no debe contener caracteres especiales como espacio, \ (barra invertida), / (barra), * (asterisco), ? (signo de cierre de interrogación), " (comillas), < (menor que) y > (mayor que).
- *NEW_SNAPSHOT_NAME* es el nombre único que desea asignar a la instantánea.

Ejemplo

```
epmautomate renameSnapshot "Example Snapshot" Example_Snapshot_18_09_25
```

replay

Reproduce la carga de Oracle Smart View for Office, la API de REST o la carga de EPM Automate en un entorno con el fin de activar las pruebas de rendimiento si hay una carga muy elevada para verificar que la experiencia del usuario es aceptable cuando el servicio tiene una carga especificada.

Debe crear un archivo de reproducción que identifique las actividades que se deben ejecutar en el servicio. Consulte Preparación para ejecutar el comando replay para obtener información detallada sobre cómo crear el archivo de reproducción.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Sintaxis

`epmautomate replay REPLAY_FILE_NAME.csv [duration=N] [trace=true] [lagTime=t] [encrypt=true|false]`, donde:

- *REPLAY_FILE_NAME* es un archivo CSV en el que se almacenan las actividades que se van a ejecutar en el entorno.
- `Duration`, opcionalmente, indica el durante cuántos minutos se ejecutan las actividades en el entorno.
  Las actividades del archivo HAR se ejecutan una sola vez si no se define este valor. Si las actividades del archivo HAR finalizan en el tiempo especificado por este parámetro, el comando vuelve a ejecutar el archivo HAR hasta que finalicen las actividades. Por ejemplo, imagine que ha establecido `duration=10` para reproducir un archivo HAR que tarda tres minutos en ejecutarse. En este escenario, el comando replay ejecuta las actividades del archivo HAR cuatro veces (unos 12 minutos) hasta que finalice la cuarta iteración.
- `trace=true` es una configuración opcional que indica al comando que cree archivos de rastreo en formato XML.
  Si se especifica esta configuración opcional, el comando crea una carpeta para cada archivo HAR incluido en el archivo CSV de reproducción y almacena todos los archivos de

rastreo relacionados en ella. Por cada actividad en el archivo HAR, este comando genera un archivo de rastreo que contiene la respuesta de Smart View. Los archivos de rastreo se denominan `trace-N.xml`; por ejemplo, `trace-1.xml`, donde `N` es un contador que comienza en 1.

Las carpetas que almacenan los archivos de rastreo se crean en el directorio desde el que se ejecuta EPM Automate. El comando usa una combinación de hora actual del sistema del entorno y nombre del archivo HAR con formato `YYYY_MM_DD_HH_MM_SS_HAR_FILE_NAME` para asignar un nombre a las carpetas. Por ejemplo, si el nombre del archivo HAR es `forecast1.har`, el nombre de la carpeta puede ser `2016_06_08_10_21_42_forecast1`.

- `[lagTime=t]`, opcionalmente, especifica el número de segundos que el comando debe esperar entre la ejecución de cada archivo HAR incluido en el archivo de reproducción. El valor predeterminado es 5 segundos.
  El comando muestra un error si especifica un valor inferior a 5 segundos. Los números negativos (por ejemplo -1) y las fracciones (por ejemplo, 1/2) no se aceptan como valor del parámetro. Se soportan valores decimales.

  Una vez que se ha iniciado la ejecución del primer archivo HAR, el comando espera el número de segundos especificado por este parámetro para iniciar el procesamiento del siguiente archivo HAR. Debido a que las actividades de usuario normalmente no se inician de manera simultánea, establecer este parámetro ayuda a crear una simulación de carga más realista en un entorno.

  Por ejemplo, suponga que desea simular la carga de 1000 usuarios iniciando sesión en un entorno durante la hora punta para realizar actividades. Puede crear archivos HAR para simular estas sesiones y, después, ejecutar este comando con un tiempo de demora de 6 segundos para replicar la carga utilizada en el entorno.
- `encrypt=true|false`, opcionalmente, especifica si se van a cifrar todas las contraseñas incluidas en el archivo de reproducción. El valor predeterminado es `true`. Se utiliza una clave de cifrado aleatoria para cifrar la contraseña.

Consulte Sesión de reproducción de ejemplo para obtener los pasos detallados implicados en la ejecución de este comando.

Ejemplo

```
epmautomate replay forecast1.CSV duration=60 lagTime=5.6
```

resetService

Reinicia el entorno. Opcionalmente, puede ajustar de forma automática el entorno antes de reiniciarlo para asegurarse de que las cachés del índice de Oracle Essbase para los cubos de Opción de almacenamiento de bloques (BSO) estén optimizadas para su aplicación.

De forma predeterminada, los entornos se reinician justo después de finalizar el mantenimiento diario. El ajuste automático del entorno es importante; por ejemplo, después de importar una instantánea en un entorno. Utilice este comando solo cuando observe una degradación grave del rendimiento o si recibe mensajes de error que indiquen que el entorno no puede utilizarse. El reinicio de un entorno no afecta a las personalizaciones de la aplicación (por ejemplo, el cambio de configuración regional, los valores relacionados con el tema y la moneda, ect.). El reinicio tarda hasta 15 minutos.

Antes de utilizar este comando, asegúrese de que las reglas de negocio no se estén ejecutando en el entorno.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate resetService "comment" [AutoTune=true|false] [-f]`, donde:

- `Comment` es una descripción del problema que ha causado que restableciera el entorno. Los comentarios deben escribirse entrecomillados.
- Opcionalmente, `AutoTune` indica si se va a ajustar automáticamente el entorno para optimizar los cubos BSO de las cachés de Essbase de su aplicación. El valor predeterminado es `false`.

  Utilice este parámetro solo en entornos que usen cubos BSO de Essbase: Planning (incluido Planning Modules), Financial Consolidation and Close y Tax Reporting.
- `-f`, de forma opcional, especifica que desea forzar el reinicio del entorno sin interacción adicional del usuario. Si no emplea esta opción, el comando le pedirá que confirme la acción. Esta opción resulta útil si programa un script que utilice este comando.

Ejemplos

- `epmautomate resetService "Users experience slow connections; force restarting the environment" -f`
- `epmautomate resetService "Users experience unacceptably slow connections"`
- `epmautomate resetService "Optimizing the Essbase cache" AutoTune=true`

restoreBackup

Copia una instantánea de copia de seguridad disponible para que esté disponible para su importación en el entorno.

Use el comando listBackups para determinar si la copia de seguridad que desea restaurar está disponible. Al llevar a cabo una restauración de autoservicio de una instantánea en el entorno conseguirá ahorrar tiempo de procesamiento. Después de restaurar la instantánea, use el comando importSnapshot para importarla en el entorno.

> **Note:**
>
> **Solo Narrative Reporting:** utilice este proceso para importar la instantánea restaurada:
>
> 1. Utilice el comando downloadFile para descargar la instantánea restaurada en una máquina local.
> 2. Cambie el nombre de la instantánea descargada en `EPRCS_Backup`.
> 3. Utilice el comando uploadFile para cargar la instantánea con el nuevo nombre (`EPRCS_Backup`) en el entorno. Asegúrese de especificar la ubicación de carga `to_be_imported` en el comando.
> 4. En Narrative Reporting, abra la tarjeta **Mantenimiento diario**.
> a. Seleccione **Usar una instantánea de copia de seguridad cargada**.
> b. Haga clic en **Programar restauración** para importar la instantánea durante el siguiente mantenimiento diario.
> c. Haga clic en **Sí**.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado y el rol de aplicación Migraciones - Administrar (Oracle Enterprise Data Management Cloud y Profitability and Cost Management)

Sintaxis

`epmAutomate restoreBackup` *`SNAPSHOT_NAME`* `[targetName=`*`TARGET_SNAPSHOT_NAME`*`]`, donde:

- *`SNAPSHOT_NAME`* es el nombre de una instantánea de copia de seguridad disponible en los entornos que muestra el comando listBackups.
- `targetName`, opcionalmente, es el nombre de la instantánea de copia de seguridad, sin extensión, en el entorno de destino. Si no especifica este valor, la instantánea de copia de seguridad se restaura en el entorno de destino con *`SNAPSHOT_NAME`*, pero con un _ (guion bajo) reemplazando a la / (barra invertida). Por ejemplo, si el valor de *`SNAPSHOT_NAME`* es `2022-05-14T00:08:56/Artifact_Snapshot.zip`, `targetName` será `2022-05-14T00:08:56_Artifact_Snapshot.zip`.

Ejemplos

- Para servicios que no sean Narrative Reporting:
```
epmAutomate restoreBackup 2022-05-14T00:08:56/Artifact_Snapshot.zip
targetName=example_Artifact_Snapshot
```
- Solo para Narrative Reporting:
```
epmAutomate restoreBackup 2022-02-16T21:00:02/EPRCS_Backup
targetName=Example_EPRCS_Backup
```

restructureCube

Realiza una reestructuración completa de un cubo de almacenamiento de bloques para eliminar o reducir la fragmentación. La reestructuración también elimina los bloques vacíos, y no traspasará ningún cambio de la aplicación al cubo.

> **Nota:**
>
> Antes de ejecutar este comando, asegúrese de que nadie utiliza la aplicación.

Se aplica a

Planning, Planning Modules, FreeForm, Financial Consolidation and Close, Tax Reporting, Sales Planning y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate restructureCube` *`CUBE_NAME`* donde *`CUBE_NAME`* es el nombre de un cubo exactamente igual que en la aplicación

Ejemplo

```
epmautomate restructureCube Plan1
```

roleAssignmentAuditReport

En entornos de OCI (Gen 2), crea un informe de auditoría que muestra los cambios realizados en las asignaciones de roles de aplicación y predefinidas en un período de tiempo que corresponde al período de retención de datos auditoría especificado para el entorno. El período de retención predeterminado es de 30 días. Puede ampliarlo a un máximo de 90 días cambiando el valor de **Período de retención de auditoría (días)** en la consola de Oracle Cloud Identity. Para retener los datos de auditoría durante un período superior a 90 días, descargue y archive este informe de forma periódica, así como el Informe de inicios de sesión no válidos.

En el informe de auditoría de asignación de roles se muestra el nombre de inicio de sesión de un usuario, el nombre del grupo de IDCS o el nombre del grupo de EPM para el que se ha realizado un cambio de rol (en la columna Acción). También incluye el rol o la aplicación predefinidos que se han asignado o cuya asignación se ha anulado, el usuario que ha realizado el cambio de rol (columna Realizado por) y el registro de hora (UTC) con formato de 24 horas cuando se ha completado la acción. En la columna Tipo se indica lo que representa la columna Nombre. Puede tener uno de estos tres valores: Usuario (si en la columna Nombre se identifica un nombre de inicio de sesión de un usuario), Grupo de IDCS (si en la columna Nombre se muestra un nombre de grupo de IDCS) o Grupo de EPM (si en la columna Nombre se muestra un nombre de grupo de 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

Se muestra información sobre los usuarios suprimidos que estaban anteriormente asignados a roles predefinidos en el entorno con el nombre mostrado (nombre y apellido) del usuario en la columna Nombre. En esos casos, en la columna Rol se indica el rol predefinido que el usuario tenía antes de que se suprimiera su cuenta. Este cambio no se aplica a los roles de aplicación, de haberlos, que se hayan asignado al usuario suprimido; esas asignaciones se muestran con el nombre de inicio de sesión del usuario. Para obtener un ejemplo, consulte la información del cuadro rojo de la siguiente ilustración.

	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

Se aplica a

Planning, Planning Modules, 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 y Sales Planning.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido y el rol de administrador de dominio de identidad
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Gestionar
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Ver

Sintaxis

`epmAutomate roleAssignmentAuditReport` *`FROM_DATE TO_DATE FILE_NAME`*`.CSV`, donde:

- *`FROM_DATE`* indica la fecha de inicio (con formato `YYYY-MM-DD`) del periodo para el que se va a generar el informe. Esta fecha debe estar dentro del período de retención de auditoría especificado en la consola de Oracle Cloud Identity.
- *`TO_DATE`* indica la fecha de finalización (con formato `YYYY-MM-DD`) del periodo para el que se va a generar el informe.
- *`FILE_NAME`* es el nombre de un archivo CSV para el informe. Puede descargar el informe generado mediante el comando downloadFile.

Ejemplo

```
epmAutomate roleAssignmentAuditReport 2024-12-11 2025-01-09 RoleAuditReport.CSV
```

roleAssignmentReport

Genera un informe de asignación de roles (.CSV).

Este informe muestra de forma predeterminada los roles predefinidos (por ejemplo, Administrador del servicio) y los roles de aplicación (por ejemplo, Persona que asigna la propiedad de aprobaciones, Supervisor de aprobaciones, Administrador de aprobaciones y Diseñador de procesos de aprobaciones, que son roles de la aplicación Planning) asignados a los usuarios. De manera opcional, este informe también se puede generar para mostrar los administradores del dominio de identidad del entorno. Este informe coincide con la versión CSV del informe de asignaciones de roles generado desde Control de acceso.

	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 generación de este informe refresca la información de usuario y rol disponible en Control de acceso.

Solo para OCI (Gen 2): Oracle Fusion Cloud Enterprise Performance Management considera los usuarios desactivados como idénticos a los usuarios no asignados a ningún rol predefinido, incluso aunque dichos usuarios hayan podido tener roles predefinidos cuando se desactivaron. La información sobre los usuarios desactivados no se incluye en este informe.

> **Nota:**
>
> Este comando genera un informe equivalente al que se crea mediante el comando provisionReport.

Puede descargar el informe mediante el comando downloadFile.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Gestionar

- Cualquier rol predefinido y el rol de aplicación Control de acceso - Ver

Sintaxis

`epmautomate roleAssignmentReport REPORT_NAME.CSV [userType=IDAdmins|serviceUsers]`
donde:

- *`REPORT_NAME`* es un nombre para el informe.
- De manera opcional, `userType` identifica el tipo de usuarios cuya información se incluirá en el informe. El valor predeterminado es `serviceUsers`. Los valores válidos son:
  - `serviceUsers` crea un informe que contiene información de todos los usuarios funcionales (no incluye los administradores de dominio de identidad si no se han asignado a un rol predefinido que otorgue acceso a la aplicación).
  - `IDAdmins` crea un informe que muestra solo los usuarios asignados al rol de administrador de dominio de identidad.

Ejemplos

- Genere un informe que muestre solo los usuarios funcionales:
  - `epmautomate roleAssignmentReport myReport.CSV`
  - `epmautomate roleAssignmentReport myReport.CSV userType=serviceUsers`
- Genere un informe que muestre solo los administradores de dominio de identidad:
  `epmautomate roleAssignmentReport myReport.CSV userType=IDAdmins`

runAutomatch

Ejecuta el proceso de coincidencia automática para hacer coincidir transacciones utilizando las reglas definidas por un administrador del servicio.

> **Nota:**
>
> Ejecute este comando después de importar datos de transacciones en la coincidencia de transacciones con el comando importTmPremappedTransactions o runDataRule.

Puede supervisar el estado del proceso de coincidencia automática en el separador **Historial de trabajos** de Account Reconciliation.

Se aplica a

Account Reconciliation

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

`epmautomate runAutomatch RECONCILIATION_TYPE` donde `RECONCILIATION_TYPE` es un tipo de conciliación definido en Account Reconciliation.

Ejemplo

```
epmautomate runAutomatch INTERCOMPANY
```

runBatch

Ejecuta un lote de Data Management.

Si el modo de ejecución de lote de Data Management está establecido en En serie, se devuelve el control cuando todos los trabajos del lote están completados; si está establecido en Paralelo, se devuelve el control cuando todos los trabajos del lote se envían para la ejecución.

> **Nota:**
>
> Este comando no se puede usar para ejecutar la integración de la carga de datos directa desde orígenes de datos en Oracle Fusion Cloud Enterprise Performance Management. Use EPM Integration Agent para integrar las cargas de datos directas. Para obtener información detallada, consulte Realización de una carga directa con el Agente de integración de EPM en *Administración de Data Integration*.

Se aplica a

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

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

`epmautomate runBatch BATCH_NAME`, donde *`BATCH_NAME`* es el nombre de un lote definido en Data Management.

Ejemplos

```
epmautomate runBatch Accounting_batch
```

runBusinessRule

Inicia una regla de negocio.

Se aplica a

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

Roles necesarios

- Administrador del servicio
- Usuario avanzado (si se otorga el acceso de inicio a la regla)

Sintaxis

`epmautomate runBusinessRule RULE_NAME [PARAMETER=VALUE]` donde:

- *RULE_NAME* es el nombre de una regla de negocio exactamente como se define en el entorno.
- *PARAMETER=VALUE* indica los parámetros de tiempo de ejecución opcionales y sus valores necesarios para ejecutar la regla de negocio.

> **Nota:**
>
> – Con este comando se ejecuta solo una única regla de negocio. Para ejecutar un conjunto de reglas, use el comando runRuleSet.
>
> – La regla se ejecuta en el tipo de plan en el que se ha desplegado.
>
> – Se utilizan los valores predeterminados si no se proporcionan valores para los parámetros de tiempo de ejecución. El comando ignora las peticiones de datos en tiempo de ejecución que no coincidan exactamente con las definidas en la regla.
>
> – Utilice el par *PARÁMETRO=VALOR* para especificar tantas peticiones de datos en tiempo de ejecución como reglas de negocio se necesiten. En el siguiente ejemplo se utilizan dos peticiones de datos en tiempo de ejecución (`Period` y `Entity`) y sus valores (`Q1` y `USA`).
> Consulte Especificación de varios valores para un parámetro para obtener información sobre cómo introducir varios valores para un parámetro.

Ejemplo

```
epmautomate runBusinessRule RollupUSSales Period=Q1 Entity=USA
```

runCalc

Realiza los cálculos en una aplicación.

Mediante el uso de este comando, puede ejecutar cálculos con reglas en un PDV de modelo en datos de un PDV de datos diferente sin copiar las reglas en los distintos PDV.

Se aplica a

Profitability and Cost Management

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

`epmautomate runCalc APPLICATION_NAME POV_NAME [DATA_POV_NAME] PARAMETER=VALUE [comment="comment"] stringDelimiter="DELIMITER"`, donde:

- *APPLICATION_NAME* es el nombre de la aplicación de Profitability and Cost Management que contiene el PDV que se va a calcular.
- *POV_NAME* es el nombre del PDV de modelo que se va a calcular.
- *NOMBRE_PDV_DATOS* es, opcionalmente, el nombre del PDV de datos que se va a calcular mediante el uso de reglas del PDV de modelo.

  Si no se especifica *NOMBRE_PDV_DATOS*, de forma predeterminada se utilizará *NOMBRE_PDV*.

  Solo puede utilizar `exeType=ALL_RULES` si especifica *NOMBRE_PDV_DATOS*.
- `PARAMETER=VALUE` indica los parámetros de tiempo de ejecución y sus valores para ejecutar el cálculo. Especifique tantos pares de parámetro y valor como requiera el proceso. Parámetros válidos y sus valores:
  - `exeType=ALL_RULES|RULESET_SUBSET|SINGLE_RULE` identifica el tipo de ejecución de regla. Es un parámetro necesario.
    Según el valor definido para `exeType`, se pueden especificar los siguientes parámetros:
    * Si especifica `exeType=ALL_RULES`, no incluya el subconjunto de reglas o parámetros relacionados con una única regla como `subsetStart`, `subsetEnd`, `ruleSetName` y `ruleName`. Debe utilizar este `exeType` si establece el parámetro *NOMBRE_PDV_DATOS*.
    * Si especifica `exeType=SINGLE_RULE`, especifique los valores solo para `ruleSetName` y `ruleName`.
    * Si especifica `exeType=RULESET_SUBSET`, especifique los valores para `subsetStart` y `subsetEnd`.
  - `subsetStart` especifica el número de secuencia de la primera regla del conjunto de reglas para ejecutar
  - `subsetEnd` especifica el número de secuencia de la última regla del conjunto de reglas para ejecutar
  - `ruleSetName` identifica el conjunto de reglas que contiene los cálculos que desea realizar
  - `ruleName` es el nombre de la regla que se va a ejecutar (para ejecutar una sola regla)
  - `isClearCalculated=true|false` especifica si se van a borrar los cálculos existentes
  - `isExecuteCalculations=true|false` especifica si se van a realizar cálculos
  - `isRunNow=true|false` defina este valor en `true` para ejecutar el proceso ahora
  - `optimizeReporting=true|false` establezca este valor opcional en `false` si los cálculos se van a realizar sin optimización para la generación de informes. El valor predeterminado es `true`
    Mejores prácticas:
    * Establezca `optimizeReporting=false` solo cuando sea necesario para ahorrar tiempo de procesamiento; por ejemplo, al ejecutar una sola regla o una serie secuencial de varios PDV.

* Cuando ejecute varios trabajos de cálculo simultáneos, establezca `optimizeReporting=true` para todos los trabajos; solo el último trabajo en completarse realizará la agregación, evitando el procesamiento redundante e impidiendo que los trabajos en ejecución se ralenticen.

> **Nota:**
>
> Los valores de parámetros (`true` o `false`) deben ir en minúscula.

- `comment` es un comentario opcional entre comillas dobles
- `stringDelimiter` es el delimitador utilizado en los valores de PDV. El delimitador debe ir entre comillas dobles.

Ejemplo

```
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

Genera un informe que está definido en Conformidad de conciliación.

Consulte las siguientes fuentes de información en *Administración de Account Reconciliation*:

- Uso de informes para obtener instrucciones sobre la definición de informes.
- Generación de informes predefinidos en Conformidad de conciliación para obtener una lista de informes predefinidos de Conformidad de conciliación y los parámetros para generarlos.

Se aplica a

Account Reconciliation

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

`epmautomate runComplianceReport` *`FILE_NAME GROUP_NAME REPORT_NAME REPORT_FORMAT`* `[Param=value]` donde:

- `FILE_NAME` es un nombre de archivo único para el informe que se va a generar. Si ya existe un informe con este nombre en el servidor, este se sobrescribirá. Utilice el comando downloadFile para descargar este informe en un equipo local.
- `GROUP_NAME` es el nombre del grupo con el que está asociado el informe.

- `REPORT_NAME` es un nombre único para el informe que se va a generar.
- `REPORT_FORMAT` es uno de los siguientes formatos para el informe:
  - PDF
  - HTML (no está soportado para gráficos)
  - XLSX (no está soportado para gráficos)
  - CSV
  - CSV2

> **Nota:**
>
> *`REPORT_FORMAT`* `CSV` no permite el formato de los datos en función de una plantilla, mientras que `CSV2` sí lo hace. Generar informes con formato `CSV2` es un proceso más lento que si se genera una salida `CSV`.

- `Param=value`, de forma opcional, identifica los parámetros necesarios para generar el informe. Por ejemplo, el informe `Balance By Account Type` toma dos parámetros: `Period` con el valor `July 2017` y `Currency Bucket` con el valor `Entered`. Debe especificar estos parámetros como `"Period=July 2017" "Currency Bucket=Entered"`.

Ejemplo

```
epmautomate runComplianceReport "Example_File Name""Reconciliation Manager"
"Balance By Account Type" PDF "Period=July 2017" "Currency Bucket=Entered"
```

runDailyMaintenance

Inicia el proceso de mantenimiento de servicio diario inmediatamente, en lugar de esperar a que se muestre la ventana del mantenimiento diario programado.

Este comando le permite forzar la creación de una instantánea de copia de seguridad y actualizar el entorno. Antes de ejecutar este comando, asegúrese de que nadie utiliza el entorno. Este comando no afecta a la programación de mantenimiento diario. Utilice este comando si no desea esperar a la ventana del próximo mantenimiento para que se apliquen los cambios realizados en el entorno; por ejemplo, después de aplicar una revisión puntual.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio, usuario (solo Oracle Enterprise Data Management Cloud)

Sintaxis

`epmautomate runDailyMaintenance [skipNext=true|false] [-f]` donde:

- `skipNext`, opcionalmente, indica si se debe omitir la siguiente aparición del proceso de mantenimiento diario. El valor predeterminado es false.

- `-f`, opcionalmente, indica si se debe forzar el inicio del proceso de mantenimiento sin la confirmación del usuario. Si no emplea la opción `-f`, el comando le pedirá que confirme la acción.

Ejemplos

- Para forzar el inicio de un mantenimiento diario fuera del ciclo sin omitir el siguiente mantenimiento programado: `epmautomate runDailyMaintenance -f`
- Para forzar el inicio de un mantenimiento diario fuera del ciclo y omitir el siguiente mantenimiento programado: `epmautomate runDailyMaintenance skipNext=true -f`
- Para iniciar un mantenimiento diario fuera del ciclo y omitir el siguiente mantenimiento programado: `epmautomate runDailyMaintenance skipNext=true`

runDataRule

Ejecuta una regla de carga de datos de Data Management basada en el periodo inicial y en el periodo final, así como en las opciones de importación o exportación que especifique.

> **Nota:**
>
> Este comando no se puede usar para ejecutar la integración de la carga de datos directa desde orígenes de datos en Oracle Fusion Cloud Enterprise Performance Management. Use EPM Integration Agent para integrar las cargas de datos directas. Para obtener información detallada, consulte Realización de una carga directa con el Agente de integración de EPM en *Administración de Data Integration*.

Se aplica a

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

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

`epmautomate runDataRule` *`RULE_NAME START_PERIOD END_PERIOD IMPORT_MODE EXPORT_MODE [FILE_NAME]`* donde:

- *`NOMBRE_REGLA`* es el nombre de una regla de carga de datos definida en Data Management. Debe entrecomillar el nombre de la regla si contiene espacios.
- *`START_PERIOD`* es el primer período para el que se van a cargar los datos. Este nombre de periodo debe estar definido en la asignación de periodos de Data Management
- *`END_PERIOD`* es, para una carga de datos en varios períodos, el último período para el que se van a cargar los datos. Cuando la carga abarca un solo periodo, introduzca el mismo periodo que el periodo inicial. Este nombre de periodo debe estar definido en la asignación de periodos de Data Management
- *`MODO_IMPORTACIÓN`* determina cómo se importan los datos a Data Management.

La configuración del modo de importación es sensible a mayúsculas/minúsculas. Los valores aceptables son los siguientes:

- `APPEND` para agregar los datos de PDV existentes a Data Management.
- `REPLACE` para suprimir los datos de PDV y reemplazarlos por los datos del archivo
- `RECALCULATE` para volver a calcular los datos
- `NONE` para omitir la importación de datos a la tabla temporal de Data Management

- *`EXPORT_MODE`* determina cómo se exportan los datos a la aplicación.

  La configuración del modo de exportación es sensible a mayúsculas/minúsculas. Los valores aceptables son los siguientes:

  - `STORE_DATA` para fusionar los datos de la tabla temporal de Data Management con los datos existentes. Utilice siempre esta opción de exportación en los trabajos de Data Management utilizados para cargar los metadatos.
  - `ADD_DATA` para agregar los datos de la tabla temporal de Data Management a la aplicación.
  - `SUBTRACT_DATA` para restar los datos de la tabla temporal de Data Management de los datos existentes.
  - `REPLACE_DATA` para borrar los datos de PDV y reemplazarlos por los datos de la tabla temporal de Data Management. Se borran los datos de Escenario, Versión, Año, Periodo y Entidad.
  - `NONE` para omitir la exportación de datos de Data Management a la aplicación.

  > **Nota:**
  >
  > Para Financial Consolidation and Close, solo están soportados los siguientes modos de exportación:
  >
  > - `MERGE` para fusionar los datos de la tabla temporal de Data Management con los datos existentes
  > - `REPLACE` para eliminar entradas de la tabla temporal de DM y sustituirlas por las de la carga de datos
  > - `NONE` para omitir la exportación de datos de Data Management a la aplicación
  >
  > Para Oracle Fusion Cloud como destino, solo se admiten los siguientes modos de exportación:
  >
  > - `MERGE` para fusionar los datos de la tabla temporal de Data Management con los datos existentes
  > - `NONE` para omitir la exportación de datos de Data Management a la aplicación

- *`FILE_NAME`* es un nombre de archivo opcional. Si no especifica ningún nombre de archivo, el comando importa los datos incluidos en el nombre de archivo especificado en la regla de carga de datos. Este archivo debe estar disponible en la carpeta de la bandeja de entrada o en una carpeta dentro de ella.

  Al cargar los archivos de formato Bank Administration Institute (BAI) para Account Reconciliation, no especifique un valor para este parámetro. Siempre debe especificar el nombre de archivo para cargar archivos BAI en la definición de la regla de datos.

> **Nota:**
>
> Si se especifica una ruta en la regla de datos, no especifique la ruta de archivo en el comando. Especifique solo el nombre de archivo. Si no se especifica una ruta en la regla de datos, especifique la ruta completa para el archivo de datos.

Ejemplos

- Importación de varios periodos:
  `epmautomate runDataRule VisionActual Mar-15 Jun-15 REPLACE STORE_DATA inbox/Vision/GLActual.dat`
- Importación de un solo periodo:

  ```
  epmautomate runDataRule "Vision Actual" Mar-15 Mar-15 REPLACE STORE_DATA inbox/Vision/GLActual.dat
  ```

runDMReport

Crea un informe de Data Management y lo almacena en la carpeta `outbox/reports`.

El informe generado se denomina según el ID del trabajo de Data Management que genera el informe y el formato del informe. Por ejemplo, si el ID de trabajo del informe es 2112 y el formato de salida del informe que especifique es PDF, el nombre del informe es `2112.pdf`. El nombre del informe aparece en la consola después de que se genere el informe. También puede identificar el nombre del informe en el separador Detalles del proceso de Data Management o mediante el comando listFiles.

Utilice el comando downloadFile para descargar el informe en un equipo local.

Se aplica a

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

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

`epmautomate runDMReport` *`REPORT_NAME PARAMETER=Value`* `"Report Output Format=[PDF|HTML|XLS|XLSX]"` donde:

- *`REPORT_NAME`* es el nombre de la plantilla de informe de Data Management que se va a utilizar para la generación del informe.
- *`PARAMETER=Value`* indica parámetros de informe y sus valores. Puede especificar tantos parámetros como sea necesario en formato `PARÁMETRO=Valor`. La lista de parámetros necesarios dependerá del informe que desee generar.

> **Nota:**
>
> Los parámetros de tiempo de ejecución de informe se definen al diseñar los informes. Para ejecutar este comando, debe generar y utilizar estos parámetros y valores en el separador Flujo de trabajo. Para generar los parámetros de tiempo de ejecución de un informe, en el separador Flujo de trabajo de Data Management, haga clic en **Ejecución del informe** y, a continuación, seleccione un grupo de **Grupo de informes**. Seleccione el informe para el que desea generar los parámetros y, a continuación, haga clic en **Crear script de informe**. Opcionalmente, puede especificar valores de parámetros de informe y, a continuación, seleccionar un formato de salida y, luego, hacer clic en **Aceptar**. Utilice los parámetros que se muestran en **Generar script de informe** para especificar los parámetros de tiempo de ejecución y los valores para generar el informe.

- `Formato de salida de informe` indica el formato de salida del informe. Las opciones válidas son PDF, HTML, XLS y XLSX. El formato de informe predeterminado es PDF.

Ejemplo

```
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

Ejecuta un trabajo de Data Integration para importar datos en un proceso de negocio de Oracle Fusion Cloud Enterprise Performance Management o exporta datos desde un proceso de negocio en un sistema externo.

Este comando sustituye al comando runDataRule. Oracle recomienda que empiece a usar este comando en lugar del comando runDataRule.

Se aplica a

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

Roles necesarios

- Administrador del servicio
- Usuario avanzado

Sintaxis

```
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]
```

- En las integraciones de modalidad estándar, debe especificar los valores para `importMode`, `exportMode` y `periodName`
- En las integraciones de modalidad rápida, debe especificar el valor para `exportMode`
- Los nombres de parámetros y sus valores distinguen mayúsculas y minúsculas

En este comando:

- *JOB_NAME* es el nombre de un trabajo de integración definido en Data Integration.
- `importMode` determina cómo se importan los datos en Data Integration. Los modos de importación aceptables son:
  - `Append` para agregar los datos de PDV existentes a Data Integration.
  - `Replace` para suprimir los datos de PDV y reemplazarlos por los datos del archivo.
  - `Map and Validate` para omitir la importación de datos y volver a procesar los datos con cuentas lógicas y asignaciones actualizadas.
  - `No Import` para omitir la importación de datos a la tabla temporal de Data Integration.
- `exportMode` determina cómo se cargan los datos en la aplicación de destino. En las integraciones de modalidad rápida, no puede usar `Check` ni `No Export` como valor del parámetro `exportMode`. Los valores de modo de exportación aceptable son:
  - `Merge` para actualizar los datos existentes y agregar nuevos datos.
  - `Replace` para borrar los datos existentes en el PDV y cargarlos con los datos nuevos. En la modalidad estándar, se borran los datos de las dimensiones Scenario, Version, Year, Period y Entity. En la modalidad rápida, los datos se borran para las dimensiones Year, Period y Entity. Puede definir regiones de borrado personalizado para ambas modalidades.
  - `Accumulate` para agregar los datos a los datos existentes. Aplicable a Planning, Planning Modules, Financial Consolidation and Close, Tax Reporting, Profitability and Cost Management y Enterprise Profitability and Cost Management.
  - `Subtract` para restar los datos del balance existente. Aplicable a Profitability and Cost Management y Enterprise Profitability and Cost Management.
    Para las integraciones de modalidad rápida:
    * No puede utilizar `Check` y `No Export` como valor de este parámetro.
    * Para Planning, Planning Modules y Financial Consolidation and Close, los únicos valores válidos son `Replace`, `Merge` y `Accumulate`.
  - `No Export` para omitir la exportación de datos. Use este modo para cargar los datos en la tabla temporal para su revisión antes de cargarlos en la aplicación de destino.
  - `Check` para llevar a cabo solo una comprobación de validación de datos.

    Para Oracle Fusion Cloud como destino, solo se admiten los siguientes modos de exportación:
    * `MERGE` para fusionar los datos de la tabla temporal de Data Integration con los datos existentes
    * `NONE` para omitir la exportación de datos de Data Integration a la aplicación
- `periodName` es el nombre de uno o más periodos o rangos de periodos, cada uno entre llaves, para los que importar o exportar datos. Las convenciones de nomenclatura de periodos aceptables son las siguientes:
  - Para cargas de un solo periodo, especifique el nombre del periodo entre llaves, por ejemplo, `{Jan-21}`
  - Para cargas de varios periodos, incluya los nombres de periodo de inicio y finalización entre llaves, por ejemplo, `{Jan-21}{Mar-21}` (para cargar datos para todos los periodos del 21 de enero al 21 de marzo)
  - **Para Planning, Planning Modules, Financial Consolidation and CloseFreeForm y Tax Reporting:** puede especificar el nombre del periodo y el año del proceso de

negocio con el formato `{Jan#FY21}{Mar#FY21}` para cargar los datos para todos los periodos con inicio el 21 de enero y fin el 21 de marzo.
El nombre de periodo se debe escribir entre llaves.

* Un solo periodo: hace referencia al nombre de periodo de Data Management para un único periodo definido en la asignación de Period.
* Varios periodos: hace referencia a una carga de varios periodos. El parámetro se especifica con el formato `{Month-Year}{Month-Year}`. Por ejemplo, {Jan-20}{Mar-20} para una carga de varios periodos desde el 20 de enero hasta el 20 de marzo.
* Nombre de periodo de Planning: hace referencia a un nombre de periodo de Planning con formato `{Month#Year}`, por ejemplo, {Jan#FY20}{Mar#FY20}. Con esta convención, no tiene que especificar los nombres de periodos de Data Integration. En lugar de especificar los nombres de miembro para las dimensiones Year y Scenario.
  Este parámetro está soportado en los procesos de negocio de Planning,Tax Reporting y Financial Consolidation and Close. Es funcional tanto para sus aplicaciones de servicio como para los despliegues en la nube obtenidos de orígenes de datos locales.

  Usar esta convención es útil si se activa desde un script de Groovy de Cloud EPM mediante la captura de los nombres de miembro Year y Period. La asignación de periodo de aplicación o la asignación de periodo global debe existir con el valor Year y Month en los valores de destino de la asignación de periodo.
* Variable de sustitución: es una extensión del formato de nombre de periodo de Planning anterior donde una variable de sustitución, en lugar de los nombres de miembro Year y Month reales, que se puede especificar con formato `{Month#&CurYr}{&FcstMonth#&CurYr}`; por ejemplo, `{Jan#&CurYr}{&FcstMonth#&CurYr}`.
  Está soportada una combinación tanto de nombres de miembro reales como de variables de sustitución.

  Este formato está soportado en los procesos de negocio de Planning,Tax Reporting y Financial Consolidation and Close.

  La asignación de periodo de aplicación o la asignación de periodo global debe existir en Data Integration del entorno donde de ejecuta el comando, con los valores Year y Month disponibles en los valores de destino de la asignación de periodo. En este caso, Year y Month hacen referencia al valor actual de la variable de sustitución durante la ejecución.
* GLOBAL POV: ejecuta la carga de datos para el periodo de PDV global. Use el formato `{GLOBAL_POV}`.

> **Note:**
>
> Si usa cualquier parámetro de nomenclatura de periodos que no sea el parámetro descrito en esta sección, verá un mensaje de error `Entrada no válida - HTTP 400`.

– `{GLOBAL_POV}` para ejecutar la carga de datos para el periodo definido en el PDV global en el sistema o en la configuración de la aplicación en Data Integration.

> **Note:**
>
> Está soportado el formato de convención de nomenclatura para periodos {Month#Year} para Planning, Planning Modules, Financial Consolidation and Close y Tax Reporting. Según esta convención, puede especificar nombres de miembro para las dimensiones Year y Scenario en lugar de los nombres de periodo de Data Integration. Este enfoque resulta útil si el comando se activa desde un script de Groovy capturando los nombres de miembro Year y Period.
> La convención de nomenclatura de la variable de sustitución `{Jan#&CurYr}` `{&FcstMonth#&CurYr}` es una extensión de la convención de nomenclatura del periodo anterior. Puede especificar una variable de sustitución en lugar de los nombres de miembro Year y Month si está ejecutando este comando en Planning, Planning Modules, Financial Consolidation and Close y Tax Reporting. También está soportada una combinación de nombres de miembro y variables de sustitución.
>
> Las convenciones anteriores de nomenclatura de periodos y de variables de sustitución solo funcionan si las asignaciones de periodos de aplicación o de periodos globales con los valores Year y Month en los valores de destino ya existen en Data Integration.

- `inputFileName`, para cargas de datos basadas en archivos, especifica el nombre del archivo, disponible en la bandeja de entrada, de las que se van a importar los datos. Si especifica el nombre de directorio en la definición de integración, transfiera solo el nombre de archivo. Si no incluye un nombre de directorio en la definición de integración, use el formato `inbox/`*`DIR_NAME/FILE_NAME`*, por ejemplo, `inbox/GLBALANCES.txt` o `inbox/EBSGL/GLBALANCES.txt`. Si el archivo se ha cargado en la ubicación predeterminada del entorno, use la convención `#epminbox/`*`FILE_NAME`*, por ejemplo, `#epminbox/GLBALANCES.txt`, para identificar el archivo de datos de entrada.
  Este parámetro solo se puede aplicar a cargas de datos basadas en archivos nativos. Si no especifica este valor de parámetro para cargas de datos basadas en archivos, con este comando se importan datos del archivo especificado en la definición de integración. Si especifica este valor de parámetro para cargas de datos que no estén basadas en archivos, el comando lo ignora.
- *`PARAMETERS`*, opcionalmente, identifica parámetros de tiempo de ejecución con el formato *`PARAMETER_NAME="PARAMETER"`*. Los parámetros incluyen tanto filtros de origen como opciones de destino.

> **Note:**
>
> El único parámetro que puede usar ahora para un tipo de dimensión (metadatos) de una aplicación de destino es `"Refresh Database"=Yes|No`.

Ejemplos

- Importación de un solo periodo:
  ```
  epmAutomate runIntegration VisionDataLoad importMode=Replace exportMode=Merge
  periodName="{Mar-15}"
  ```
- Importación de varios periodos:
  ```
  epmAutomate runIntegration VisionDataLoad importMode=Replace exportMode=Merge
  periodName="{Mar-15}{Jun-15}"
  ```

- Integración incremental de datos basados en archivos:
  `epmAutomate runIntegration IncrementalFileLoad importMode=Replace exportMode=Merge periodName="{Jan-20}{Mar-20}" inputFileName=File1.txt`

runIntercompanyMatchingReport

Genera un informe de confrontación intercompañía que le ayuda a confrontar correctamente las transacciones entre las entidades y los socios relacionados como parte del proceso de consolidación y para realizar análisis y auditorías.

Se aplica a

Financial Consolidation and Close

Roles necesarios

- Administrador del servicio
- Usuario avanzado (si se otorga seguridad adicional mediante ACL)
- Usuario (si se otorga seguridad adicional mediante ACL)

Sintaxis

`epmautomate runIntercompanyMatchingReport` *`JOB_NAME FILE_NAME`* `[scenario=scenario] [years=years] [period=period_name] [ReportFormat=HTML]`, donde:

- `JOB_NAME` es el nombre de una definición del trabajo de informe intercompañía que existe en la aplicación. Consulte Gestión de informes de confrontación intercompañía en *Trabajar con Financial Consolidation and Close* para obtener más información. Este comando genera el informe de confrontación intercompañía con la configuración disponible en el trabajo. Asegúrese de seleccionar el buzón de salida como ubicación para el informe generado.

  > **Note:**
  >
  > Puede reemplazar la configuración de escenario, años, periodo y formato de informe especificada en el trabajo introduciendo estos valores opcionales al ejecutar este comando.

- `FILE_NAME` es el nombre del archivo de informe. Este informe se genera en el buzón de salida. Utilice el comando downloadFile para descargarlo en un equipo local o el comando sendMail para enviarlo por correo electrónico.
- `SCENARIO` es, opcionalmente, el nombre de un escenario, definido en la aplicación, para el que se va a generar el informe.
- Opcionalmente, `YEARS` es el año para el que se va a generar el informe.
- `PERIOD` es, opcionalmente, el periodo para el que se va a generar el informe.
- Opcionalmente, `ReportFormat` es el formato del informe. Los valores aceptables son HTML, PDF y XLSX.

> **Note:**
>
> - Puede generar este informe para una combinación de un escenario, año y periodo. No especifique varios escenarios, años o periodos.
> - Si no especifica ningún valor para escenario, años, periodo y formato de informe, se utilizan los valores correspondientes especificados en la definición del trabajo de informe (identificado por `JOB_NAME`). Si se especifican, estos valores opcionales reemplazarán los valores establecidos en la definición del trabajo de informe.

Ejemplos

- Creación del informe reemplazando el conjunto de valores en la definición del trabajo:
  ```
  epmautomate runIntercompanyMatchingReport IC_Job_01 SampleICReport.html
  scenario=Actual years=FY22 period=Jan reportFormat=HTML
  ```
- Creación del informe utilizando el conjunto de valores en la definición del trabajo:
  ```
  epmautomate runIntercompanyMatchingReport IC_Job_01 SampleICReport
  ```

runMatchingReport

Genera un informe que está definido en Confrontación de transacciones.

Consulte Generación de informes predefinidos en Coincidencia de transacciones en *Administración de Account Reconciliation* para obtener la lista de informes predefinidos de Confrontación de transacciones y los parámetros para generarlos.

Se aplica a

Account Reconciliation

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

`epmautomate runMatchingReport` *`FILE_NAME GROUP_NAME REPORT_NAME REPORT_FORMAT`* `[Param=value]` donde:

- `FILE_NAME` es un nombre de archivo único para el informe que se va a generar. Si ya existe un informe con este nombre en el servidor, este se sobrescribirá. Utilice el comando downloadFile para descargar este informe en un equipo local.
- `GROUP_NAME` es el nombre del grupo con el que está asociado el informe.
- `REPORT_NAME` es un nombre único para el informe que se va a generar.
- `REPORT_FORMAT` es uno de los siguientes formatos para el informe:

  - PDF
  - HTML (no está soportado para gráficos)
  - XLSX (no está soportado para gráficos)
  - CSV
  - CSV2

> **Nota:**
>
> *REPORT_FORMAT* `CSV` no permite el formato de los datos en función de una plantilla, mientras que `CSV2` sí lo hace. Generar informes con formato `CSV2` es un proceso más lento que si se genera una salida `CSV`.

- `Param=Value`, de forma opcional, identifica los parámetros necesarios para generar el informe. Por ejemplo, para el informe `Match Type Configuration`, que toma el parámetro `status` con el valor `approved`, especifique el parámetro y el valor como `status=Approved`.

Ejemplo

```
epmautomate runMatchingReport Example_FileName "Transaction Matching" "Match Type
Configuration" HTML "status=Approved"
```

runPipeline

Ejecuta un pipeline de Data Integration (organización de una serie de trabajos como un solo proceso) en función de las variables definidas para el pipeline. Para obtener información detallada sobre los pipelines, consulte Uso del pipeline en *Administración de Data Integration para Oracle Enterprise Performance Management Cloud*.

Se aplica a

Planning, Planning Modules, FreeForm, Financial Consolidation and Close, Enterprise Profitability and Cost Management, Tax Reporting, Sales Planning y Strategic Workforce Planning

Roles necesarios

El administrador del servicio o cualquier usuario configurado como usuario proxy para ejecutar una definición de pipeline (mediante la seguridad de la ubicación o mediante la especificación de la opción **Activar ejecución de pipeline para usuarios no administradores** de Configuración del sistema en *Sí*).

Sintaxis

`epmautomate runPipeline` *`PIPELINE_CODE`* `[PARAMETER=VALUE]`, donde:

- *`PIPELINE_CODE`* es un código definido al crear el pipeline en Data Integration. Para obtener más información, consulte Descripción del proceso del pipeline en *Administración de Data Integration*.
- `PARAMETER=`*`VALUE`*, opcionalmente, identifica los parámetros y sus valores para ejecutar el pipeline. Especifique tantos pares de parámetro y valor como necesite el proceso.
  La lista de parámetros depende del número de variables definidas en la pantalla de variables de pipeline de Data Integration. Parámetros de pipeline predeterminados y valores aceptables:

- `STARTPERIOD` El primer periodo durante el que se deben cargar los datos. Este nombre de período debe estar definido en la asignación de períodos de Data Integration.
- `ENDPERIOD` El último periodo durante el que se deben cargar los datos. Este nombre de período debe estar definido en la asignación de períodos de Data Integration.
- `IMPORTMODE` determina cómo se importan los datos en Data Integration. Los valores aceptables son los siguientes:
  * `Append` agrega los datos de PDV existentes a Data Integration.
  * `Replace` Suprime los datos del PDV y los reemplaza por los datos del archivo especificados en la definición del pipeline o como un parámetro.
  * `Map and Validate` Vuelve a procesar los datos con asignaciones actualizadas y cuentas lógicas sin importar los datos del archivo especificados en la definición del pipeline o como un parámetro.
  * `No Import` Omite la importación de datos en la tabla temporal de Data Integration.
- `EXPORTMODE` determina cómo se exportan los datos en Data Integration. Los modos de exportación aceptables son
  * `Merge` Fusiona los datos en la tabla temporal de Data Integration con los datos existentes en el proceso de negocio.
  * `Accumulate` Agrega los datos de la tabla temporal de Data Integration en el proceso de negocio.
    Este modo de exportación se aplica solo a Planning, Planning Modules, FreeForm, Sales Planning y Strategic Workforce Planning.
  * `Replace` Borra los datos de PDV en el proceso de negocio y los reemplaza por los datos de la tabla temporal de Data Integration. Se borran los datos de las dimensiones Scenario, Version, Year, Period y Entity.
  * `No Export` Omite la exportación de datos de Data Integration en el proceso de negocio.
- `ATTACH_LOGS` especifica si se van a comprimir y adjuntar los archivos de registro en las notificaciones relacionadas con la ejecución del pipeline. Los valores aceptables son `Y` para Sí y `N` para No.
- `SEND_MAIL` especifica si se va a enviar un correo electrónico al finalizar la ejecución del pipeline. Los valores aceptables son los siguientes:: `Always`, `No` (valor predeterminado), `On Failure` y `On Success`.
- `SEND_TO` es una lista separada por comas de ID de correo electrónico a la que se enviará el correo electrónico.

Ejemplo

```
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

Copia los datos de una base de datos de almacenamiento de bloques en una base de datos de almacenamiento agregado o de un almacenamiento de bloques en otro almacenamiento de bloques según la configuración especificada en un trabajo de tipo `plan type map`.

Se aplica a

Planning, Planning Modules, FreeForm, Sales Planning y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate runPlanTypeMap` *`JOB_NAME`* `[clearData=true|false]` donde:

- `NOMBRE_TRABAJO` es el nombre de un trabajo de tipo `plan type map` definido en la aplicación.
- `clearData` es una configuración opcional que indica si los datos de la base de datos destino se deben eliminar antes de copiar los datos. Si no se establece este valor de parámetro, se utiliza el valor predeterminado `true`.

  Los valores de parámetros (`true` o `false`) deben ir en minúscula.

Ejemplo

```
epmautomate runPlanTypeMap CampaignToReporting clearData=false
```

runRuleSet

Inicia un conjunto de reglas de negocio.

Se aplica a

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

Roles necesarios

Administrador del servicio, Usuario avanzado (si se ha otorgado el acceso de inicio de la regla)

Sintaxis

`epmautomate runRuleSet` *`RULESET_NAME`* `[`*`PARAMETER=VALUE`*`]` donde:

- *`RULESET_NAME`* es el nombre de un conjunto de reglas de negocio exactamente como se define en el entorno.
- *`PARAMETER=VALUE`* indica los parámetros de tiempo de ejecución opcionales y sus valores necesarios para ejecutar el conjunto de reglas.

> **Nota:**
>
> El conjunto de reglas se ejecuta en el tipo de plan en el que se despliega.
>
> Utilice el par *PARÁMETRO=VALOR* para especificar tantas peticiones de datos en tiempo de ejecución como conjuntos de reglas se necesiten. En el siguiente ejemplo se utilizan dos peticiones de datos en tiempo de ejecución (`Period` y `Entity`) y sus valores (`Q1` y `USA`).
>
> Se utilizan los valores predeterminados si no se proporcionan valores para los parámetros de tiempo de ejecución. El comando ignora las peticiones de datos en tiempo de ejecución que no coincidan exactamente con las definidas en el conjunto de reglas.
>
> Consulte Especificación de varios valores para un parámetro para obtener información sobre cómo introducir varios valores para un parámetro.

Ejemplo

```
epmautomate runRuleSet RollupUSSales Period=Q1 Entity=USA
```

runSupplementalDataReport

Genera informes relacionales que muestran datos del Gestor de datos adicionales.

Los informes de datos adicionales se agrupan como Informes de no consolidación en Financial Consolidation and Close y Tax Reporting. Consulte la sección sobre la lista de informes y parámetros predefinidos de Generación de informes para Financial Consolidation and Close y Tax Reporting en la guía *API de REST* para obtener una lista de los informes que puede generar y los parámetros que necesita para generarlos.

Se aplica a

Financial Consolidation and Close, y Tax Reporting.

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Sintaxis

`epmautomate runSupplementalDataReport` *`FILE_NAME GROUP_NAME REPORT_NAME REPORT_FORMAT`* `[Param=value]` donde:

- `FILE_NAME` es un nombre de archivo único para el informe.
- `GROUP_NAME` es el nombre del grupo con el que está asociado el informe.
- `REPORT_NAME` es un nombre único para el informe que se va a generar.
- `REPORT_FORMAT` es uno de los siguientes formatos para el informe:
  - PDF
  - HTML (no está soportado para gráficos)

  - XLSX (no está soportado para gráficos)
  - CSV
  - CSV2

  REPORT_FORMAT `CSV` no permite el formato de los datos en función de una plantilla, mientras que `CSV2` sí lo hace. Generar informes con formato `CSV2` es un proceso más lento que si se genera una salida `CSV`.

- `Param=value`, de forma opcional, identifica los parámetros necesarios para generar el informe. Por ejemplo, para generar el informe `At Risk Tasks`, que toma un valor `schedule name` con el valor `monthly` y un valor `period` con el valor `Jan`, especifique `"schedule name"=monthly period=Jan`.

Ejemplo

```
epmautomate runSupplementalDataReport Example_File_name Group1 "At Risk Tasks"
html "schedule name"=monthly period=Jan
```

runTaskManagerReport

Genera informes relacionales que muestran datos del Gestor de tareas.

Se aplica a

Enterprise Profitability and Cost Management, Financial Consolidation and Close y Tax Reporting.

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Sintaxis

`epmautomate runTaskManagerReport` *`FILE_NAME GROUP_NAME REPORT_NAME REPORT_FORMAT`* `[Param=value]` donde:

- `FILE_NAME` es un nombre de archivo único para el informe.
- `GROUP_NAME` es el nombre del grupo con el que está asociado el informe.
- `REPORT_NAME` es un nombre único para el informe que se va a generar.
- `REPORT_FORMAT` es uno de los siguientes formatos para el informe:
  - PDF
  - HTML (no está soportado para gráficos)
  - XLSX (no está soportado para gráficos)
  - CSV
  - CSV2

> **Nota:**
>
> *REPORT_FORMAT* CSV no permite el formato de los datos en función de una plantilla, mientras que CSV2 sí lo hace. Generar informes con formato CSV2 es un proceso más lento que si se genera una salida CSV.

- Param=value, de forma opcional, identifica los parámetros necesarios para generar el informe. Por ejemplo, para generar el informe Early Tasks, que toma un valor schedule name con el valor monthly y un valor period con el valor Jan, especifique "schedule name"=monthly period=Jan.

Ejemplo

```
epmautomate runTaskManagerReport Example_File_name Group1 "Early Tasks" PDF
"schedule name"=monthly period=Jan
```

sendMail

Envía un correo electrónico, con la opción de adjuntar archivos de Oracle Fusion Cloud Enterprise Performance Management.

Puede incorporar este comando en scripts para notificar a los usuarios sobre los distintos estados o para enviarles informes.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

epmautomate sendMail ToAddress Subject [Body="MessageBody"] [Attachments=*FILE1,FILE2*], donde:

- ToAddress identifica las direcciones de correo electrónico de los destinatarios separadas por puntos y coma con comillas dobles. Ejemplo, "jdoe@example.com;jane.doe@example.com".
- Subject identifica el asunto del correo electrónico.
- Body="MessageBody", opcionalmente, es el contenido del correo electrónico. Si no se especifica, el correo electrónico no tiene cuerpo.

> **Note:**
>
> Use etiquetas HTML válidas para aplicar formato al cuerpo del mensaje y crear el formato de correo electrónico deseado. El cuerpo del mensaje entero (incluidas las etiquetas HTML) debe especificarse en una línea y no debe contener caracteres de nueva línea. Consulte Ejemplos.

- `Attachments`, opcionalmente, identifica una lista de archivos separadas por comas disponible en Cloud EPM que adjuntar al correo electrónico. Por ejemplo, `outbox/errorFile.txt,inbox/users.csv`.

> **Note:**
>
> – Use * (asterisco) como comodín para un carácter en el nombre de archivo. Por ejemplo, especifique `outbox/user*.csv` para adjuntar todos los archivos del buzón de salida con nombres de archivo de cinco letras que se ajusten al patrón.
>
> – Puede adjuntar cualquier archivo, que no sean instantáneas, de los enumerados por el comando listFiles, como documento adjunto a un correo electrónico. El tamaño del documento adjunto no debe superar los 10 MB.

Ejemplos

- **Correo electrónico sin formato:** `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`
- **Correo electrónico con formato:** `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

Pone la aplicación en modo de administración, de forma que se limita el acceso a la aplicación solo a administradores de servicio.

Este comando es útil para evitar que los usuarios trabajen en la aplicación cuando los administradores de servicio realizan operaciones administrativas. La aplicación permanece en el modo de administración hasta que se cambie para que todos los usuarios puedan acceder.

> **Note:**
>
> Este comando sustituye al comando applicationAdminMode, que está anticuado, pero no se ha eliminado de EPM Automate.

Utilice el comando getApplicationAdminMode para comprobar el estado actual del entorno.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

```
epmautomate setApplicationAdminMode true|false
```

En este comando, especifique `true` para poner la aplicación en modo de administración y `false` para devolverla al modo normal, para que todos los usuarios puedan acceder a ella.

Ejemplos

- Ponga la aplicación en modo de administración:
  `epmautomate setApplicationAdminMode true`
- Devuelva la aplicación al modo normal:
  `epmautomate setApplicationAdminMode false`

setDailyMaintenanceStartTime

Establece la hora (UTC u otra zona horaria) a la que se inicia el mantenimiento diario del entorno. El mantenimiento del entorno no tiene por qué empezar a la hora en punto; puede establecer la hora y el minuto en los que debe comenzar el mantenimiento.

Para garantizar que el uso de este comando no interfiera con el requisito de Oracle de crear copias de seguridad, este comando no cambiará la hora de inicio del mantenimiento si el proceso de mantenimiento diario no se ha realizado en las últimas 36 horas.

> **Nota:**
>
> Los administradores del servicio que están actualmente conectados al entorno usando un navegador solo verán la nueva hora de inicio del mantenimiento diario después de cerrar sesión y volverla a iniciar.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio, usuario (solo Oracle Enterprise Data Management Cloud), cualquier rol predefinido y el rol de aplicación Migraciones - Administrar

Sintaxis

`epmautomate setDailyMaintenanceStartTime` *`StartTime`* donde *`StartTime`* es la hora (con formato HH:MM de 24 horas) a la que debe empezar el proceso de mantenimiento y una zona horaria opcional. El rango de valor temporal aceptable oscila entre 00:00 y 23:59. Si la hora de inicio no se va a establecer en UTC, especifique una zona horaria estándar válida; por ejemplo, `"14:35 America/Los_Angeles"` para la hora oficial del Pacífico 2:35 p.m.

Ejemplos

- Establezca el inicio del mantenimiento diario en 2:20 p.m. UTC:
  `epmautomate setDailyMaintenanceStartTime 14:20`

- Establezca el inicio del mantenimiento diario en 2:35 p.m. hora oficial del Pacífico:
  `epmautomate setDailyMaintenanceStartTime "14:35 America/Los Angeles"`

setDemoDates

Actualiza los datos de demostración internos de Oracle según sea necesario.

Utilice este comando solo en la configuración de instalaciones con datos de la demostración interna de Oracle.

SoloAccount Reconciliation: este comando restablece las fechas de todas las conciliaciones que tienen asociados los valores del atributo `Demo Code setdemodates` o `setdemodatesnostatuschange`. Este comando gestiona conciliaciones durante un máximo de 12 períodos: un período actual y 11 períodos anteriores (históricos). Si las conciliaciones de más de dos períodos se etiquetan con el atributo `Demo Code`, el comando trata estos periodos como si pertenecieran al periodo anterior. Las conciliaciones que no tienen este valor de atributo no se ven afectadas.

- So el valor es `setdemodates`, el comando restablece las fechas de conciliación en función de la fecha especificada y un estado aleatorio
- Si el valor es `setdemodatesnostatuschange`, el comando restablece las fechas de conciliación en función de la fecha especificada sin cambiar el estado de conciliación

Todos los demás procesos de negocio: este comando restablece la fecha de inicio y finalización de las tareas y otros datos referentes a fechas para que las tareas sean adecuadas para una demostración. Calcula las nuevas fechas de tarea basándose en el valor del conjunto de atributos `SETDEMODATES` de la programación de tareas junto con el valor `Demo Date` proporcionado. Si no se especifica el valor `Demo Date`, el comando utiliza la fecha del día para calcular las nuevas fechas de tareas.

> **Nota:**
>
> Las tareas de programas que no tienen el valor `SETDEMODATES` no se ven afectadas.

En función de la `fecha de demostración` que especifique, este comando adelanta todas las fechas asociadas a la tarea. Esto incluye las fechas principales de tiempo de ejecución (fecha de inicio, fecha de finalización, etc.) y fechas auxiliares, incluidas fechas del historial, fechas de vencimiento de flujo de trabajo individual y fechas de inicio (reales). El estado de la tarea no se ve afectado.

Se aplica a

Account Reconciliation, Enterprise Profitability and Cost Management, Financial Consolidation and Close, FreeForm, Planning, Planning Modules, Sales Planning, Strategic Workforce Planning y Tax Reporting

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

`epmautomate setDemoDates [DEMO_DATE]`, donde *DEMO_DATE* es una fecha opcional con el formato `YYYY-MM-DD`. Las conciliaciones se restablecen a la fecha actual si no especifica este valor.

Ejemplo

```
epmautomate setDemoDates 2020-02-15
```

setEJJournalStatus

En Financial Consolidation and Close, establece el resultado de la contabilización de Enterprise Journal del sistema ERP. Use este comando para actualizar el estado de contabilización de los asientos que tengan el estado `Post in Progress` con independencia del estado de su flujo de trabajo.

Este comando usa un archivo CSV que identifica el estado de la importación en el sistema ERP. Use el comando uploadFile para cargar el archivo de importación en el entorno. El formato del archivo CSV es el siguiente:

```
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"
```

La columna de mensaje es opcional y se puede omitir.

Con este comando no se exportan datos de Enterprise Journal de Financial Consolidation and Close ni se importan en el sistema ERP.

Se aplica a

Financial Consolidation and Close

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate setEJJournalStatus FILE_NAME.csv [OPERATION=posting|validation]`, donde:

- *FILE_NAME* identifica el archivo CSV que contiene el estado de la importación en el sistema ERP.
- `Operation`, opcionalmente, es la operación que se va a realizar. Los valores posibles son `posting` y `validation`. El valor predeterminado es `posting` `operation=posting` proporciona un archivo CSV que contiene los resultados de la contabilización de asientos con estas columnas: Year, Period, Journal ID, Posting Status

(Posted/Failed), Message, Error Code y Error Message. Las columnas Message, Error Code y Error Message son opcionales. El estado de contabilización de los asientos se actualiza en función del estado proporcionado en el archivo. Solo se actualizarán los asientos que tengan un estado de contabilización *Contabilización en curso* .

`operation = validation` genera un archivo CSV que contiene los resultados de la validación de asientos con estas columnas: Year, Period, Journal ID, Validation Status (Valid/Failed), Message, Error Code y Error Message. Las columnas Message, Error Code y Error Message son opcionales. El estado de validación de los asientos se actualiza en función del estado proporcionado en el archivo. Solo se actualizarán los asientos que tengan un estado de validación *Validación en curso* .

Ejemplo

- Establezca los resultados de contabilidad de Enterprise Journals a partir de un archivo:

  ```
  epmautomate setEJJournalStatus JournalStatus.csv
  ```

- Establezca los resultados de contabilidad de Enterprise Journals a partir de un archivo y realice la operación de validación:
  ```
  epmautomate esetEJJournalStatus JournalStatus.csv Operation=validation
  ```

setEncryptionKey

Establece una clave de cifrado personalizada para el acceso a la base de datos.

El uso de este comando proporciona una solución Traiga su propia clave (BYOK) para que los clientes incluyan Oracle Fusion Cloud Enterprise Performance Management en su rotación de gestión de claves estándar.

La clave de cifrado personalizada se aplica después del siguiente mantenimiento diario del entorno. Puede activarla inmediatamente ejecutando el comando resetService.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate setEncryptionKey key=`*`key`* donde *`key`* es una cadena personalizada de cualquier longitud que desea utilizar como clave de cifrado.

Ejemplos

- Establecer la clave de cifrado: `epmautomate setEncryptionKey key=se!m+a2J`
- Eliminar la clave de cifrado: `epmautomate setEncryptionKey key=`

setEssbaseQryGovExecTime

Define la cantidad máxima de tiempo, en segundos, que puede utilizar una consulta de Oracle Essbase para recuperar y entregar información antes de la terminación de la consulta.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate setEssbaseQryGovExecTime` *`TIME`* donde `TIME` identifica el número de segundos tras los cuales se terminarán las consultas de Essbase. Este valor debe ser un número entero que no sea superior a 70000.

Oracle recomienda que no defina este valor en 0 (cero) para evitar que las consultas de Essbase se ejecuten de forma indefinida.

Ejemplo

```
epmautomate setEssbaseQryGovExecTime 600
```

setIdleSessionTimeout

Cambia el tiempo de espera de sesión (en minutos) del entorno de Oracle Fusion Cloud Enterprise Performance Management. El nuevo tiempo de espera de sesión cambia a activo después del siguiente mantenimiento diario del entorno. Utilice este comando para cambiar el tiempo de espera de sesión predeterminado (75 minutos) a un valor diferente. Después de que una sesión esté inactiva durante el tiempo especificado mediante este comando, se redirecciona al usuario a la página de inicio de sesión.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate setIdleSessionTimeout` *`MINUTES`*, donde MINUTES identifica el nuevo tiempo de espera de sesión inactiva (mínimo de 15 y máximo de 150) en minutos.

Ejemplo

```
epmautomate setIdleSessionTimeout 30
```

setIPAllowlist

Configura una lista de direcciones IP permitidas y Classless Inter-Domain Routing (CIDR) que pueden acceder a Oracle Fusion Cloud Enterprise Performance Management. Este comando agrega o elimina direcciones IPv4 y CIDR.

Este comando proporciona un método de autoservicio para configurar una lista de permitidos para entornos de Cloud EPM.

> **Note:**
>
> - Al configurar la lista de direcciones IP permitidas para un entorno de Cloud EPM, permite conexiones solo desde esas direcciones IP específicas. En este escenario, solicitar el acceso desde otro entorno de Cloud EPM no funcionará, a menos que agregue las direcciones IP salientes del centro de datos o de la región en los que se encuentra el entorno solicitante a la lista de direcciones IP permitidas. Consulte Direcciones IP de salida de los centros de datos y las regiones de Cloud EPM en la *Guía de operaciones* para las direcciones IP que necesita agregar para asegurar que otros entornos se puedan comunicar con el entorno para el que se está configurando la lista de direcciones IP permitidas.
> - Si está utilizando el perímetro de red de IDCS en Identity Cloud Service en el que se encuentra el entorno, debe agregar las mismas direcciones IP en este perímetro de red. Puede optar por agregar las direcciones IP solo en el perímetro de red en lugar de configurar una lista de permitidos para los distintos entornos.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmAutomate setIPAllowlist add|remove` *`FILE_NAME`*`.txt`, donde:

- `add` agrega las direcciones IP y CIDR que se muestran en un archivo de texto a la lista de permitidos.
- `remove` suprime las direcciones IP y CIDR que se muestran en un archivo de texto de la lista de permitidos.
- *`FILE_NAME`* es el nombre de un archivo de texto que muestra las direcciones IP y CIDR que se van a agregar a la lista de permitidos o eliminarse de esta. Cada entrada del archivo debe ir separada por un carácter de nueva línea. Use el comando uploadFile para cargar

este archivo en el entorno. Cada línea del archivo debe ser una dirección IPv4 o un CIDR con el siguiente formato:

```
xxx.xxx.xxx.xxx
xxx.xxx.xxx.xxx/n
```

> **Note:**
>
> – Solo están soportadas las direcciones IP IPv4.
>
> – Utilice el formato CIDR en lugar de direcciones IP individuales para especificar un rango continuo de direcciones IP.
>
> – Para hacer que una lista de permitidos no permita el acceso desde cualquier dirección IP, use el comando getIPAllowlist para escribir todas las direcciones IP y CIDR existentes en un archivo. Cargue el archivo en el entorno y, a continuación, ejecute este comando con la opción `remove` como se muestra en este ejemplo:
>
> ```
> epmAutomate getIPAllowlist > myRemoveList.txt
> epmAutomate uploadFile myRemoveList.txt
> epmAutomate setIPAllowlist remove myRemoveList.txt
> ```

Ejemplos

- Adición de algunas direcciones IP y CIDR a una lista de permitidos:

  ```
  epmAutomate setIPAllowlist add myAddList.txt
  ```

- Eliminación de algunas direcciones IP de una lista de permitidos:

  ```
  epmAutomate setIPAllowlist remove myRemoveList1.txt
  ```

setManualDataAccess

Especifica si Oracle puede acceder manualmente a la bases de datos relacionales y de Oracle Essbase de un entorno en situaciones de emergencia cuando un entorno no responda y el cliente aún no haya proporcionado una solicitud de servicio para investigar y hacer que el entorno esté disponible.

En una situación de emergencia, Oracle recurre a un proceso interno donde un ejecutivo de desarrollo de alto nivel, tras un proceso de verificación independiente, permite un acceso manual a las bases de datos relacionales y de Essbase. Utilice este comando para impedir que Oracle acceda a estas bases de datos sin su aprobación explícita. Además, puede prohibir a Oracle que acceda manualmente a las bases de datos relacionales y de Essbase en situaciones de emergencia incluso si hay abierta una solicitud de servicio.

El valor que especifique con este comando se aplicará de forma inmediata. De forma predeterminada, se permite el acceso manual a los datos mediante Oracle. Utilice este comando para revocar este acceso.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate setManualDataAccess Allow|Revoke [disableEmergencyAccess=true|false]`, donde `disableEmergencyAccess`, opcionalmente, especifica si desea prohibir todo el acceso manual a las bases de datos relacionales y de Essbase. Al establecer este valor de propiedad como `true`, Oracle dejará de acceder manualmente a estas bases de datos, incluso aunque haya una solicitud de servicio abierta. El valor predeterminado es `false`.
Si especifica el valor `disableEmergencyAccess` como `true`, Oracle no podrá acceder a las bases de datos relacionales ni de Essbase si se necesita acceso para solucionar y corregir un entorno caído. Si el entorno está caído, no podrá emitir este comando para permitir que Oracle acceda manualmente a las bases de datos.

Ejemplos

- Revocar el permiso que se otorgó para acceder manualmente a las bases de datos relacionales y de Essbase en situaciones de emergencia sin aprobación explícita:
  `epmautomate setManualDataAccess revoke`
- Permitir el acceso manual a las bases de datos relacionales y de Essbase en emergencias:
  `epmautomate setManualDataAccess allow`
- Prohibir el acceso manual a las bases de datos relacionales y de Essbase, incluso aunque esté abierta una solicitud de servicio:
  `epmautomate setManualDataAccess revoke disableEmergencyAccess=true`

setPeriodStatus

Establece un estado específico en un período.

Se aplica a

Account Reconciliation

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Los usuarios con los roles predefinidos Usuario avanzado, Usuario y Visor puede que necesiten más roles de aplicación.

Sintaxis

`epmautomate setPeriodStatus` *`PERIOD STATUS`* donde:

- *`PERIODO`* es el nombre de un periodo
- *`ESTADO`* es `OPEN`, `CLOSED` o `LOCKED`

Ejemplo

```
epmautomate setPeriodStatus "January 2015" OPEN
```

setRestrictedDataAccess

Configura el entorno de Oracle Fusion Cloud Enterprise Performance Management para impedir que los administradores del servicio consientan el envío de una instantánea de aplicación a Oracle al usar la utilidad Proporcionar comentarios.

Si tiene una política que no permite que Oracle acceda a sus datos, utilice este comando para gestionar el uso compartido de datos al enviar comentarios a Oracle. Al configurar el acceso a datos restringidos para el entorno, se desactiva la opción Enviar instantánea de aplicación que muestra la utilidad Proporcionar comentarios.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

```
epmautomate setRestrictedDataAccess true|false
```

Para ver el estado actual de las restricciones de acceso a los datos, utilice getRestrictedDataAccess.

Ejemplos

- Impida a los administradores del servicio consentir el envío de una instantánea de aplicación:
  ```
  epmautomate setRestrictedDataAccess true
  ```
- Permita a los administradores del servicio consentir el envío de una instantánea de aplicación:
  ```
  epmautomate setRestrictedDataAccess false
  ```

setSubstVars

Crea o actualiza variables de sustitución en el nivel de aplicación o de cubo.

No puede utilizar este comando para establecer múltiples valores y/o funciones para variables de sustitución.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate setSubstVars` *`CUBE_NAME SUBSTVAR=VALUE [SUBSTVAR=VALUE]`* donde:

- *`CUBE_NAME`* es el cubo (por ejemplo, Plan1 o Plan2) para el que se crea o se actualiza la variable de sustitución. Utilice `All` en lugar de un nombre de cubo para definir o actualizar la variable de sustitución en el nivel de aplicación.
- *`SUBSTVAR`* es el nombre de la variable de sustitución para la que se va a definir o actualizar un valor.
- *`VALUE`* es el nuevo valor de la variable de sustitución.

Ejemplos

- Crear o actualizar una variable de sustitución en el nivel de aplicación: `epmautomate setSubstVars ALL CurYear=2015 CurPeriod=Jan`
- Crear o actualizar variables de sustitución en el nivel de cubo: `epmautomate setSubstVars Plan2 CurYear=2013 CurPeriod=Jan`

setVirusScanOnFileUploads

Permite al entorno de OCI (Gen 2) examinar los archivos en busca de virus antes de que se carguen en Oracle Fusion Cloud Enterprise Performance Management.

Todos los entornos de OCI (Gen 2) están protegidos con un programa antivirus. Este comando proporciona seguridad adicional, ya que le permite activar la exploración de virus al realizar la carga de archivos. La exploración de archivos antes de su carga impide la posibilidad de que se carguen virus en el entorno.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate setVirusScanOnFileUploads true|false`
De forma predeterminada, la exploración de virus no está activada (está definida en `false`). Cuando este valor está definido en `true`, Cloud EPM examina todos los archivos de carga. Si un archivo está infectado con virus, no se carga en el entorno.

Ejemplos

- Activar la exploración de virus: `epmautomate setVirusScanOnFileUploads true`
- Desactivar la exploración de virus: `epmautomate setVirusScanOnFileUploads false`

simulateConcurrentUsage

Ejecuta diferentes operaciones simultáneas en un entorno mediante la simulación de usuarios.

Este comando se puede utilizar para validar el rendimiento del entorno a fin de verificar que el tiempo de respuesta es aceptable cuando el servicio está bajo carga durante determinadas operaciones ejecutadas por un número específico de usuarios. Por ejemplo, este comando se puede utilizar para medir el rendimiento si 50 usuarios abren simultáneamente un formulario utilizando diferentes PDV. Permite la prueba de carga de autoservicio de los entornos.

Este comando realiza la simulación ejecutando las operaciones especificadas para un número determinado de usuarios e iteraciones. Ejecuta varias iteraciones para calcular el tiempo mínimo, el tiempo máximo y el tiempo medio de una operación en particular. Soporta estas operaciones para realizar una prueba de carga de uso simultáneo:

- Abrir formularios
- Guardar formularios
- Ejecutar reglas de negocio
- Ejecutar conjuntos de reglas de negocio
- Ejecutar reglas de datos
- Abrir cuadrículas ad hoc
- Ejecutar informes
- Ejecutar libros

> **Note:**
>
> Este comando no soporta los informes y los libros de Financial Reporting; solo soporta los libros y los informes que pertenecen a Informes (anteriormente Informes de gestión).

> **Caution:**
>
> Este comando ejecuta las operaciones especificadas en el entorno actual y puede, en función de la operación, actualizar los datos en el entorno. Ejecute este comando en entornos de prueba. No se recomienda su ejecución en entornos de producción.

Este comando acepta un archivo ZIP que se debe haber cargado en la bandeja de entrada del entorno como entrada. Este archivo ZIP contiene un archivo `requirement.csv` y los archivos de entrada que soportan los casos de uso incluidos en `requirement.csv`. Opcionalmente, el archivo ZIP puede contener un archivo `userVarMemberMapping.csv` para proporcionar la asignación de miembros de variable de usuario, un archivo `options.xml` para proporcionar opciones de Oracle Smart View for Office para algunos casos de uso y un archivo `users.csv` para proporcionar nombres de usuario y contraseñas de los usuarios existentes para su uso

en lugar de crear nuevos usuarios. Este comando simula a continuación los casos de uso y crea un informe que se puede enviar por correo electrónico a uno o más destinatarios.

> **Note:**
>
> Este comando no genera envíos de Proporcionar comentarios. Puede utilizar la opción Proporcionar comentarios (consulte Contribución a la recopilación de información de diagnóstico de Oracle mediante la utilidad de proporción de comentarios en *Introducción a Oracle Enterprise Performance Management Cloud para administradores*) de una pantalla de proceso de negocio, la API de REST Proporcionar comentarios o el comando feedback para generar un envío de Proporcionar comentarios para obtener detalles del entorno después de ejecutar la simulación.

Escenario de uso 1: prueba de aceptación del rendimiento de aplicación para 50 usuarios que abren un formulario de forma simultánea.

Solución:

1. Cree `requirement.csv` con entradas similares a las siguientes, suponiendo que desea abrir un formulario denominado `Exchange Rates` almacenado en la carpeta `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. Cree `open_form_input.csv` utilizando el formato especificado en Archivo de entrada de apertura de formulario. Tendrá una entrada en este archivo que se utilizará 50 veces. Si desea abrir el mismo formulario con diferentes PDV, tendrá tantas entradas como el número de PDV que desee utilizar.
3. Cree `userVarMemberMapping.csv` con el formato especificado en Creación del archivo UserVarMemberMapping.csv si es necesario establecer la asignación de miembros de variable de usuario.
4. Exporte las opciones de Smart View a `options.xml`, si es necesario utilizar las opciones de Smart View. Consulte Creación del archivo options.xml para obtener más información.
5. Cree un archivo ZIP que contenga los archivos de los pasos anteriores y cárguelo en la bandeja de entrada.
6. Ejecute el comando `simulateConcurrentUsage` utilizando el archivo ZIP del paso anterior como archivo de entrada.

Escenario de uso 2: simulación del rendimiento para un aumento de uso estacional; por ejemplo, al final del año fiscal. Suposición: 100 usuarios tienen un formulario con un tiempo de retraso de seis segundos entre cada usuario.

Solución:

1. Cree `requirement.csv` con entradas similares a las siguientes, suponiendo que desea guardar un formulario denominado `Accessories Revenue` almacenado en la carpeta `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. Cree `save_form_input.csv` utilizando el formato especificado en Archivo de entrada de almacenamiento de formulario.
3. Cree `userVarMemberMapping.csv` con el formato especificado en Creación del archivo UserVarMemberMapping.csv si es necesario establecer la asignación de miembros de variable de usuario.
4. Exporte las opciones de Smart View a `options.xml`, si es necesario utilizar las opciones de Smart View. Consulte Creación del archivo options.xml para obtener más información.
5. Cree un archivo ZIP que contenga los archivos de los pasos anteriores y cárguelo en la bandeja de entrada.
6. Ejecute el comando `simulateConcurrentUsage` utilizando el archivo ZIP del paso anterior como archivo de entrada y estos valores de propiedad, `iteration=1` y`lagTime=6`.

Se aplica a

Planning, Planning Modules, FreeForm, Financial Consolidation and Close, Tax Reporting, Strategic Workforce Planning y Sales Planning.

Roles necesarios

Administrador del servicio. Necesita también el rol Administrador de dominio de identidad para utilizar testModes 0, 1 y 2.

Sintaxis

`epmautomate simulateConcurrentUsage` *`INPUT_FILE.zip`* `[iterations=`*`COUNT`*`] [notificationEmails="EMAIL_ADDRESS"] [testMode=0|1|2|3|4] [lagTime=`*`LAG_TIME`*`]`, donde:

- *`INPUT_FILE.zip`* es el nombre del archivo ZIP que identifica sus casos de uso. Utilice el comando uploadFile (ejemplo de sintaxis de comandos `epmautomate uploadFile "C:/uploads/`*`INPUT_FILE.zip`*`" inbox`) para cargar este archivo en la bandeja de entrada antes de ejecutar este comando. Este archivo ZIP debe contenes los siguientes archivos:
  - Un archivo CSV de casos de uso denominado `requirement.csv`. Cada línea de este archivo CSV identifica el tipo de operación que se debe realizar, el nombre del artefacto, el número de usuarios simultáneos, el archivo de entrada que especifica los detalles de la operación e información adicional relacionada con cada caso de uso. Consulte Creación del archivo requirement.csv.
  - Los archivos de entrada que contienen los detalles de cada operación. Consulte estos temas:
    - Archivo de entrada de apertura de formulario
    - Archivo de entrada de almacenamiento de formulario
    - Archivo de entrada de ejecución de regla de negocio
    - Archivo de entrada de ejecución de conjunto de reglas de negocio

    * Archivo de entrada de ejecución de regla de datos
    * Archivo de entrada de cuadrícula ad hoc
    * Archivo de entrada de ejecución de libro
    * Archivo de entrada de ejecución de informe
  - Opcionalmente, agregue `userVarMemberMapping.csv` al archivo ZIP de entrada para proporcionar la asignación de miembros de variable de usuario. Consulte Creación del archivo UserVarMemberMapping.csv.
  - Opcionalmente, agregue `options.xml` al archivo ZIP de entrada para utilizar las opciones de Smart View. Consulte Creación del archivo options.xml.
  - Opcionalmente, agregue `users.csv` al archivo ZIP de entrada para proporcionar los nombres de usuario y las contraseñas de los usuarios existentes. Consulte Creación del archivo users.csv.
- `iterations` es un número positivo que indica el número de veces que se ejecutará cada caso de uso identificado en `requirement.csv` para medir el tiempo de respuesta. Si no se especifica, la operación solo se ejecuta una vez.
- De manera opcional, `notificationEmails` indica las direcciones de correo electrónico a las que se enviarán los resultados de estos comandos. Si se especifica más de una dirección de correo electrónico, utilice un punto y coma para separarlas. Incluya también la lista de direcciones entre comillas dobles. Si no se especifica, los resultados se envían por correo electrónico al usuario que inició el comando. Para obtener más información detallada sobre este informe, consulte Informe Simulación de uso simultáneo de ejemplo.
- De manera opcional, `[testMode]` especifica el modo de simulación de uso simultáneo. El valor predeterminado es 0. Los valores aceptables son los siguientes:
  - `0`: el modo de simulación predeterminado, que agrega usuarios simulados al entorno y les asigna el rol Administrador del servicio, ejecuta la simulación y, a continuación, suprime los usuarios simulados. Este modo resulta útil si desea ejecutar la prueba solo una vez.
    Los usuarios simulados tienen las siguientes propiedades:

    Nombre: testuser1, testuser2, etc.

    Apellido: testuser1, testuser2, etc.

    Dirección de correo electrónico: testuser1@discard.oracle.com, testuser2@discard.oracle.com, etc.

    Nombre de usuario: testuser1, testuser2, etc.
  - `1`: agregar usuarios simulados al entorno y les asigna el rol Administrador del servicio. No ejecuta la simulación ni suprime los usuarios simulados.
    Después de utilizar este modo, ejecute el comando con el modo 3 para ejecutar la simulación tantas veces como sea necesario. Al final, ejecute el comando con el modo 2 para suprimir los usuarios simulados.
  - `2`: suprime los usuarios simulados. No crea usuarios ni ejecuta la simulación.
  - `3`: ejecuta la simulación utilizando los usuarios simulados existentes sin agregar ni suprimir usuarios.
  - `4`: utiliza los usuarios definidos en el archivo `users.csv` incluidos en el archivo ZIP de entrada para ejecutar el comando. Consulte Creación del archivo users.csv. En este modo no se crean usuarios para la simulación. En su lugar, utiliza los usuarios existentes.

Si desea ejecutar el uso simultáneo una vez, utilice `testMode=0`. Para ejecutar una serie de pruebas:

- En primer lugar, ejecute el comando utilizando `testMode=1` para agregar los usuarios simulados y asignarles el rol Administrador del servicio.
- A continuación, ejecute el comando utilizando `testMode=3` para ejecutar la simulación tantas veces como sea necesario.
- Por último, ejecute el comando utilizando `testMode=2` para suprimir los usuarios simulados.

- De manera opcional, `[lagTime]` especifica el número de segundos (5 segundos o más) que debe esperar el comando entre la ejecución de cada caso de uso en `requirement.csv`. El valor predeterminado es 5 segundos. No utilice números negativos (por ejemplo -1), fracciones (por ejemplo, 1/2) ni valores decimales.
  Después de que una usuario haya iniciado la ejecución de un caso de uso de `requirement.csv`, el comando espera el número de segundos especificado en este parámetro para que el siguiente usuario inicie la ejecución del caso de uso. Debido a que las actividades de usuario normalmente no se inician de manera simultánea, establecer este parámetro ayuda a crear una simulación de carga más realista en un entorno.

Ejemplo

```
epmautomate simulateConcurrentUsage test_simulation.zip iterations=5
notificationEmails="jane.doe@example.com;john.doe@example.com;example@example.
com" lagTime=6
```

skipUpdate

Solicita que Oracle omita la aplicación de actualizaciones mensuales en un entorno durante un máximo de tres ciclos consecutivos, o elimina todas las solicitudes de omisión de actualización realizadas previamente mediante este comando para que el entorno se actualice a la línea de código principal.

También puede utilizar este comando para mostrar las solicitudes de omisión de actualización actualmente especificadas para un entorno. El estado de actualización omitida del entorno se incluye en el Informe de actividad (en las métricas operativas) que se genera después de que haya utilizado este comando para omitir actualizaciones en un entorno. Consulte Métricas operativas en *Guía de introducción para administradores*.

Los parches semanales y de emergencia para el mes actual, de haberlos, se seguirán aplicando al entorno. No se realizarán actualizaciones durante los meses para los que se solicite el retraso de la actualización.

No es posible omitir actualizaciones para un entorno que está en un parche puntual. Además, no se pueden omitir actualizaciones mensuales que estén separadas más de dos meses de la actualización en la que está el entorno actualmente. Por ejemplo, si el entorno está actualmente en la 23.12, puede omitir la 24.01, la 24.02 y la 24.03, pero no la 24.04. Para obtener información detallada sobre cómo funcionan los retrasos de actualización, consulte Solicitud de un retraso de actualización para entornos de producción en *Guía de operaciones*.

> **Nota:**
>
> Si omite la actualización solo para uno de sus entornos (por ejemplo, omite la actualización en el entorno de producción, pero no en el entorno de prueba) durante tres meses, los entornos tendrán tres versiones de diferencia. Puede que no sea capaz de migrar las instantáneas de un entorno a otro en dicho escenario.
> Por ejemplo, suponga que los entornos de prueba y producción fueran actualmente de la versión 23.12, y que omite las actualizaciones para las versiones 24.01, 24.02 y 24.03 solo para el entorno de producción. Cuando la versión 24.03 esté disponible, el entorno de prueba será de la versión 24.03, mientras que el entorno de producción seguirá siendo de la versión 23.12. En este caso, no está soportada la migración entre los entornos de prueba y de producción.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate skipUpdate add|remove|list [version=UPDATE_NUMBER comment="COMMENT"]`
donde:

- `add` define una solicitud de omisión de actualización para una actualización mensual específica. Debe especificar los siguientes parámetros:
  - `version`: la actualización mensual que se debe omitir. Puede omitir una, dos o tres de las tres siguientes actualizaciones mensuales. Por ejemplo, si el entorno se encuentra en la actualización mensual 23.12, puede omitir la 24.01, la 24.02, la 24.03 o cualquier de estas actualizaciones. Para omitir tres actualizaciones mensuales, ejecute el comando tres veces especificando cada vez una actualización específica cuyo uso se va a omitir, por ejemplo, `version=24.01`, después,`version=24.02` y, por último, `version=24.03`. En este escenario, el entorno se actualizará a la línea de código principal en el ciclo mensual 24.04.

    Si hay un salto entre el ciclo mensual para el que se ha especificado una omisión de actualización y el ciclo mensual actual, Oracle actualizará el entorno según sea necesario y, después, omitirá las actualizaciones en el ciclo mensual especificado. Por ejemplo, el entorno está en la actualización mensual 23.12 y especifica una omisión de actualización para las versiones 24.02 y 24.03. En este caso, el entorno se actualizará a la 24.01; se omitirán las actualizaciones 24.02 y 24.03. El entorno se actualizará a la línea de código principal en la 24.04.
  - `comment`: texto que describe por qué es necesaria una omisión de actualización. Los comentarios deben ir entre comillas dobles.
- `remove` elimina todas las solicitudes de omisión de actualización especificadas para el entorno de modo que este pueda actualizarse a la línea de código principal durante el siguiente mantenimiento diario. Si tiene más de una solicitud de omisión de actualización en un entorno, este comando las elimina todas.

- `list` para mostrar las solicitudes de omisión de actualización (el identificador de inicio de sesión del usuario que realizó la solicitud de omisión, el comentario, la versión para la que se omitirán las actualizaciones y la fecha en la que se realizó la solicitud) definidas actualmente para el entorno, como se muestra en el siguiente gráfico:

```
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
```

Ejemplos

- Solicitar una omisión de actualización: `epmautomate skipUpdate add version=24.01 comment="We are in the process of closing the quarter"`
- Ver los detalles de una omisión de actualización: `epmautomate skipUpdate list`
- Eliminar todas las solicitudes de omisión de actualización: `epmautomate skipUpdate remove`

snapshotCompareReport

Compara dos instantáneas y crea el informe de comparación de instantáneas que identifica las diferencias en conjuntos de reglas y reglas de cálculo y formularios de datos incluidos en las instantáneas.

En el informe se muestran las diferencias en las reglas y las definiciones de formularios. También se identifica si una instantánea contiene una regla, un conjunto de reglas o un formulario que esté presente en la otra instantánea. Las diferencias en los conjuntos de reglas se identifican en función de las reglas que incluyen. Tenga en cuenta que el informe generado no notifica las diferencias en las propiedades de los cubos o de las dimensiones.

- Deterioro reciente del rendimiento en un entorno. Puede comparar la instantánea anterior con la actual para comprobar las diferencias que puede haber provocado el deterioro del rendimiento.
- Observa un rendimiento o comportamiento diferente entre dos entornos de los que esperaba que se comportasen o rindiesen de forma idéntica. En este caso, puede comparar las instantáneas de dos entornos para entender las diferencias entre ellos.
- Sospecha que algunas reglas o algunos formularios han desaparecido de un entorno. Utilice este informe para comparar los artefactos que había antes y los que hay ahora.

Se aplica a

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

Roles necesarios

Administrador del servicio

Sintaxis

`epmAutomate snapshotCompareReport SOURCE_SNAPSHOT TARGET_SNAPSHOT [reportName=REPORT_NAME.html]` donde:

- *SOURCE_SNAPSHOT* es el nombre de la instantánea con la que se va a hacer la comparación. El informe contiene datos sobre las diferencias en las reglas y los formularios en esta instantánea.
- *TARGET_SNAPSHOT* es el nombre de la instantánea que desea comparar.

> **Note:**
>
> – Los nombres de las instantáneas pueden especificarse con o sin la extensión .ZIP.
>
> – Ambas instantáneas deben estar disponibles en el entorno. Use uploadFile, copyFromObjectStorage o el comando copySnapshotFromInstance para cargarlas en el entorno.

- *REPORT_NAME*, opcionalmente, es el nombre del archivo de informe. El nombre de informe predeterminado es `SnapshotCompare.html`.
  Utilice el comando downloadFile para descargar el informe.

Ejemplos

- `epmAutomate snapshotCompareReport "Artifact Snapshot" 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

Ordena los miembros de dimensiones de entidad, cuenta, escenario, versiones y de dimensiones personalizadas.

Este comando ayuda a ordenar los miembros de dimensión después de cargar los miembros en la aplicación.

> **Nota:**
>
> No puede utilizar este comando para ordenar miembros de dimensiones de periodo, años y moneda.

Se aplica a

Planning, Planning Modules, FreeForm, Enterprise Profitability and Cost Management, Sales Planning y Strategic Workforce Planning.

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate sortMember Member [type=children|descendants] [order=ascending|descending]` donde:

- *`Member`* es el nombre del miembro padre cuyos descendientes o hijos se van a ordenar.
- `type`, especifica, de manera opcional, los miembros que se van a ordenar. Los valores aceptables son los siguientes:
  - `descendants` ordena todos los submiembros (hijos y descendientes) del miembro padre que especifique como el valor de `Member`.
  - `children`, el valor predeterminado, ordena solo los miembros en el nivel inmediatamente inferior al miembro padre que especifique como el valor de `Member`.
- `order`, identifica, de forma opcional, un orden de clasificación. Los valores aceptables son los siguientes:
  - `ascending` es el orden de clasificación predeterminado.
  - `descending`

Ejemplos

- Ordenar los hijos de la dimensión Entidad en orden ascendente: `epmautomate sortMember Entity`
- Ordenar todos los miembros secundarios de la dimensión Entidad en orden descendente: `epmautomate sortMember Entity type=descendants order=descending`

unassignRole

Elimina un rol asignado actualmente a los usuarios, incluido el usuario que ejecuta este comando, cuyos ID de inicio de sesión se incluyen en el archivo CSV con codificación ANSI o UTF-8 que se utiliza con este comando. Puede usar este comando para eliminar la asignación de un rol predefinido o un rol de aplicación.

El formato del archivo CSV es el siguiente:

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

Un administrador del servicio utiliza el comando uploadFile para cargar el archivo en el entorno.

> **Nota:**
>
> Utilice las comillas dobles para incluir los nombres de rol que contienen caracteres de espacio.

Cuando finalice, el comando imprime información sobre cada una de las entradas con fallos en la consola. Consulte esta información para saber el motivo por el que se ha producido un fallo en la ejecución del comando en algunas entradas del archivo CSV.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Para eliminar asignaciones de roles predefinidos:
  - Administrador del servicio y cualquier rol predefinido
  - Administrador de dominio de identidad y cualquier rol predefinido
- Para eliminar roles de aplicación:
  - Administrador del servicio
  - Cualquier rol predefinido y el rol de aplicación Control de acceso - Gestionar

Sintaxis

`epmautomate unassignRole` *`FILE_NAME ROLE`* donde:

- *`FILE_NAME`* es el nombre del archivo CSV que contiene los ID de inicio de sesión de los usuarios cuya asignación de roles se va a revocar. Especifique la extensión CSV en minúsculas.
  Los valores de User Login no distinguen mayúsculas y minúsculas. Por ejemplo, `jane.doe@example.com` se considera que es idéntico a `Jane.Doe@Example.com` o a cualquier variación de sus mayúsculas o minúsculas.
- `ROL` identifica una de las siguientes acciones. Los nombres de roles no distinguen mayúsculas y minúsculas.
  - Si elimina la asignación de usuarios a roles predefinidos, `ROL` debe identificar un rol predefinido aplicable al servicio. Consulte Descripción de los roles predefinidos en *Guía de introducción para administradores*.
  - Si elimina la asignación de usuarios a roles de aplicación, `ROLE` debe identificar un rol que pertenezca a la aplicación en el entorno actual. Los roles de aplicación se enumeran en el separador **Roles** de Control de acceso . Para obtener una descripción de los roles de aplicación para cada proceso de negocio, consulte estos temas en *Administración del control de acceso para Oracle Enterprise Performance Management Cloud*:
    - Account Reconciliation
    - Enterprise Profitability and Cost Management
    - Planning, FreeForm, Financial Consolidation and Close y Tax Reporting
    - Profitability and Cost Management
    - Oracle Enterprise Data Management
    - Narrative Reporting

Ejemplos

- Anule la asignación de los usuarios de un rol predefinido del dominio de identidad:

```
epmautomate unassignRole remove_roles.csv "Service Administrator"
```

- Anule la asignación de los usuarios de un rol de aplicación:
  `epmautomate unassignRole example_file.csv "Task List Access Manager"`

updateGuidedLearningSettings

Establece los valores necesarios para activar Oracle Guided Learning (OGL) en un entorno.

Se aplica a

Planning, Planning Modules, FreeForm, Financial Consolidation and Close, Tax Reporting, Enterprise Profitability and Cost Management, Sales Planning y Strategic Workforce Planning

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate updateGuidedLearningSettings [oglAppId=APP_ID] [oglServerUrl=OGL_SERVER_URL]`, donde:

- `OglAppId`, opcionalmente, es el ID de la aplicación de OGL. Se trata de un ID de una agrupación lógica de guías (contenido) en el servidor de OGL. Este valor puede contener hasta 100 caracteres.
- `OglServerUrl`, opcionalmente, es la URL del servidor de OGL donde tenga una cuenta de OGL activa. El valor predeterminado es `https://guidedlearning.oracle.com`. Esta URL debe especificar el protocolo y puede tener un máximo de 300 caracteres.

> **Note:**
>
> Al menos se necesita uno de los parámetros anteriores.

Ejemplo

```
epmautomate updateGuidedLearnignSettings "oglAppId=123xyz" "oglServerUrl=https://
guidedlearning.oracle.com"
```

updateUsers

Modifica atributos como el correo electrónico, el nombre y el apellido de los usuarios de Oracle Fusion Cloud Enterprise Performance Management en un dominio de identidad con los nuevos valores identificados en un archivo de valores separados por comas (CSV) codificado mediante UTF-8 o ANSI cargado en el entorno.

Un administrador del servicio utiliza el comando uploadFile para cargar archivos en un entorno. Todas las columnas del archivo CSV son obligatorias; debe proporcionar una entrada válida en cada una de las columnas. Este comando valida cada una de las definiciones de estos valores obligatorios y muestra un mensaje de error que identifica cada uno de los valores que faltan o no válidos. El formato del archivo de entrada es el siguiente:

```
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 el valor Inicio de sesión de usuario del archivo CSV coincide con una cuenta que exista en el dominio de identidad, el comando modifica la cuenta de usuario para que coincida con los valores del archivo de entrada. Dado que las cuentas de usuario son comunes a todos los entornos que soporta un dominio de identidad, la información de usuario actualizada está disponible para todos los entornos que comparten el dominio de identidad. Los roles específicos de la aplicación y predefinidos asignados al usuario no se ven afectados por este comando

> **Note:**
>
> - No puede usar este comando para modificar los valores de User Login.
> - No puede modificar los atributos de su propia cuenta.
> - El archivo de entrada que contiene caracteres de varios bytes debe utilizar la codificación de caracteres UTF-8.

Se aplica a

Planning, Planning Modules, 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 y Sales Planning.

Roles necesarios

Administrador de dominio de identidad y cualquier rol predefinido

Sintaxis

`epmautomate updateUsers` *`FILE_NAME`*, donde *`FILE_NAME`* es el nombre de un archivo CSV que contiene la información de usuario que se va a modificar.

Ejemplo

```
epmautomate updateUsers update_user_info.csv
```

upgrade

Descarga automáticamente la versión más reciente de EPM Automate y la instala de forma silenciosa.

Una vez que se ha ejecutado el comando login para iniciar una sesión, EPM Automate identifica la versión instalada actualmente. Si la versión instalada no es la más reciente disponible, le informa de que hay una versión disponible más actual.

> **Nota:**
>
> La instancia de EPM Automate desplegada por un administrador de Windows solo se puede actualizar si el usuario que ha iniciado sesión es un administrador de Windows.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado
- Usuario
- Visor

Sintaxis

```
epmautomate upgrade
```

Ejemplo

```
epmautomate upgrade
```

uploadFile

Carga un archivo del equipo local en el servicio. Utilice este comando para cargar archivos que contengan datos, metadatos, definiciones de reglas, definiciones de dimensiones, transacciones asignadas, plantillas e instantáneas de copia de seguridad.

Con este comando no se sobrescriben los archivos existentes en el entorno. Muestra un error si el nombre del archivo que se está cargando es idéntico al de un archivo en la ubicación de carga.

Con este comando no se pueden cargar archivos binarios (por ejemplo, archivos con extensiones como .exe, .com, .cmd, .bin, .bat. .msi y .vbs) y archivos con extensiones no permitidas.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Usuario avanzado y el rol de aplicación Migraciones - Administrar (Oracle Enterprise Data Management Cloud y Profitability and Cost Management)

Sintaxis

`epmautomate uploadFile "FILE_NAME" [UPLOAD_LOCATION]`, donde:

- *`FILE_NAME`* es el nombre del archivo, incluida la ruta absoluta si el archivo no está en el directorio desde el que está ejecutando EPM Automate.

- *UPLOAD_LOCATION* es, opcionalmente, la ubicación de Oracle Fusion Cloud Enterprise Performance Management en la que desea cargar el archivo. No especifique una ubicación de carga si desea cargar archivos en la ubicación de carga predeterminada. Para obtener información detallada, consulte Ubicaciones de archivo predeterminadas. Entre los valores soportados se incluyen:
  - `inbox` para cargar archivos en el buzón. A excepción de Profitability and Cost Management, los procesos de negocio de Cloud EPM buscan los archivos en esta ubicación para procesarlos.
  - `profitinbox` para cargar los archivos que va a procesar Profitability and Cost Management.
  - `to_be_imported` para cargar una instantánea de Narrative Reporting que se vaya a importar durante el siguiente mantenimiento diario del entorno.
  - `inbox/`*`directory_name`* para cargar archivos en un directorio en la bandeja de entrada para su procesamiento por parte de Data Management.
  - `outbox` para cargar los archivos en la bandeja de salida que usan los procesos de negocio distintos de Profitability and Cost Management.
  - `profitoutbox` para cargar los archivos en la bandeja de salida que usa Profitability and Cost Management.

Ejemplos

- Cargue una instantánea en la ubicación predeterminada:
  `epmautomate uploadFile "C:/snapshots/backup_snapshot.zip"`
- Cargue un archivo en el buzón de Data Management:
  `epmautomate uploadFile "C:/pbcsdata/quarterlydata.csv" inbox`
- Cargue un archivo en una carpeta del buzón (para Data Management):
  `epmautomate uploadFile "C:/fdmee_data/data.zip" inbox/dm_folder`
- Cargue un archivo en profitinbox (Profitability and Cost Management):
  `epmautomate uploadFile "C:/profitability_data/data.zip" profitinbox`
- Cargue la instantánea de Narrative Reporting del directorio `C:\temp` en la ubicación to_be_imported:
  `epmautomate uploadFile "C:\temp\EPRCS_Backup.tar.gz" to_be_imported`

userAuditReport

Genera un informe de auditoría de usuarios (archivo .CSV) y lo almacena en la ubicación de descarga predeterminada.

El informe de auditoría de usuarios contiene información sobre los usuarios que iniciaron sesión en un entorno en el periodo de tiempo especificado (como máximo los últimos 120 días). Muestra el ID de conexión de usuario, la dirección IP del equipo desde el que ha iniciado sesión el usuario y la fecha y hora (por ejemplo, July 28, 2022 18:43:21 UTC) en las que el usuario ha accedido al entorno.

> **Nota:**
>
> En el informe de auditoría de usuarios solo se muestra una entrada de inicio de sesión para un usuario que haya iniciado sesión en un entorno de Oracle Fusion Cloud Enterprise Performance Management varias veces en un intervalo de cinco minutos.

	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

Utilice el comando downloadFile para descargar el informe generado en su equipo.

Se aplica a

Planning, Planning Modules, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Gestionar
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Ver

Sintaxis

`epmautomate userAuditReport` *`FROM_DATE TO_DATE REPORT_NAME`* donde:

- *`FROM_DATE`* indica la fecha de inicio (con formato `YYYY-MM-DD`) del período para el que va a generarse el informe de auditoría
- *`TO_DATE`* indica la fecha de finalización (con formato `YYYY-MM-DD`) del período para el que va a generarse el informe de auditoría
- *`NOMBRE_INFORME`* es el nombre del archivo de informe

> **Nota:**
>
> Este informe solo se puede generar para los últimos 120 días.

Ejemplo

```
epmautomate userAuditReport 2016-10-15 2016-12-15 myAuditReport.CSV
```

userGroupReport

Genera un informe (archivo CSV) que muestra los grupos a los que están asignados los usuarios en Control de acceso y lo almacena en la ubicación de descarga predeterminada.

El informe indica si la asignación de usuario a un grupo es directa (como miembro de un grupo) o indirecta (como miembro de un grupo que es hijo de un grupo anidado).

El informe identifica el nombre de inicio de sesión, el nombre, el apellido, la dirección de correo electrónico, el grupo asignado y el tipo de asignación del usuario con el siguiente formato. Es idéntico a la versión CSV del informe creado desde el separador Informe de grupo de usuarios en Control de acceso. Por ejemplo, suponga que el usuario `jdoe` es miembro del grupo `Test1`, el cual es hijo del grupo anidado `Test2`. En este escenario, el informe mostrará la siguiente información para `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
```

Utilice el comando downloadFile para descargar el informe generado en su equipo.

Se aplica a

Planning, Planning Modules,, 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 y Strategic Workforce Planning.

Roles necesarios

- Administrador del servicio
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Gestionar
- Cualquier rol predefinido y el rol de aplicación Control de acceso - Ver

Sintaxis

`epmautomate userGroupReport` *`REPORT_NAME`*, donde *`REPORT_NAME`* es el nombre del archivo de informe.

Ejemplo

```
epmautomate userGroupReport UsrGrpReport.CSV
```

validateConsolidationMetadata

Valida los metadatos del entorno para garantizar el refrescamiento y la consolidación sin errores de la base de datos.

Después de importar los metadatos mediante el comando importMetadata, ejecute este comando para validar los metadatos a fin de garantizar un refrescamiento sin errores de la base de datos al ejecutar el comando refreshCube. Si los metadatos de consolidación no son correctos, es posible que también falle la consolidación.

Este comando muestra 0 (cero) o un recuento del números de errores de validación en la consola desde la que se ejecuta. Escribe errores de validación en un archivo CSV, que puede

utilizar para corregir los errores de metadatos. Utilice el comando downloadFile para descargar el archivo CSV resultante en un servidor local.

Se aplica a

Financial Consolidation and Close

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate validateConsolidationMetadata` *`LOG_FILE_NAME`* donde `LOG_FILE_NAME` identifica el nombre del archivo que incluirá la información sobre los errores identificados por este comando.

Ejemplos

```
epmautomate validateConsolidationMetadata validation_error.csv
```

validateModel

Valida un modelo de Enterprise Profitability and Cost Management y escribe la salida de validación en un archivo.

Se aplica a

Enterprise Profitability and Cost Management

Roles necesarios

Administrador del servicio

Sintaxis

`epmautomate validateModel "modelName"` *`FILE_NAME`*`.txt [messageType=All|Warning|Error] [ruleStatus=All|Enabled|Disabled]`, donde:

- `modelName` es el nombre del modelo de Enterprise Profitability and Cost Management que se va a validar. Este valor se debe escribir entre comillas dobles.
- *`FILE_NAME`* es un nombre único para un archivo de texto en el que debe escribir la salida de la validación del modelo. Este archivo, creado en el buzón de salida, se puede descargar con el comando downloadFile.
- `messageType`, opcionalmente, es el estado de la información que se incluirá en la salida de validación del modelo. Los valores posibles del parámetro son los siguientes:
  - `All`, que escribe tanto los errores como las advertencias en el archivo de salida de validación.
  - `Error`, que registra solo los errores en el archivo de salida de validación. Este es el valor predeterminado.
  - `Warning`, que registra solo las advertencias de validación del modelo en el archivo de salida de validación.
- `ruleStatus`, opcionalmente, es uno de los siguientes:

- `All`, que valida todas las reglas para el modelo seleccionado. Este es el valor predeterminado.
- `Enabled`, que valida solo las reglas activadas.
- `Disabled`, que valida solo las reglas desactivadas.

Ejemplo

```
epmautomate validateModel "10 Actuals Allocation Process" results.txt
messageType=All ruleStatus=Enabled
```

Códigos de salida

EPM Automate devuelve un código de salida y un mensaje para indicar el estado de la operación. Los códigos de salida se agrupan en cinco números de código; cada código puede indicar muchas condiciones de error. Revise el mensaje adjunto para identificar la condición específica que causó el error.

Además, EPM Automate crea un archivo de registro (*`NOMBRE_COMANDO_MARCA_TIEMPO.log`*, por ejemplo, `uploadfile_16_11_2016_11_27_10.log`) para cada ejecución de comando fallida. Los archivos de registro se crean en la computadora desde la que se ejecuta EPM Automate.

Errores de código de salida 1

`Error del comando al ejecutarse` EPM Automate usa este código de salida para mostrar los mensajes relacionados con el código de estado HTTP 200 y 400. Estos códigos son devueltos por las API de REST que usa EPM Automate.

`Privilegios insuficientes para realizar la operación` Este error se muestra si el usuario cuyas credenciales se utilizan para conectarse al servicio no tiene privilegios suficientes para realizar la operación que ha intentado.

Conéctese con una cuenta que tenga privilegios suficientes para realizar la operación. Normalmente, solo los administradores del servicio pueden realizar operaciones en el servicio.

`El recurso no existe` Este error se muestra si el archivo o instantánea que desea suprimir o descargar no existe en el servicio.

Utilice el comando `listfiles` para verificar el nombre del archivo y su ubicación.

`Instantánea no válida` ***`SNAPSHOT`*** Este error se muestra cuando el servicio no puede validar la instantánea que ha especificado para la operación de exportación o importación.

Verifique que está utilizando una instantánea válida.

`Error del servidor interno. No es posible suprimir el archivo:` ***`FILE_NAME`*****`Utilice "Proporcionar comentarios" con los detalles`** Este error se muestra si el archivo o instantánea no se ha podido suprimir del servicio debido a un error del servidor.

Informe de este problema a Oracle con el comando Comentarios o con la función Proporcionar comentarios.

`Archivo no válido:` ***`FILE_NAME`*** Este error se muestra si el archivo o la instantánea que desea suprimir o descargar no existe en el servicio o si el nombre de archivo no está en el formato requerido.

Utilice el comando `listfiles` para verificar el nombre del archivo y su ubicación.

`Recrear lleva bastante tiempo ejecutándose. Póngase en contacto con los servicios de soporte` Este error se muestra si la operación de nueva creación que ha iniciado no se termina en una hora.

Informe de este problema a Oracle con el comando de comentarios o con la función Proporcionar comentarios.

`Restablecer lleva bastante tiempo ejecutándose. Póngase en contacto con los servicios de soporte` Este error se muestra si la operación de restablecimiento que ha iniciado no se termina en una hora.

Informe de este problema a Oracle con el comando de comentarios o con la función Proporcionar comentarios.

`No puede realizar la operación. Hay otra instancia en curso. Vuelva a intentarlo más tarde` Este error se muestra si intenta ejecutar el comando `copysnapshotfrominstance` cuando otra instancia del comando está activa.

Espere a que el comando `copysnapshotfrominstance` finalice antes de intentar ejecutar el comando de nuevo.

`No puede realizar la operación. Hay otro script de mantenimiento en curso. Vuelva a intentarlo más tarde` Este error se muestra si intenta ejecutar el comando `copysnapshotfrominstance`, `recreate` o `resetservice` cuando el proceso de restablecimiento del servicio o de mantenimiento diario está en ejecución.

Vuelva a ejecutar la operación cuando termine el proceso de mantenimiento o restablecimiento.

`Error al conectar con la instancia de origen: SOURCE_URL` Este error se muestra si EPM Automate no puede conectarse al entorno de origen para iniciar el comando `copysnapshotfrominstance`.

Verifique que las credenciales, el dominio de identidad y la URL que se utilizan para acceder al entorno de origen son válidos.

`Error del servidor interno. Error al copiar la instantánea de la instancia de origen. Utilice "Proporcionar comentarios" con los detalles` Este error se muestra cuando EPM Automate detecta un problema inesperado al ejecutar el comando `copysnapshotfrominstance`.

Informe de este problema a Oracle con el comando`feedback` o con la función Proporcionar comentarios.

`Error del servidor interno. Utilice "Proporcionar comentarios" con los detalles` Este error se muestra para indicar muchas condiciones del servidor interno que requieren acciones correctivas por parte de Oracle.

Informe de este problema a Oracle con el comando`feedback` o con la función Proporcionar comentarios.

La **`instantánea`***`SNAPSHOT_NAME`***` ya existe. Suprima la instantánea y vuelva a ejecutar el comando`** Este error se muestra cuando descarga o carga una instantánea en una ubicación en la que hay otra instantánea con un nombre idéntico.

Suprima o elimine la instantánea existente y, a continuación, vuelva a ejecutar el comando.

`Error al extraer la instantánea. Vuelva a intentarlo con una instantánea correcta` Este error se muestra si EPM Automate no puede extraer el contenido de la instantánea al ejecutar el comando `importsnapshot`.

Verifique que la instantánea es válida.

`Error del servidor interno. No es posible abrir el archivo para escribirlo. Utilice "Proporcionar comentarios" con los detalles` Este error se muestra si los errores provocan la creación o actualización de archivos CSV, por ejemplo, al generar el informe de auditoría.

Informe de este problema a Oracle con el comando`feedback` o con la función Proporcionar comentarios.

`No se han encontrado registros coincidentes. Seleccione otro rango de fechas.` Este error se muestra si ejecuta el comando `userauditreport` para generar el informe de auditoría para un rango de fechas durante el que los datos de auditoría no estarán disponibles.

Especifique un rango de fechas válido y, a continuación, vuelva a ejecutar el comando `userauditreport`. Tenga en cuenta que el servicio solo mantiene el historial de auditoría de los últimos 365 días.

`Ya existe un archivo con el mismo nombre: FILE_NAME. Seleccione otro nombre de archivo` Este error se muestra si un informe con el nombre de informe de auditoría que ha especificado ya existe en el servicio.

Suprima el archivo existente del servicio o especifique un nombre de archivo diferente y, a continuación, vuelva a ejecutar el comando `userauditreport`.

`Fallo en la operación con el estado $1. Utilice "Proporcionar comentarios"` Este mensaje indica un error interno del servidor que provoca un error en el proceso de restablecimiento o nueva creación del servicio.

Informe de este problema a Oracle con el comando`feedback` o con la función Proporcionar comentarios.

`EPMCSS-20643: Fallo al agregar usuarios. No se encontró el archivo .csv FILE_NAME`. Proporcione un nombre de archivo válido Este error se muestra si el archivo CSV especificado que contiene información sobre los usuarios que deben agregarse no está disponible en la bandeja de entrada.

Utilice el comando `listfiles` para verificar el nombre del archivo y su ubicación. Si el archivo no está en la bandeja de entrada, utilice el comando uploadFile para cargar el archivo.

`EPMCSS-20644: Fallo al eliminar usuarios. No se encontró el archivo .csv FILE_NAME. Proporcione un nombre de archivo válido` Este error se muestra si el archivo CSV especificado que contiene información sobre los usuarios que deben suprimirse no está disponible en la bandeja de entrada.

Utilice el comando `listfiles` para verificar el nombre del archivo y su ubicación. Si el archivo no está en la bandeja de entrada, utilice el comando uploadFile para cargar el archivo.

`20645: Fallo al asignar el rol a los usuarios. Nombre de rol no válido rol. Proporcione un nombre de rol válido` Este error se muestras si el rol especificado en el archivo CSV no está soportado.

Verifique que el nombre del rol especificado en el archivo es `Administrador del servicio`, `Usuario avanzado`, `Usuario` o `Visor`.

Utilice el comando `listfiles` para verificar el nombre del archivo y su ubicación. Si el archivo no está en la bandeja de entrada, utilice el comando uploadFile para cargar el archivo.

Errores de código de salida 6

`Servicio no disponible` El servicio no está disponible debido al error de HTTP 404.

Verifique la disponibilidad del servicio accediendo al mismo desde un explorador en la computadora desde el que está ejecutando EPM Automate. Si el servicio está caído por cualquier motivo, espere unos minutos y vuelva a intentarlo o póngase en contacto con los servicios de soporte de Oracle.

`Tiempo de espera de lectura/escritura` Este error se muestra si el socket de cliente sufre un timeout (que dura unos 15 minutos) durante cualquier operación de lectura/escritura debido a que la red es lenta o a problemas con el cortafuegos.

Vuelva a ejecutar el comando fallido cuando el rendimiento global de la red sea alto. Si el fallo se debe a la configuración del cortafuegos, póngase en contacto con el administrador de red.

Errores de código de salida 7

EPM Automate muestra este error si no ha podido ejecutar un comando. El mensaje de error, por ejemplo, `Comando no válido`, especifica por qué se ha producido el error.

`No es posible abrir el archivo de contraseñas`*`FILE_NAME`* Archivo de contraseñas cifrado no válido, por ejemplo, *`PASSWORD_FILE`*`.EPW`. EPM Automate no ha encontrado el archivo en la ubicación especificada o el archivo no está en el formato necesario.

Verifique el nombre del archivo y la ruta. Si el archivo no se puede analizar porque el formato no es válido, utilice el comando encrypt para volver a crear el archivo.

`No es posible analizar el archivo de contraseñas` ***FILE_NAME*** EPM Automate no ha podido analizar el archivo de contraseñas cifrado porque el formato no es válido o porque se ha dañado.

Utilice el comando encrypt para volver a crear el archivo.

`No es posible conectar con la`*`URL`*`. Causa raíz`*`MESSAGE`* Este error se muestra si no se puede establecer una conexión debido a una URL incorrecta. El mensaje que se muestra como causa raíz detalla el fallo subyacente que se produce por el uso de una URL incorrecta.

- Verifique que está utilizando una URL válida
- Si la configuración de proxy requiere la autentificación con el servidor proxy para conectarse a Internet, especifique un nombre de usuario, dominio y contraseña de servidor de proxy (o utilice un archivo de contraseñas cifradas que contenga la contraseña del servidor proxy) para iniciar sesión. Póngase en contacto con el administrador de red si necesita ayuda.

`No es posible conectar con la`*`URL`*`Protocolo no soportado` El comando login ha fallado porque la URL especificada utiliza un protocolo no soportado. El mensaje de error adjunto identifica el protocolo no soportado.

Asegúrese de que la URL que está utilizando con el comando de conexión utiliza el protocolo seguro (HTTPS).

`La sesión no está autenticada. Ejecute el comando "login" antes de ejecutar cualquier otro comando` Ha intentado ejecutar un comando antes de establecer una sesión con el comando `login`.

Ejecute el comando login para establecer una conexión segura con el entorno antes de intentar ejecutar otros comandos.

`Parámetro no válido` Este mensaje indica un error de sintaxis en un comando causado por una secuencia incorrecta de parámetros de comando o la ausencia de algunos valores de parámetros de comando necesarios.

Revise y corrija los parámetros de comando y la secuencia en que se han especificado.

El comando `COMMAND_NAME` no está soportado por `SERVICE_TYPE` EPM Automate no ha podido ejecutar el comando en el servicio al que está conectado porque el proceso de negocio no soporta el comando.

Consulte Referencia de comandos para visualizar listas de comandos soportados por cada proceso de negocio.

`El archivo no existe en la ubicación: PATH` EPM Automate no ha encontrado el archivo que desea procesar, por ejemplo, mediante el comando `upload` o `replay`.

Asegúrese de que el nombre de archivo y la ruta son correctos.

`No es posible abrir el archivo para lectura: PATH` EPM Automate no ha podido leer el archivo especificado.

Asegúrese de que el archivo está en el formato necesario. Verifique que el usuario que está ejecutando EPM Automate tiene acceso de lectura al archivo.

`No es posible abrir el archivo para escritura: PATH` EPM Automate no ha podido escribir en el archivo especificado.

Asegúrese de que el archivo no está bloqueado por otro proceso. Verifique que el usuario que está ejecutando EPM Automate tiene acceso de escritura al archivo.

`Comando no válido` EPM Automate ha encontrado un comando no soportado.

Verifique que EPM Automate admite el comando; además, asegúrese de que el nombre del comando está escrito correctamente.

`Formato de fecha no válido` La herramienta ha encontrado un formato de fecha no válido.

Especifique las fechas de generación de informes en un formato de fecha soportado.

`FROMDATE DATE no puede ser posterior a TODATE DATE` EPM Automate ha encontrado una fecha de finalización que es anterior a la fecha de inicio.

Asegúrese de que el valor de `to date` en un rango de fechas especificado es una fecha posterior al valor de `from date`.

`Se ha excedido el número máximo de comentarios (6) para un día` Este error se muestra al superar el número de comentarios que puede enviar mediante el comando `feedback`.

`Ya existe un archivo con el mismo nombre en la ruta de descarga PATH`. Suprima el archivo y vuelva a ejecutar el comando Este error se muestra cuando se intenta descargar un archivo en una ubicación que ya tiene un archivo que coincide con su nombre.

Suprima, cambie el nombre o mueva el archivo existente y, a continuación, vuelva a ejecutar el comando.

`El archivo está vacío: PATH` Este error se muestra si el archivo de reproducción no tiene ningún contenido.

Asegúrese de que el archivo de reproducción (archivo CSV) muestra las credenciales (nombre de usuario y contraseña) y el nombre de los archivos HAR que se van a usar para ejecutar el comando `replay`.

`No es posible cifrar la contraseña ya que el host local no se puede resolver. Asegúrese de que los nombres de host se han asignado a las direcciones IP correctamente` Este error se muestra si EPM Automate no ha podido resolver la definición de host local en una dirección MAC porque el archivo de hosts en su computadora contiene un nombre de servidor en lugar de `localhost` para la dirección `127.0.0.1`.

Asegúrese de que el archivo de hosts especifica `localhost` como nombre del servidor para `127.0.0.1`

`El nombre de la instantánea no es válido.` Este error se muestra si no especifica el nombre de la instantánea cuyo nombre se va a cambiar.

Especifique el nombre de una instantánea disponible en el entorno.

`El nuevo nombre de la instantánea no es válido` Este error se muestra si no especifica un nombre nuevo para la instantánea.

Especifique el nuevo nombre de la instantánea.

`Nombre de la instantánea no válido: {0}. No se permiten los caracteres \\/*?"<>|` Este error se muestra si el nombre de la instantánea contiene caracteres especiales como espacio, \ (barra invertida), / (barra), * (asterisco), ? (signo de cierre de interrogación), " (comillas), < (menor que) y > (mayor que).

Especifique un nombre de instantánea nuevo que no contenga estos caracteres especiales.

`No es posible cambiar el nombre de la instantánea: {0}. Podría haber otro proceso para acceder a ella. Inténtelo en otro momento` Este error se muestra si EPM Automate no puede obtener un bloqueo exclusivo de la instantánea porque otro proceso la está usando.

Espere a que finalice la operación actual que está utilizando la instantánea y, a continuación, vuelva a intentarlo.

`La instantánea {0} ya existe. Suprima la instantánea y vuelva a ejecutar el comando` Este error se muestra cuando el nuevo nombre de la instantánea es idéntico al de una instantánea existente en el entorno.

Utilice un nombre de instantánea diferente o suprima la instantánea existente mediante el uso del comando `deletefile`.

Errores de código de salida 9

`Credenciales no válidas` Este error aparece cuando el nombre de usuario o la contraseña utilizados con el comando login son incorrectos.

Especifique las credenciales válidas para el entorno al que intenta conectar.

`Fallo en la autentificación al ejecutar el comando. Vuelva a intentarlo` Este error aparece cuando la autenticación básica falla durante la ejecución de un comando que no sea login. Este error se puede producir también para llamadas HTTP cuando la ejecución de un comando se vuelve a intentar (hasta tres veces).

Errores de código de salida 11

Error del servidor interno. Debido al restablecimiento manual del servicio, el entorno de Oracle EPM Cloud Service no se encuentra disponible actualmente. Este error se muestra si se ejecutan los comandos de EPM Automate cuando está en curso un restablecimiento del entorno.

`Error interno del servidor MENSAJE` Este error se muestra si EPM Automate encuentra excepciones desconocidas que no están relacionadas con las conexiones HTTP. Incluye los errores del servidor 503 y 500.

`No es posible conectar a URL: MENSAJE` Este error se muestra cuando el servidor no está disponible. El mensaje de error indica la excepción que ha provocado que falle el comando.

Si el servidor no está disponible, póngase en contacto con los Servicios de Soporte Oracle. Si el mensaje indica problemas con la URL, verifique que la URL que está utilizando es válida.

3

Escenarios de ejemplo de ejecución de comandos

EPM Automate se puede utilizar para automatizar muchas tareas administrativas comunes de Oracle Fusion Cloud Enterprise Performance Management.

- Escenarios de ejemplo para todos los servicios
- Escenarios de ejemplo para Planning, Consolidation, Tax Reporting y Enterprise Profitability and Cost Management
- Escenarios de ejemplo para Account Reconciliation
- Escenarios de ejemplo para Profitability and Cost Management
- Escenarios de ejemplo para Oracle Enterprise Data Management Cloud

Acerca de la copia de scripts de ejemplo

No copie los scripts de ejemplo de la versión en PDF de este documento. Para evitar saltos de líneas e información de pie de página que inutilizarán los scripts, Oracle recomienda que copie los scripts de ejemplo de la versión HTML de *Trabajar con EPM Automate*.

Escenarios de ejemplo para todos los servicios

En los siguientes escenarios se describe una secuencia de comandos típica que se puede utilizar para realizar operaciones específicas en los entornos de Oracle Fusion Cloud Enterprise Performance Management.

Consulte también:

- Copia de seguridad de instantáneas de aplicación en un equipo
  En este caso se explica cómo automatizar el proceso de creación de una copia de seguridad de la instantánea creada durante el mantenimiento del servicio diario en un equipo local.
- Información a los usuarios de la finalización del mantenimiento diario
  El mantenimiento diario de los entornos de Oracle Fusion Cloud Enterprise Performance Management normalmente tarda mucho menos tiempo que la hora que tiene destinada.
- Copia de una instantánea en o de Oracle Object Storage
- Creación de usuarios y asignación a roles predefinidos
  Utilice los scripts de esta sección para crear usuarios y asignarlos a roles predefinidos en el dominio de identidad.
- Recuento del número de usuarios con licencia (usuarios asignados a roles)
  Utilice el script de esta sección para generar el Informe de asignación de roles para contar el número de usuarios de un entorno.
- Creación de informes de auditoría de usuarios asignados a roles
  Utilice los scripts de esta sección para automatizar el proceso de creación de un informe de auditoría para usuarios asignados a roles predefinidos en un entorno y, opcionalmente, enviarlo por correo electrónico a un destinatario.

- Creación de informe de auditoría de asignación y revocación de roles
  Utilice el script de PowerShell de esta sección para automatizar el proceso de creación de un informe de auditoría que detalla la asignación de roles y la revocación de roles en un entorno.
- Enmascaramiento de registros de acceso e informe de actividades para cumplir las normas de privacidad
  Utilice los scripts de esta sección para automatizar el proceso de enmascaramiento de información en el Informe de actividad o los Registros de acceso para cumplir con las leyes de privacidad y para, si lo desea, enviar por correo electrónico el informe a un destinatario.
- Automatización de descargas de informes de actividad en un equipo local
  Utilice el script de esta sección para automatizar la descarga de informes de actividad de un entorno en un equipo local.
- Descarga de logs de acceso desde un entorno
  Utilice el script de esta sección para automatizar el proceso de descarga de registros de acceso de un entorno en un equipo local.
- Automatización de la clonación de entornos
  Utilice el script de esta sección para automatizar la clonación de entornos.
- Clonación diaria del entorno principal al entorno de espera una vez completado el mantenimiento diario en el entorno principal
  Para mantener el entorno en espera actualizado con el entorno principal, utilice estos scripts para clonar el entorno principal de Oracle Fusion Cloud Enterprise Performance Management en el entorno en espera justo después de que haya terminado el mantenimiento diario en el entorno principal.
- Eliminación de archivos innecesarios desde un entorno
  Utilice estos scripts para eliminar archivos innecesarios de un entorno.
- Búsqueda y descarga de archivos de un entorno
  Utilice el script de ejemplo de esta sección para automatizar el proceso de descarga de uno o más archivos des de un entorno de Oracle Fusion Cloud Enterprise Performance Management utilizando una cadena de texto como comodín.
- Nueva creación de un entorno antiguo de Cloud EPM para auditorías
  Utilice el script de esta sección para crear una solución de autoservicio para mantener una biblioteca actualizada de instantáneas para su entorno de Oracle Fusion Cloud Enterprise Performance Management. Necesita un entorno dedicado al propósito de actualizar y mantener una biblioteca de instantáneas actualizadas.
- Automatización de auditoría y conformidad sobre el acceso a la base de datos
  Utilice los scripts de PowerShell y Bash Shell de esta sección para usar los comandos de EPM Automate a fin de recopilar datos de auditoría y conformidad sobre el acceso manual a la base de datos.
- Replicación de usuarios y asignación de role predefinidos
  Los scripts de esta sección le ayudan a migrar los usuarios y las asignaciones de roles predefinidos de un entorno a otro.
- Creación de una cadencia de cambio de versión trimestral de Cloud EPM
  Utilice estos scripts para crear una solución de autoservicio y omitir las actualizaciones para que los entornos de Cloud EPM se actualicen trimestralmente con un ciclo de prueba de dos semanas. En este caso, los entornos de producción se actualizan dos semanas más tarde que los entornos de prueba.

- Creación de una cadencia de cambio de versión trimestral de Cloud EPM con ciclos de prueba de seis semanas
  Utilice el script de esta sección para crear una solución de autoservicio y omitir las actualizaciones para que los entornos de Oracle Fusion Cloud Enterprise Performance Management se actualicen trimestralmente con un ciclo de prueba de seis semanas. En este caso, los entornos de producción se actualizan seis semanas más tarde que los entornos de prueba.

Copia de seguridad de instantáneas de aplicación en un equipo

En este caso se explica cómo automatizar el proceso de creación de una copia de seguridad de la instantánea creada durante el mantenimiento del servicio diario en un equipo local.

- Descarga la instantánea de la aplicación (`Artifact Snapshot`) que se creó en el periodo de mantenimiento
- Cambia el nombre de la instantánea descargada agregando el registro de hora
- Mantiene 10 copias de seguridad suprimiendo, si es necesario, las copias de seguridad más antiguas

> **Note:**
>
> - Este script no puede utilizarse para hacer una copia de seguridad de Narrative Reporting
> - Si vuelve a planificar este script para su uso, modifique los valores de los parámetros de tiempo de ejecución (`url`, `user`, `password` y `NumberOfBackups`) según sea necesario.

Consulte Automatización de la ejecución de scripts para obtener información sobre la programación del script con Programador de tareas de Windows.

Script de Windows de ejemplo

Cree un archivo por lotes (`.bat`) o de shell (`.sh`) que contenga un script similar al siguiente para automatizar las descargas de instantáneas.

```
@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

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%
```

Script de Linux/UNIX de ejemplo

Cree un archivo de shell (`.sh`) que contenga un script similar al siguiente para automatizar las descargas de instantáneas. Si su contraseña contiene caracteres especiales, consulte Manejo de caracteres especiales.

```
#!/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/jdk1.8.0_191/

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`

    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 --
```

Información a los usuarios de la finalización del mantenimiento diario

El mantenimiento diario de los entornos de Oracle Fusion Cloud Enterprise Performance Management normalmente tarda mucho menos tiempo que la hora que tiene destinada.

La duración del mantenimiento diario real se registra como valor de la métrica "Duración del mantenimiento diario en minutos" en la sección "Métricas de operaciones" del informe de actividad. Si no desea esperar la hora completa para utilizar el entorno, use una versión actualizada de este script para informar a los usuarios de que el mantenimiento diario se ha completado para que puedan reanudar las actividades.

Script de Windows

Cree `daily_maintenance_completed.ps1` copiando el siguiente script de PowerShell. Consulte Ejecución del script para obtener información sobre la actualización del script para utilizarlo.

```
# 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>"
    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
            $serverdown=$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 de Linux/UNIX

Cree `daily_maintenance_completed.sh` copiando el siguiente script. Consulte Ejecución del script para obtener información sobre la actualización del script para utilizarlo.

```
#!/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=""

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
}

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 de Groovy del servidor

Cree el script de Groovy `daily_maintenance_completed` copiando el siguiente código. Consulte Ejecución del script para obtener información sobre la actualización del script para utilizarlo.

```
// 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)
}

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
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.")
```

Ejecución del script

Windows y Linux/UNIX

1. Cree `daily_maintenance_completed.ps1` o `daily_maintenance_completed.sh` copiando el script de una sección anterior.
2. Actualice el script:
   - **Windows:** actualice el valor de `emailaddresses` con una lista separada por comas de direcciones de correo electrónico a las que se debe enviar una notificación cuando se complete el mantenimiento diario.
   - **Linux/UNIX:** actualice las siguientes variables:

  - `epmautomatescript` con la ubicación del ejecutable de EPM Automate. Ejemplo: `epmautomatescript="/home/utils/EPMAutomate/bin/epmautomate.sh"`
  - `javahome` con el directorio donde se instalar el JDK que utiliza EPM Automate. Por ejemplo: `"/home/user1/jdk1.8.0_191"`
  - `emailaddresses` con una lista separada por comas de direcciones de correo electrónico a las que se debe enviar una notificación cuando se complete el mantenimiento diario. Por ejemplo: `jdoe@example.com,jane_doe@example.com`

3. En una ventana de comandos o una consola, vaya hasta la carpeta donde está almacenado el script `daily_maintenance_completed`.
4. Ejecute este comando:
   - **Windows**: `./daily_maintenance_completed.ps1` *`USERNAME PASSWORD URL`*
   - **Linux/UNIX**: `./daily_maintenance_completed.sh` *`USERNAME PASSWORD URL`*, donde:
     - `USUARIO` es el nombre de usuario de un administrador de servicio
     - `CONTRASEÑA` es la contraseña del administrador de servicio
     - `URL` es la URL del entorno de Cloud EPM

Groovy del servidor:

1. Cree el script de Groovy `daily_maintenance_completed.groovy` copiándolo de una sección anterior.
2. Actualice estos valores.
   - `username` con el nombre de usuario de un administrador del servicio.
   - `password` con la contraseña del administrador del servicio
   - `url` con la URL del entorno de Cloud EPM para el que se debe realizar la notificación de finalización del mantenimiento diario. Por ejemplo: . Ejemplo: `https://testExample-idDomain.pbcs.us1.oraclecloud.com`
   - `emailaddresses` con una lista separada por comas de direcciones de correo electrónico a las que se debe enviar una notificación cuando se complete el mantenimiento diario.
3. Utilice la pantalla de Groovy en un proceso de negocio de Cloud EPM o automatice la ejecución del script mediante runBusinessRule. Para obtener más información, consulte estas fuentes de información:
   - Ejecución de comandos sin instalar EPM Automate
   - Uso de reglas Groovy en *Administración de Planning*

Copia de una instantánea en o de Oracle Object Storage

Este tema contiene scripts de ejemplo para realizar estas tareas:

- Copie `Artifact Snapshot` (la instantánea de mantenimiento) de Oracle Fusion Cloud Enterprise Performance Management en un cubo de Oracle Object Storage y cámbiele el nombre. Para ello, agregue la fecha en que se ha copiado la instantánea.
- Copie una instantánea de copia de seguridad de un cubo de Oracle Object Storage en Cloud EPM.

En los scripts de esta sección se asume que haya creado un cubo en Oracle Object Storage para incluir la instantánea. Antes de ejecutar estos scripts, personalícelos para usarlos actualizando estos marcadores de posición:

Table 3-1 Parámetros y sus valores

Marcador de posición	Valor esperado
`JAVA_HOME`	Directorio donde está instalado el JDK que usa EPM Automate. **Ejemplo:** `./home/JDK/bin`
`epmautomateExe`	Directorio donde está instalado EPM Automate. **Ejemplo:** `./home/utils/EPMAutomate/bin`
`cloudServiceUser`	ID de usuario de un administrador del servicio de Cloud EPM. **Ejemplo:** `John.doe@example.com`
`cloudServicePassword`	Contraseña del administrador de servicio o ubicación del archivo de contraseña. Si la contraseña contiene caracteres especiales, consulte Manejo de caracteres especiales. **Ejemplo:** `ex_PWD_213`
`cloudServiceUrl`	URL del entorno de Cloud EPM desde la que se copia `Artifact Snapshot`. **Ejemplo:** `https//test-cloud-id_Dom.pbcs.us1.oraclecloud.com`
`objectStorageUser`	ID de un usuario de Oracle Object Storage. Para copiar una instantánea en Object Storage, este usuario debe tener acceso de escritura para el cubo en el que se copie la instantánea. Para copiar una instantánea de Object Storage, este usuario debe tener acceso de lectura para el cubo del que se copie la instantánea. **Ejemplo:** `jDoe`
`objectStoragePassword`	Contraseña de `objectStorageUser`. **Ejemplo:** `example_PWD`
`objectStorageBucketUrl`	URL del cubo de Oracle Object Storage donde se va a copiar la instantánea. Consulte estas fuentes de información para el formato de la URL: • copyToObjectStorage • copyFromObjectStorage **Ejemplo:** `https//swiftobjectstorage.us-ashburn-1.oraclecloud.com/v1/axaxnpcrorw5/bucket-20210301-1359`
`snapshot`	Nombre del a instantánea que desea copiar del cubo de Oracle Object Storage. **Ejemplo:** `Artifact Snapshot20210429.zip`

Script de ejemplo de EPM Automate para copiar una instantánea de Cloud EPM en Oracle Object Storage

Personalice y ejecute este script para cambiar el nombre y, a continuación, copiar `Artifact Snapshot` de Cloud EPM en un cubo de Oracle Object Storage.

```
#!/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 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
```

Script de ejemplo de EPM Automate para copiar una instantánea de Oracle Object Storage en Cloud EPM

Personalice y ejecute este script para copiar `Artifact Snapshot` de una fecha específica de un cubo de Oracle Object Storage en 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
```

Creación de usuarios y asignación a roles predefinidos

Utilice los scripts de esta sección para crear usuarios y asignarlos a roles predefinidos en el dominio de identidad.

Estos scripts utilizan comandos de EPM Automate para completar las siguientes actividades:

- Iniciar sesión en un entorno como administrador del servicio con el rol Administrador de dominio de identidad.
- Exportar grupos e información de pertenencia desde el entorno para volver a generar una instantánea que especifique; por ejemplo, `example_snapshot.zip`. Se supone que previamente ha exportado los artefactos de Grupos y pertenencia mediante Migration para crear esta instantánea.

- Descargue la instantánea (`example_snapshot.zip`) en un directorio local.
- Suprima la instantánea (`example_snapshot.zip`) del entorno.
- Cierre la sesión en el entorno.
- Realizar operaciones para las que ha agregado un código personalizado. Dichas operaciones pueden incluir:
  - Extracción de contenido de `example_snapshot.zip`
  - Anexión de información de nuevo usuario a `HSS-Shared Services\resource\External Directory\Users.csv` con el formato Nombre,Apellidos,Correo electrónico,Inicio de sesión de usuario. Para obtener información detallada, consulte Importación de un lote de cuentas de usuarios en *Getting Started with Oracle Cloud*.
  - Anexión de información sobre asignaciones de roles de nuevos usuarios (con formato Nombre,Apellidos,Correo electrónico,Inicio de sesión de usuario) a los archivos de roles adecuados en `HSS-Shared Services\resource\External Directory\Roles\`. Por ejemplo, la asignación al rol de administrador del servicio se debe anexar a `<service_name> Service Administrator.csv`, mientras que las asignaciones al rol de visor se deben anexar a `HSS-Shared Services\resource\External Directory\Roles\<service_name> Viewer.csv`. Para obtener información detallada, consulte Asignación de un rol a muchos usuarios en *Getting Started with Oracle Cloud*.
  - Nueva creación de la instantánea con los archivos actualizados mediante la compresión del directorio `HSS-Shared Services` y su contenido.
- Inicie sesión en el entorno como administrador del servicio que también cuenta con el rol de administrador del dominio de identidad.
- Cargue el archivo `example_snapshot.zip` modificado en el entorno.
- Importe `example_snapshot.zip` en el entorno.
- Suprima el archivo `example_snapshot.zip` cargado del entorno.
- Cierre la sesión.

Script de Windows de ejemplo

Cree un archivo denominado `createUsersAndAssignRoles.ps1` copiando el siguiente script. Almacénelo en un directorio 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
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
```

Script de Linux/UNIX de ejemplo

Cree un archivo denominado `createUsersAndAssignRoles.sh` copiando el siguiente script. Almacénelo en un directorio 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
```

Archivo input.properties de ejemplo

Para ejecutar los scripts `createUsersAndAssignRoles`, cree el archivo `input.properties` y actualícelo con información para el entorno. Guarde el archivo en el directorio en el que está almacenado `createUsersAndAssignRoles.ps1` o `createUsersAndAssignRoles.sh`.

Windows

```
username=exampleAdmin
passwordfile=examplePassword.epw
serviceURL=exampleURL
snapshotName=SNAPSHOT_NAME
```

```
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
```

Tabla 3-2 Parámetros de input.properties

Parámetro	Descripción
`javahome`	Ubicación `JAVA_HOME`. Solo para Linux/UNIX.
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`). Solo para Linux/UNIX.
`username`	Nombre de usuario de un administrador del servicio que también tiene asignado el rol de administrador del dominio de identidad.
`password`	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.
`serviceURL`	URL del entorno a partir del cual desea generar la instantánea.
`snapshotName`	Nombre para la instantánea que desea generar. Se supone que previamente ha exportado los artefactos de Grupos y pertenencia mediante Migration para crear esta instantánea.
`userspassword`	Contraseña inicial para nuevos usuarios.
`resetPassword`	Determina si los nuevos usuarios deben restablecer la contraseña en el primer inicio de sesión. Defina este valor en `true` para forzar a los nuevos usuarios a cambiar su contraseña cuando inician sesión por primera vez.

Ejecución del script

1. Cree `createUsersAndAssignRoles.ps1` o `createUsersAndAssignRoles.sh` copiando el script de una sección anterior.
2. Agregue código personalizado para realizar las siguientes operaciones:
   - Extraer el contenido de la instantánea
   - Anexar información de nuevo usuario a `HSS-Shared Services\resource\External Directory\Users.csv`.
   - De manera opcional, anexe información sobre asignaciones de roles de nuevos usuarios (con formato Nombre,Apellidos,Correo electrónico,Inicio de sesión de usuario) a los archivos de roles adecuados en `HSS-Shared Services\resource\External Directory\Roles\`.
   - Volver a crear la instantánea con los archivos actualizados.

3. Cree el archivo `input.properties` y guárdelo en el directorio en el que está ubicado el script `createUsersAndAssignRoles`. El contenido de este archivo difiere según su sistema operativo. Consulte Archivo input.properties de ejemplo.
   Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar el script.
4. Inicie el script.
   - **Windows PowerShell:** ejecute `createUsersAndAssignRoles.ps1`.
   - **Linux/UNIX:** ejecute `./createUsersAndAssignRoles.sh`.

Recuento del número de usuarios con licencia (usuarios asignados a roles)

Utilice el script de esta sección para generar el Informe de asignación de roles para contar el número de usuarios de un entorno.

Cree `provisionedUsersCount.bat` mediante la copia del siguiente script.

> **Nota:**
>
> - Los parámetros de entrada para ejecutar `provisionedUsersCount.bat` son `username`, `password/password_file`, `service_url` y `report_file_name`. Por ejemplo, en el símbolo del sistema, introduzca un comando similar al siguiente: `provisionedUsersCount.bat jdoe password.epw https://example.oraclecloud.com myRole_assign.CSV`
> - Si la contraseña contiene caracteres especiales, consulte Manejo de caracteres especiales.

```
@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
  )
```

```
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
         )
    )

    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%.
)
```

Creación de informes de auditoría de usuarios asignados a roles

Utilice los scripts de esta sección para automatizar el proceso de creación de un informe de auditoría para usuarios asignados a roles predefinidos en un entorno y, opcionalmente, enviarlo por correo electrónico a un destinatario.

En este informe de auditoría se muestran los usuarios asignados a roles o grupos predefinidos que hayan cambiado desde la última vez que se generó el informe. Para crear un informe de auditoría diario, ejecute este script a diario.

Cree `provisioningAuditReport.bat` copiando el siguiente script. Este script por lotes de envoltorio llama al script PowerShell `provisioningAuditReport.ps1`, el código de origen para el que se proporciona más adelante en este escenario.

> **Nota:**
>
> - Los parámetros de entrada para ejecutar `provisioningAuditReport.bat` son: `username`, `password` o `password_file`, `service_url` y `report_email_to_address` (opcional, solo necesario si desea enviar el informe a una dirección de correo electrónico).
> - Si la contraseña contiene caracteres especiales, consulte Manejo de caracteres especiales.

```
@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
  )
```

```
PowerShell.exe -File provisioningAuditReport.ps1 %*
```

`provisioningAuditReport.bat` llama a `provisioningAuditReport.ps1`, que se crea copiando el siguiente script.

Con `provisioningAuditReport.ps1` se crea el informe de auditoría. Colóquelo en el mismo directorio en el que se encuentra `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
```

```
{
  $op=$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"
```

```
    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
```

```
}

Init
EchoAndLogMessage "Starting EPMAutomate provisioning audit reporting"
RunEpmAutomateCommands
CreateProvisionReportTempFile
CreateUniqueElementsFile
CheckBaselineAndCreateAuditReport
PostProcess
EchoAndLogMessage "EPMAutomate provisioning audit reporting completed"
```

Creación de informe de auditoría de asignación de roles y revocación

Utilice el script de PowerShell de esta sección para automatizar el proceso de creación de un informe de auditoría que detalla la asignación de roles y la revocación de roles en un entorno.

Cree `AuditReportRoleAssignment.bat` copiando el siguiente script. Este script por lotes de contenedor llama al script PowerShell `AuditReportRoleAssignment.ps1`, el código de origen para el que se proporciona más adelante en este escenario.

> **Nota:**
>
> - Los parámetros de entrada para ejecutar `AuditReportRoleAssignment.bat` son `username`, `password` o `password_file`, y `service_url`.
> - Si la contraseña contiene caracteres especiales, consulte Manejo de caracteres especiales.

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 %*
```

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
```

```
    }
}

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
```

```
    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"
```

Enmascaramiento de registros de acceso e informe de actividades para cumplir las normas de privacidad

Utilice los scripts de esta sección para automatizar el proceso de enmascaramiento de información en el Informe de actividad o los Registros de acceso para cumplir con las leyes de privacidad y para, si lo desea, enviar por correo electrónico el informe a un destinatario.

Debido a las estrictas leyes de privacidad vigentes en algunos países, la información disponible en los informes de actividad y los registros de acceso puede ocultarse a los administradores de servicio para proteger la privacidad de los usuarios.

Utilice `anonymizeData.bat` para enmascarar información en el informe de actividad o los registros de acceso para cumplir las leyes de privacidad y, opcionalmente, enviarlo por correo electrónico. Para enmascarar la información, programe este script o una variación mediante el programador de Windows para que se ejecute todos los días una vez que finalice el proceso de mantenimiento diario de cada entorno.

Utilice las siguientes fuentes de información:

- Uso de informes de actividad y registros de acceso para supervisar el uso en *Guía de introducción para administradores*
- Automatización de la ejecución de scripts

Para crear manualmente `anonymizeData.bat`, copie el script de Windows proporcionado en el siguiente procedimiento y prográmelo mediante el programador de Windows. Puede crear y

ejecutar scripts similares para otras plataformas en caso de que no utilice Windows para la programación.

`anonymizeData.bat` es un script contenedor que ejecuta el script `anonymizeData.ps1`, el cual puede crear y actualizar tal y como se explica en el procedimiento siguiente.

Si la contraseña contiene caracteres especiales, consulte Manejo de caracteres especiales

1. Cree un archivo por lotes (`BAT`) llamado `anonymizeData.bat` que contenga el siguiente script y guárdelo en una ubicación apropiada, por ejemplo, `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. Cree un script PowerShell (`PS1`) llamado `anonymizeData.ps1` que contenga el siguiente script y guárdelo en una ubicación apropiada, por ejemplo, `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
}
```

```
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]
```

```
    $user=$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 {
```

```
                    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
```

```
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. Con el programador de Windows, programe `anonymizeData.bat`. Consulte Automatización de la ejecución de scripts para obtener pasos detallados.

   Debe proporcionar los siguientes valores de parámetros para ejecutar `anonymizeData.bat`.

   - Nombre de usuario de un administrador del servicio.
   - Contraseña del administrador del servicio o la ubicación en la que está disponible el archivo de contraseña cifrado.
   - URL del entorno del servicio en el que se van a enmascarar los registros de acceso e informes de actividad.
   - **Opcional:** dirección de correo electrónico a la que se va a enviar el informe. El informe se envía por correo electrónico solo si se especifica este valor.

Automatización de descargas de informes de actividad en un equipo local

Utilice el script de esta sección para automatizar la descarga de informes de actividad de un entorno en un equipo local.

Utilice `syncAprReports.bat` para descargar informes de actividad. Puede programar el archivo por lotes mediante el programador de Windows para automatizar la descarga de informes de actividad. Consulte Uso de informes de actividad y registros de acceso para supervisar el uso en *Guía de introducción para administradores* para obtener información detallada sobre informes de actividad.

Cree manualmente `syncAprReports.bat` mediante la copia del script proporcionado en el siguiente procedimiento y la actualización de los parámetros de conexión. Este script

comprueba el entorno y descarga solo los informes que sean más recientes que los que están disponibles en el directorio de descarga del equipo local.

> **Nota:**
>
> - El script se ejecutará desde un equipo de Windows únicamente.
> - Con este script no se descarga el informe de actividad de comentarios, que se genera cuando el usuario envía comentarios.
> - Si la contraseña contiene caracteres especiales, consulte Manejo de caracteres especiales

1. Cree un archivo por lotes (`.BAT`) denominado `syncAprReports.bat` que contenga el siguiente script y guárdelo en una ubicación adecuada (por ejemplo, `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
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!" equ ".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!
                            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. Modifique `syncAprReports.bat` para definir los valores de los parámetros en la siguiente tabla. Estos valores se utilizan para acceder al entorno y descargar informes de actividad.

Tabla 3-3 Valores de parámetro que se van a incluir en `syncAprReports.bat`

Parámetro	Valor esperado
`set apr_dir=%1`	Especifique un directorio existente en el que se descargarán los informes de actividad. **Ejemplo:** `set apr_dir="C:\Oracle\apr"`

Tabla 3-3 (Continuación) Valores de parámetro que se van a incluir en `syncAprReports.bat`

Parámetro	Valor esperado
`set username=%2`	Un nombre de usuario de Oracle Fusion Cloud Enterprise Performance Management que se va a utilizar para iniciar sesión en el entorno y descargar los informes de actividad. **Ejemplo:** `set username="ServiceAdmin"`

`set password=%3`	El nombre y la ubicación del archivo que almacena la contraseña cifrada del usuario especificado por la variable `username`. También puede especificar la contraseña de texto sin formato del usuario (no recomendado). Consulte el comando encrypt para obtener información sobre la creación de un archivo de contraseñas cifradas.
	Ejemplos:
	`set password="C:\mySecuredir\password.epw"`
	`set password="Ex@mple1"`

`set url=%4`	La URL del entorno. **Ejemplo:** `set url="https://test-example.stg-pbcs.us1.oraclecloud.com"`

3. Con el programador de Windows, programe `syncAprReports.bat`. Consulte Automatización de la ejecución de scripts para obtener pasos detallados.

Descarga de logs de acceso desde un entorno

Utilice el script de esta sección para automatizar el proceso de descarga de registros de acceso de un entorno en un equipo local.

Puede programar `syncAccessLog.bat` utilizando el programador de Windows para automatizar la descarga de los archivos de registro. Consulte Visualización y descarga de informes de actividad y registros de acceso en *Guía de introducción para administradores* para conocer los procedimientos de descarga de registros de acceso mediante la administración de aplicaciones.

El siguiente script comprueba el entorno y descarga solo los archivos de registro que son más recientes que los disponibles en el directorio de descarga del equipo local. Este script es de Windows; puede crear un script de shell similar para entornos de Linux/UNIX.

1. Cree un archivo por lotes (.BAT) llamado `syncAccessLog.bat` que contenga el siguiente script y guárdelo en una ubicación apropiada, por ejemplo, `C:\automate_scripts`.

> **Nota:**
>
> Si su contraseña contiene caracteres especiales, consulte Manejo de caracteres especiales.

```
@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="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
)
```

```
: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
```

```
for /L %%n in (1 1 !output_cnt!) DO echo !output[%%n]!
goto :end

:end
cd /D %epmautomate_dir%
endlocal
```

2. Modifique `syncAccessLog.bat` para establecer los valores de los parámetros en la siguiente tabla: Estos valores se utilizan para acceder al entorno y descargar los registros de acceso.

Tabla 3-4 Valores de variable que se van a incluir en `syncAccessLog.bat`

Variable	Valor esperado
`set apr_dir=%1`	Especifique el directorio en el que se van a descargar los registros de acceso. **Ejemplo:** `set apr_dir="C:\Oracle\apr"`
`set username=%2`	Un nombre de usuario de Oracle Fusion Cloud Enterprise Performance Management que se va a utilizar para iniciar sesión en el entorno y descargar los registros de acceso. **Ejemplo:** `set username="ServiceAdmin"`

`set password=%3`	El nombre y la ubicación del archivo que almacena la contraseña cifrada del usuario especificado por la variable `username`. También puede especificar la contraseña de texto sin formato del usuario (no recomendado). Consulte el comando encrypt para obtener información sobre la creación de un archivo de contraseñas cifradas.
	Ejemplos:
	`set password="C:\mySecuredir\password.epw"`
	`set password="P@ssword1"`

`set url=%4`	La URL del entorno. **Ejemplo:** `set url="https://test-cloudpln.pbcs.us1.oraclecloud.com"`

3. Con el programador de Windows, programe `syncAccessLog.bat`. Consulte Automatización de la ejecución de scripts para obtener pasos detallados.

Automatización de la clonación de entornos

Utilice el script de esta sección para automatizar la clonación de entornos.

Cree un archivo de lote (`.bat`) o shell (`.sh`) que contenga un script similar al siguiente para clonar un entorno. Los siguientes scripts de ejemplo gestionan estas actividades:

- Inicie sesión en el entorno de origen.
- Use `Artifact Snapshot` (la instantánea creada durante el último mantenimiento diario del entorno de origen) u otra instantánea disponible en el entorno de origen para convertir el entorno de destino como un clon del origen.
- También puede crear usuarios y sus asignaciones predefinidas de roles de aplicación que coincidan con las del entorno de origen.
- También puede cambiar la hora de inicio del mantenimiento diario para que coincida con la del entorno de origen.

- Cierre la sesión.

Para obtener información detallada sobre el proceso de clonación, consulte "Clonación de entornos de Cloud EPM " en *Administración de la migración*.

Consulte Automatización de la ejecución de scripts para obtener información sobre la programación del script con Programador de tareas de Windows.

Windows

1. Cree un archivo por lotes (.BAT) llamado `cloneEnvironment.bat` que contenga el siguiente script y guárdelo en una ubicación apropiada, por ejemplo, `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%
```

2. Modifique `cloneEnvironment.bat` para establecer los valores de estos parámetros:

Tabla 3-5 Parámetros que establecer en cloneEnvironment.bat

Parámetro	Descripción
`usersandpredefinedroles`	Establezca el valor de este parámetro en `true` para clonar los usuarios y sus asignaciones de roles de aplicación predefinidas. Para clonar asignaciones de usuarios y roles, el usuario que ejecuta el script debe tener el rol de administrador del servicio y el administrador del dominio de identidad en el entorno de destino.
`snapshotname`	Nombre de la instantánea del entorno de origen que se debe usar para la clonación.
`dailymaintenancestarttime`	Establezca el valor de este parámetro en `true` para cambiar la hora de inicio del mantenimiento diario del entorno clonado en la del entorno de origen. Establezca este valor en `false` para conservar la hora de inicio del mantenimiento diario actual del entorno clonado.

3. Cree un script de PowerShell denominado `cloneEnvironment.ps1` que contenga el siguiente script y guárdelo en el directorio en el que haya guardado `cloneEnvironment.bat`, por ejemplo, `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. Ejecute `cloneEnvironment.bat` con este comando:

```
cloneEnvironment.bat "SOURCE USERNAME" "SOURCE PASSWORD FILE" "SOURCE URL"
"TARGET USERNAME" "TARGET PASSWORD FILE" "TARGET URL"
```

Por ejemplo:

```
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. Cree el script de shell denominado `cloneEnvironment.sh` que contenga el siguiente script y guárdelo en una ubicación apropiada.

```
#!/bin/bash

# Update the following parameters
# -------------------------------
epmautomatescript=/home/user1/epmautomate/bin/epmautomate.sh
javahome=/home/user1/jdk1.8.0_191/
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. Modifique `cloneEnvironment.sh` para establecer los valores de estos parámetros:

Tabla 3-6 Parámetros que establecer en cloneEnvironment.sh

Parámetro	Descripción
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`).
`javahome`	Ubicación JAVA_HOME.

Tabla 3-6 (Continuación) Parámetros que establecer en cloneEnvironment.sh

Parámetro	Descripción
`usersandpredefinedroles`	Establezca el valor de este parámetro en `true` para clonar los usuarios y sus asignaciones de roles de aplicación predefinidas. Para clonar asignaciones de usuarios y roles, el usuario que ejecuta el script debe tener el rol de administrador del servicio y el administrador del dominio de identidad en el entorno de destino.
`snapshotname`	Nombre de la instantánea del entorno de origen que se debe usar para la clonación.
`dailymaintenancestarttime`	Establezca el valor de este parámetro en `true` para cambiar la hora de inicio del mantenimiento diario del entorno clonado en la del entorno de origen. Establezca este valor en `false` para conservar la hora de inicio del mantenimiento diario actual del entorno clonado.

3. Ejecute `cloneEnvironment.sh`.

```
./cloneEnvironment.sh "SOURCE USERNAME" "SOURCE PASSWORD FILE" "SOURCE
URL" "TARGET USERNAME" "TARGET PASSWORD FILE" "TARGET URL"
```

Por ejemplo:

```
./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.
```

Clonación diaria del entorno principal al entorno de espera una vez completado el mantenimiento diario en el entorno principal

Para mantener el entorno en espera actualizado con el entorno principal, utilice estos scripts para clonar el entorno principal de Oracle Fusion Cloud Enterprise Performance Management en el entorno en espera justo después de que haya terminado el mantenimiento diario en el entorno principal.

Estos scripts personalizados identifican si el mantenimiento diario programado para el día se ha realizado y, a continuación, clonan el entorno.

Script de Windows

Cree `dailyclone.ps1` copiando el siguiente script de PowerShell.

```
# Clone Environment script
#
# Update the following parameters
# -------------------------------
$users_and_predefined_roles="false"
$daily_maintenance_start_time="true"
$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")
    }
}

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 de Linux/UNIX

Cree `dailyclone.sh` copiando el siguiente script.

```
#!/bin/bash

# Update the following parameters
# -------------------------------
epmautomatescript="LOCATION_EPM_AUTOMATE_EXECUTABLE"
javahome="LOCATION_JAVA_HOME"
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 [[ ${filelist} == *"Artifact Snapshot"* ]]
    then
        command_del=$(${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 ..."
${epmautomatescript} logout
echo "Clone environment script processing has completed."
```

Script de Groovy

Cree el script Groovy `dailyclone` copiando el siguiente código.

```
// 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()
    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)
        }
    }
}
```

```
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.")
```

Ejecución del script

1. Cree el script `dailyclone.ps1`, `dailyclone.sh` o el script Groovy `dailyclone` copiando un script de una de las secciones anteriores.
2. Actualice los scripts:
   Actualización para `dailyclone.sh`:
   - `epmautomatescript` con la ubicación del ejecutable de EPM Automate. Ejemplo: `epmautomatescript="/home/utils/EPMAutomate/bin/epmautomate.sh"`
   - `javahome` con el directorio donde se instalar el JDK que utiliza EPM Automate. Por ejemplo: `"/home/user1/jdk1.8.0_191"`

   Actualización para `dailyclone.ps1` y `dailyclone.sh`:

- `users_and_predefined_roles`: establezca este valor en `true` para clonar usuarios y sus asignaciones de rol predefinido (los grupos de Control de acceso siempre se clonan).
- `data_management`: establezca este valor en `false` para no clonar registros de Data Integration. Tenga en cuenta que los registros de Data Integration solo se pueden clonar si tanto el entorno de origen como el de destino están en la misma actualización mensual o el entorno de destino está en una actualización más actual que el entorno de origen. Por ejemplo, puede clonar registros del entorno 22.01 de Data Management a otro entorno 22.01 o solo a un entorno de 22.02.
  Se ignora para entornos de Narrative Reporting y Oracle Fusion Cloud Enterprise Data Management.
- `app_audit`: establezca este valor en `false` si no desea clonar los datos de auditoría de aplicación para las aplicaciones de Planning, FreeForm y Enterprise Profitability and Cost Management.
  La información de auditoría de Financial Consolidation and Close y Tax Reporting se clona siempre.
- `job_console`: establezca este valor en `false` si no desea clonar los datos de la consola de trabajos.
- `stored_snapshots_and_files`: establezca este valor en `true` si desea clonar el contenido de las carpetas de nivel superior de la bandeja de entrada y el buzón de salida (las subcarpetas no se clonan nunca) del entorno de origen.
- `daily_maintenance_start_time`: establezca este valor en `false` si no desea restablecer la hora de inicio del mantenimiento del entorno clonado en la del entorno de origen.

Actualizaciones para el script Groovy `dailyclone`:

- `source_username` es el nombre de usuario de un administrador del servicio. Se necesita el rol Administrador de dominio de identidad para clonar usuarios y roles predefinidos.
- `source_password` es la contraseña del usuario identificado por *`SOURCE_USERNAME`*.
- `source_url` es la URL del entorno que desea clonar.
- `target_username` es el nombre de usuario de un administrador del servicio. Se necesita el rol Administrador de dominio de identidad para clonar usuarios y roles predefinidos.
- `target_password` es la contraseña del usuario identificado por *`TARGET_USERNAME`*.
- `target_URL` es la URL del entorno de destino.

3. Ejecute `dailyclone.ps1`, `dailyclone.sh` o el script Groovy `dailyclone`:
   Windows
   - En una ventana de comandos, navegue a la carpeta en la que se almacena `dailyclone.ps1`.
   - Ejecute el comando: `./dailyclone.ps1` *`SOURCE_USERNAME SOURCE_PASSWORD SOURCE_URL TARGET_USERNAME TARGET_PASSWORD TARGET_URL`*. Consulte la tabla siguiente para ver una descripción de estos valores.

Table 3-7 Descripción de los parámetros

Parámetro	Descripción
SOURCE_USERNAME	Nombre de usuario de un administrador del servicio. Además, este usuario debe tener el rol Administrador de dominio de identidad para clonar usuarios y roles predefinidos.
SOURCE_PASSWORD	Contraseña del usuario identificado por *SOURCE_USERNAME*.
SOURCE_URL	URL del entorno que desea clonar.
TARGET_USERNAME	Nombre de usuario de un administrador del servicio en el entorno de destino. Además, este usuario debe tener el rol Administrador de dominio de identidad para clonar usuarios y roles predefinidos.
TARGET_PASSWORD	Contraseña del usuario identificado por *TARGET_USERNAME*.
TARGET_URL	URL del entorno de destino.

Linux/UNIX

- En un shell de UNIX o Linux, navegue al directorio en el que se almacena `dailyclone.sh`.
- Ejecute el comando: `./dailyclone.sh` *SOURCE_USERNAME SOURCE_PASSWORD SOURCE_URL TARGET_USERNAME TARGET_PASSWORD TARGET_URL*. Consulte la tabla anterior para ver una descripción de estos valores.

Groovy
Consulte Ejecución de comandos sin instalar EPM Automate.

Eliminación de archivos innecesarios desde un entorno

Utilice estos scripts para eliminar archivos innecesarios de un entorno.

Estos scripts realizan los siguientes pasos:

- Iniciar sesión en el entorno.
- Mostrar los archivos y las instantáneas en el entorno.
- Suprimir los archivos especificados en `input.properties`.
- Cerrar sesión.

Script de Windows de ejemplo

Cree un archivo denominado `removeUnnecessaryFiles.ps1` copiando el siguiente script. Almacénelo en un directorio 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}
```

```
epmautomate deletefile ${file2}
epmautomate logout
```

Script de Linux/UNIX de ejemplo

Cree un archivo denominado `removeUnnecessaryFiles.sh` copiando el siguiente script. Almacénelo en un directorio local.

```
#!/bin/bash
. ./input.properties
export JAVA_HOME=${javahome}
${epmautomatescript} login "${username}" "${passwordfile}" "${serviceURL}"
${epmautomatescript} listfiles
${epmautomatescript} deletefile "${file1}"
${epmautomatescript} deletefile "${file2}"
${epmautomatescript} logout
```

Creación del archivo input.properties

Para ejecutar los scripts `removeUnnecessaryFiles`, cree el archivo `input.properties` y actualícelo con información para el entorno. Guarde el archivo en el directorio en el que está almacenado `removeUnnecessaryFiles.ps1` o `removeUnnecessaryFiles.sh`.

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
```

Tabla 3-8 Parámetros de input.properties

Parámetro	Descripción
`javahome`	Ubicación `JAVA_HOME`. Solo para Linux/UNIX.
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`). Solo para Linux/UNIX.
`username`	Nombre de usuario de un administrador del servicio que también tiene asignado el rol de administrador del dominio de identidad.
`password`	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.

Tabla 3-8 (Continuación) Parámetros de input.properties

Parámetro	Descripción
`serviceURL`	URL del entorno a partir del cual desea generar la instantánea.
`file1` y `file2`	Nombre de un archivo o una instantánea para suprimirlo del entorno. Si el archivo no está en la bandeja de salida, especifique la ruta y el nombre del archivo.

Ejecución del script

1. Cree `removeUnnecessaryFiles.ps1` o `removeUnnecessaryFiles.sh` copiando el script de una sección anterior.
2. Cree el archivo `input.properties` y guárdelo en el directorio en el que está ubicado el script `removeUnnecessaryFiles`. El contenido de este archivo difiere según su sistema operativo. Consulte Creación del archivo input.properties.
   Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar el script.
3. Inicie el script.
   - **Windows PowerShell:** ejecute `removeUnnecessaryFiles.ps1`.
   - **Linux/UNIX:** ejecute `./removeUnnecessaryFiles.sh`.

Búsqueda y descarga de archivos de un entorno

Utilice el script de ejemplo de esta sección para automatizar el proceso de descarga de uno o más archivos des de un entorno de Oracle Fusion Cloud Enterprise Performance Management utilizando una cadena de texto como comodín.

El siguiente script le permite que coincida la cadena que se especifica como el valor del parámetro `NOMBRE DE ARCHIVO` con los nombres de archivo mostrados mediante el comando `listfile` y, a continuación, podrá descargar automáticamente los archivos que coinciden con la cadena.

Asegúrese de asignar la cadena de búsqueda adecuada al parámetro `FILENAME`. Por ejemplo, `FILENAME="Scheduler Output/epm"` coincidirá con la cadena `Scheduler Output/epm` en los nombres de archivo de la salida del comando `listfiles` de su entorno para identificar los archivos que se van a descargar.

Los parámetros de entrada para ejecutar este script son `username`, `password` o `password_file`, y `service_url`.

> **Nota:**
>
> Si la contraseña contiene caracteres especiales, consulte Manejo de caracteres especiales.

Windows

```
@echo off
    setlocal EnableExtensions EnableDelayedExpansion
    set USERNAME="username"
    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
```

Nueva creación de un entorno antiguo de Cloud EPM para auditorías

Utilice el script de esta sección para crear una solución de autoservicio para mantener una biblioteca actualizada de instantáneas para su entorno de Oracle Fusion Cloud Enterprise Performance Management. Necesita un entorno dedicado al propósito de actualizar y mantener una biblioteca de instantáneas actualizadas.

Cloud EPM soporta la compatibilidad con instantáneas solo para un ciclo mensual. Puede migrar instantáneas de mantenimiento del entorno de prueba al de producción y viceversa. Sin embargo, es posible que los requerimientos de auditoría de algunos clientes necesiten restaurar instantáneas de varios años en el entorno más reciente, y acceder a la aplicación en un periodo de tiempo breve.

Debe programar este script para que se ejecute una vez al mes a fin de convertir las instantáneas disponibles y hacerlas compatibles con el nivel de parche más reciente de Cloud EPM . Oracle recomienda que ejecute un script después del tercer viernes del mes para asegurarse de que todos los problemas del entorno de producción se han resuelto.

> **Nota:**
>
> No puede utilizar este script para actualizar las instantáneas de Narrative Reporting, Account Reconciliation y Oracle Fusion Cloud Enterprise Data Management.

Funcionamiento del script

Por cada instantánea que el cliente almacena, el script actualizado realiza estas tareas mediante el uso de EPM Automate:

1. Con la información del archivo `input.properties`, se conecta a un entorno.
2. Utiliza el comando `recreate` volver a crear el entorno.
3. Importa la instantánea en el entorno.
4. Ejecuta el mantenimiento diario en un entorno, lo que convierte la instantánea al formato compatible con el nivel de parche de Cloud EPM actual.
5. Descarga `Artifact Snapshot` (la instantánea de mantenimiento) en una carpeta. Si ha recreado un entorno 18.05 cargando instantáneas de `snapshots/18.05`, `Artifact Snapshot` se descarga en `snapshots/18.06`.
6. Envía por correo electrónico el resultado de la nueva creación de entornos antiguos a una dirección de correo electrónico si se especifica.

Ejecución del script

1. Cree el archivo `input.properties` y actualícelo con información de su entorno. Guarde el archivo en un directorio local. Este directorio lo denominaremos `parentsnapshotdirectory`. El contenido de este archivo difiere según su sistema operativo.
   Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar los scripts.
2. Cree el script `upgradeSnapshots.ps1` (Windows PowerShell) o `upgradeSnapshots.sh` (Linux/UNIX) y guárdelo en `parentsnapshotdirectory`, donde se ubica `input.properties`.
3. Cree un subdirectorio, por ejemplo, `snapshots`, en `parentsnapshotdirectory`.
4. En el directorio que ha creado en el paso anterior (`snapshots`), cree un subdirectorio para la instantánea mensual que desea convertir para hacerla compatible con el nivel de parche de Cloud EPM actual. Asigne un nombre al directorio con el formato `YY.MM`; por ejemplo, `18.05` para el directorio en el que se almacenan las instantáneas de mayo de 2018.
5. Copie las instantáneas en el subdirectorio adecuado. Por ejemplo, copie las instantáneas de mayo de 2018 en `snapshots/18.05`.
6. Inicie el script.
   - Linux/UNIX: ejecute `./upgradeSnapshots.sh`.
   - Windows PowerShell: ejecute `upgradeSnapshots.ps1`.

Windows

Cree los scripts `input.properties` y `upgradeSnapshots.ps1` copiando los scripts de esta sección.

Creación de `input.properties`

```
username=exampleAdmin
userpassword=examplePassword
serviceurl=exapleURL
proxyserverusername=proxyServerUserName
proxyserverpassword=proxyPassword
proxyserverdomain=proxyDoamin
parentsnapshotdirectory=C:/some_directory/snapshots
emailtoaddress=exampleAdmin@oracle.com
```

Actualización de `input.properties`

> **Nota:**
>
> Si `authentication at proxy server` no está activado para su entorno de red de Windows, elimine las propiedades `proxyserverusername`, `proxyserverpassword` y `proxyserverdomain` del archivo `input.properties`.

Tabla 3-9 Parámetros de input.properties

Parámetro	Descripción
`username`	Nombre de usuario de un administrador del servicio.
`userpassword`	Contraseña del administrador de servicio.
`serviceurl`	URL del entorno que se usa en esta actividad.
`proxyserverusername`	Nombre de usuario utilizado para autentificar una sesión segura con el servidor proxy que controla el acceso a Internet.
`proxyserverpassword`	Contraseña para autentificar el usuario con el servidor proxy.
`proxyserverdomain`	Nombre del dominio definido para el servidor proxy.
`parentsnapshotdirectory`	Ruta de acceso absoluta del directorio que se utilizará como el directorio padre del directorio en el que se almacenan las instantáneas que se van a procesar. Utilice barras inclinadas (/) como separadores de directorio.
`emailtoaddress`	Opcionalmente, dirección de correo electrónico a la que se van a enviar los resultados de la nueva creación de entornos antiguos. El resultado se envía por correo electrónico solo si se especifica este valor. **Ejemplo:** `john.doe@example.com`

> **Nota:**
>
> Si su contraseña contiene caracteres especiales, consulte Manejo de caracteres especiales.

Creación de `upgradeSnapshots.ps1`

Utilice este script de ejemplo para crear `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
$parentsnapshotdirectory="$($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
{
    $op=$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
    }

    $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
    }
}

#----- 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

Cree `upgradeSnapshots.sh` y `input.properties` copiando los siguientes scripts.

Creación de `input.properties` para Linux/UNIX

> **Nota:**
>
> Si su red no está configurada para utilizar un servidor proxy para acceder a Internet, elimine las propiedades `proxyserverusername`, `proxyserverpassword` y `proxyserverdomain` del archivo `input.properties`.

```
username=exampleAdmin
userpassword=examplePassword
serviceurl=exapleURL
proxyserverusername=
proxyserverpassword=
proxyserverdomain=
jdkdir=/home/user1/jdk160_35
epmautomatescript=/home/exampleAdmin/epmautomate/bin/epmautomate.sh
parentsnapshotdirectory=/home/exampleAdmin/some_directory/snapshots
emailtoaddress=exampleAdmin@oracle.com
```

Actualización de `input.properties`

Tabla 3-10 Parámetros de input.properties

Parámetro	Descripción
`username`	Nombre de usuario de un administrador del servicio.
`userpassword`	Contraseña del administrador de servicio.
`serviceurl`	URL del entorno que se usa en esta actividad.
`proxyserverusername`	Nombre de usuario utilizado para autentificar una sesión segura con el servidor proxy que controla el acceso a Internet.
`proxyserverpassword`	Contraseña para autentificar el usuario con el servidor proxy.
`proxyserverdomain`	Nombre del dominio definido para el servidor proxy.
`jdkdir`	Ubicación `JAVA_HOME`.
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`).
`parentsnapshotdirectory`	Ruta de acceso absoluta del directorio que se utilizará como el directorio padre del directorio en el que se almacena la instantánea que se va a procesar.
`emailtoaddress`	Opcionalmente, dirección de correo electrónico a la que se van a enviar los resultados de la nueva creación de entornos antiguos.

> **Nota:**
>
> Si su contraseña contiene caracteres especiales, consulte Manejo de caracteres especiales.

Creación de `upgradeSnapshots.sh`

Utilice este script de ejemplo para crear `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."
    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: ${snapshotdirs}"
    local latestreleasesnapshotdir=$(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`

    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
| 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
```

Automatización de auditoría y conformidad sobre el acceso a la base de datos

Utilice los scripts de PowerShell y Bash Shell de esta sección para usar los comandos de EPM Automate a fin de recopilar datos de auditoría y conformidad sobre el acceso manual a la base de datos.

Puede utilizar estos scripts para completar las siguientes tareas:

- Descargar el Informe de actividad para el día actual
- Analizar el informe para determinar si se ha registrado el acceso manual a la base de datos para el entono
- Crear `./reports/dataAccessAuditReport.txt` relacionado con el directorio desde donde ejecuta el script. El informe muestra la hora de acceso a la base de datos y el comando SQL que se ha ejecutado. Se trata de un archivo acumulado, que muestra la información más reciente en la parte superior. La información disponible incluye:
  - La fecha y hora en la que se generó el informe
  - Detalles del acceso a la base de datos, si están disponibles. El acceso a la base de datos sin una solicitud de servicio y el acceso a la base de datos con solicitud de servicio se muestran en secciones distintas.
    Si no se registra el acceso manual a la base de datos en el Informe de actividad, el informe indica `No SQL statements executed`.
  - Opcionalmente, envíe el informe a una dirección de correo electrónica especificada.

Para automatizar la auditoría y la conformidad del acceso a la base de datos:

1. Copie uno de los scripts de las siguientes secciones en un archivo y guárdelo en el sistema de archivos. Asigne al archivo el nombre `parseActivityReport.ps1` (Windows, consulte Script de PowerShell (parseActivityReport.ps1)) o `parseActivityReport.sh` (Linux/UNIX, consulte Script de shell de Bash (parseActivityReport.sh)).
2. **Solo para Windows:** cree un archivo por lotes denominado `parseActivityReport.bat` copiando el siguiente script en un archivo. Guarde el archivo en el directorio donde está almacenado `parseActivityReport.ps1`.

```
@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. Modifique `parseActivityReport.bat` (Windows) o `parseActivityReport.sh` (Linux/UNIX, consulte) para establecer los valores de los parámetros de la siguiente tabla.

Tabla 3-11 Valores de variable que se deben incluir en los scripts

Variable	Descripción
`epmuser`	Nombre de usuario de un administrador del servicio. **Ejemplos:** Windows: `set epmuser="jDoe"` Linux/UNIX: `epmuser="jDoe"`
`epmpassword`	Contraseña del administrador de servicio o ubicación del archivo de contraseña cifrado. Consulte el comando encrypt para obtener información sobre la creación de un archivo de contraseñas cifradas. Si su contraseña contiene caracteres especiales, consulte Manejo de caracteres especiales. **Ejemplos:** Windows: `set epmpassword = "Example"` Linux/UNIX: `epmpassword="Example"`
`epmurl`	URL del entorno de Oracle Fusion Cloud Enterprise Performance Management. **Ejemplos:** Windows: `set epmurl="https://example.oraclecloud.com"` Linux/UNIX: `epmurl="https://example.oraclecloud.com"`
`report_email_to_address`	Opcionalmente, dirección de correo electrónico a la que se va a enviar el informe. El informe se envía por correo electrónico solo si se especifica este valor. **Ejemplo:** **`john.doe@example.com`**

4. **Solo para `parseActivityReport.sh`:** asegúrese de que los siguientes valores se han definido correctamente para el sistema:
   - *`JAVA_HOME`*
   - Ubicación de `epmautomatescript.sh` mediante la actualización del valor de la directiva `epmautomatescript`
5. Mediante un programador disponible en el sistema operativo, programe `parseActivityReport.bat` (que ejecuta `parseActivityReport.ps1`) o `parseActivityReport.sh` para que se ejecute una vez al día. Consulte Automatización de la ejecución de scripts.

Script de 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
    $global:activityreportfile=$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
"$matchfile"
        (Get-Content "$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 ${reportfile}
        echo "${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) {
        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 de shell de Bash (parseActivityReport.sh)

```
#!/bin/sh

export JAVA_HOME=/scratch/dteHome/autoWork/jdk1.8.0_191
epmautomatescript=/scratch/dteHome/autoWork/epmautomate/19.11.55/bin/
epmautomate.sh

epmuser="<EPM USER>"
epmpwd="<EPM PASSWORD>"
epmurl="<EPM URL>"
reportemailtoaddress="<EMAIL ADDRESS>"

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}
}

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() {
    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
        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
```

Replicación de usuarios y asignación de role predefinidos

Los scripts de esta sección le ayudan a migrar los usuarios y las asignaciones de roles predefinidos de un entorno a otro.

Acerca de los scripts

Utilizas dos scripts distintos: uno para replicar los usuarios a través de dominios de identidad y otro para replicar las asignaciones de roles predefinidos de los usuarios. El orden para ejecutar estos scripts es el siguiente:

- Ejecute el script para replicar usuarios (`replicateusers`) y verifique que todos los usuarios estén creados en el dominio de identidad de destino. El usuario que ejecute este script debe tener roles de Administrador de dominio de identidad y de Administrador del servicio en los dos entornos.
- Ejecute el script para replicar las asignaciones de roles (`replicatepredefinedroles`).

> **Nota:**
>
> - Si las contraseñas contienen caracteres especiales, consulte Manejo de caracteres especiales
> - Los scripts de esta sección funcionan solo para roles predefinidos: Administrador del servicio, Usuario avanzado, Usuario y Visor.

Ejecución de los scripts

Para obtener información sobre la creación de los scripts y los archivos por lotes necesarios, consulte estos temas:

- Replicación de los usuarios de un dominio de identidad en otro
- Replicar asignaciones de roles predefinidos de un entorno a otro

Pasos para Windows

1. Cree `replicateusers.bat`, `replicateusers.ps1`, `replicatepredefinedroles.bat` y `replicatepredefinedroles.ps1` y guárdelos en un directorio local en el que tenga privilegios de escritura y ejecución.

2. Actualice los archivos por lotes con información de los entornos de origen y destino, y el servidor proxy de Internet, si es necesario.
3. Ejecute `replicateusers.bat`, que ejecuta `replicateusers.ps1`. Debe especificar la contraseña predeterminada que se asignará a los usuarios replicados como un parámetro de línea de comandos de la siguiente manera:
   `replicateusers.bat Pwd_for_users`

   Si la contraseña contiene caracteres especiales, asegúrese de usar el carácter de escape adecuado. Consulte Manejo de caracteres especiales.
4. Ejecute `replicatepredefinedroles.bat` para crear asignaciones de roles idénticas a las que existen en el entorno de origen.

Pasos para Linux/UNIX

1. Cree los scripts `replicateusers.sh` y `replicatepredefinedroles.sh` y guárdelos en un directorio local en el que tenga privilegios de escritura y ejecución.
2. Actualice `replicateusers.sh` y `replicatepredefinedroles.sh` con información de los entornos de origen y destino, y el servidor proxy de Internet, si es necesario.
3. Ejecute `replicateusers.sh`. Debe especificar la contraseña predeterminada que se asignará a los usuarios replicados como un parámetro de línea de comandos de la siguiente manera:
   `./replicateusers.sh Pwd_for_users`

   Si la contraseña contiene caracteres especiales, asegúrese de usar el carácter de escape adecuado. Consulte Manejo de caracteres especiales.
4. Ejecute el script `replicatepredefinedroles.sh` para crear asignaciones de roles idénticas a las que existen en el entorno de origen.

Replicación de los usuarios de un dominio de identidad en otro

Utilice los scripts de esta sección para clonar los usuarios de un dominio de identidad a otro dominio de identidad. El usuario que ejecute estos scripts debe tener roles de Administrador de dominio de identidad y de Administrador del servicio en los entornos de origen y de destino.

Windows

Cree `replicateusers.bat` y `replicateusers.ps1` copiando los scripts de esta sección.

1. Cree `replicateusers.ps1` copiando el script:

```
# 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 $usersreport
if ($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
rm deletefile*.log | Out-Null
echo "Replicate users script completed"
```

2. Cree `replicateusers.bat` copiando este script:

```
@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. Actualice `replicateusers.bat`. Consulte la siguiente tabla para obtener los valores que debe especificar.

Parámetro	Descripción
`epmusersource`	El nombre de usuario de un usuario con roles de Administrador de dominio de identidad y Administrador del servicio en el entorno de origen. **Ejemplos:** Windows: `set epmusersource="jDoe"` Linux/UNIX: `epmusersource="jDoe"`
`epmpwdsource`	Contraseña del usuario o la ruta absoluta del archivo de contraseñas cifradas. **Ejemplos:** Windows: `set epmpwdsource="Example"` Linux/UNIX: `epmpwdsource="Example"`
`epmurlsource`	URL del entorno del que se van a copiar los usuarios. **Ejemplos:** Windows: `set epmurlsource="https://example.oraclecloud.com"` Linux/UNIX: `epmurlsource="https://example.oraclecloud.com"`
`epmidentitydomainsource`	Nombre del dominio de identidad que usa el entorno de origen. **Ejemplos:** Windows: `set epmidentitydomainsource="example_source_dom"` Linux/UNIX: `epmidentitydomainsource="example_source_dom"`
`epmusertarget`	El nombre de un usuario con roles de Administrador de dominio de identidad y Administrador del servicio en el entorno de destino. **Ejemplos:** Windows: `set epmusertarget="John.Doe"` Linux/UNIX: `set epmusertarget="John.Doe"`
`epmpwdtarget`	Contraseña del usuario o la ruta absoluta del archivo de contraseñas cifradas. **Ejemplos:** Windows: `set epmpwdtarget="Example1"` Linux/UNIX: `epmpwdtarget="Example1"`
`epmurltarget`	URL del entorno en el que se van a crear los usuarios. **Ejemplos:** Windows: `set epmurltarget="https://example.oraclecloud.com"` Linux/UNIX: `epmurltarget="https://example.oraclecloud.com"`

Parámetro	Descripción
`epmidentitydomaintarget`	Nombre del dominio de identidad que usa el entorno de destino. **Ejemplos:** Windows: `set epmidentitydomaintarget="example_source_dom"` Linux/UNIX: `epmidentitydomaintarget="example_target_dom"`
`proxyserverusername`	Nombre de usuario utilizado para autentificar una sesión segura con el servidor proxy que controla el acceso a Internet. Borre todas las apariciones de esta propiedad si no se utiliza. **Ejemplos:** Windows: `set proxyserverusername="Example"` Linux/UNIX: `proxyserverusername="Example"`
`proxyserverpassword`	Contraseña para autentificar el usuario con el servidor proxy. Borre todas las apariciones de esta propiedad si no se utiliza. **Ejemplos:** Windows: `set proxyserverpassword="examplePwd"` Linux/UNIX: `proxyserverpassword="examplePwd"`
`proxyserverdomain`	Nombre del dominio definido para el servidor proxy. Borre todas las apariciones de esta propiedad si no se utiliza. **Ejemplos:** Windows: `set proxyserverdomain="exampleDom"` Linux/UNIX: `proxyserverdomain="exampleDom"`
`emailtoaddress`	Opcionalmente, dirección de correo electrónico a la que se va a enviar el informe de asignación de roles. El informe se envía por correo electrónico solo si se especifica este valor. **Ejemplo:** `emailtoaddress=john.doe@example.com`

Linux/UNIX

1. Cree `replicateusers.sh` copiando el siguiente script.

```
#!/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 [ "${userpassword}" == "" ]
then
    echo "USER_PASSWORD is missing. This is used to set the default
password for the replicated users."
    echo "${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. Actualice `replicateusers.sh`. Consulte la tabla anterior para obtener información sobre los valores que debe especificar. Además, debe especificar los valores de estas propiedades:
   - `javahome`: la ruta absoluta al directorio donde está instalado Java.
   - `epmautomatescript`: ubicación de `epmautomatescript.sh`, por ejemplo, `epmautomatescript="/home/user1/epmautomate/bin/epmautomate.sh"`

Replicar asignaciones de roles predefinidos de un entorno a otro

Utilice los scripts de esta sección para clonar asignaciones de roles predefinidos de un entorno a otro El usuario que ejecute estos scripts debe tener un rol de Administrador del servicio en ambos entornos.

> **Nota:**
>
> **Si utiliza la versión en PDF de este documento:** para evitar saltos de líneas e información de pie de página que inutilizarán estos scripts, cópielos de la Versión HTML de este tema.

Windows

1. Cree `replicatepredefineroles.ps1` copiando el siguiente script.

```
# 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

    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
        $txt = [io.file]::ReadAllText("$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. Cree `replicatepredefineroles.bat` copiando el siguiente script.

```
@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>"
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. Actualice `replicatepredefineroles.bat` según proceda. Consulte la siguiente tabla para obtener información sobre los valores que debe establecer para las propiedades de este archivo.

Actualización de `replicatepredefineroles.bat`

Parámetro	Descripción
`epmusersource`	El nombre de usuario de un usuario con roles de Administrador de dominio de identidad y Administrador del servicio en el entorno de origen. **Ejemplos:** Windows: `set epmusersource="jDoe"` Linux/UNIX: `epmusersource="jDoe"`
`epmpwdsource`	Contraseña del usuario o la ruta absoluta del archivo de contraseñas cifradas. **Ejemplos:** Windows: `set epmpwdsource="Example"` Linux/UNIX: `epmpwdsource="Example"`
`epmurlsource`	URL del entorno del que se van a copiar los usuarios. **Ejemplos:** Windows: `set epmurlsource="https://example.oraclecloud.com"` Linux/UNIX: `epmurlsource="https://example.oraclecloud.com"`

Parámetro	Descripción
`epmidentitydomainsource`	Nombre del dominio de identidad que usa el entorno de origen. **Ejemplos:** Windows: `set epmidentitydomainsource="example_source_dom"` Linux/UNIX: `epmidentitydomainsource="example_source_dom"`
`epmusertarget`	El nombre de un usuario con roles de Administrador de dominio de identidad y Administrador del servicio en el entorno de destino. **Ejemplos:** Windows: `set epmusertarget="John.Doe"` Linux/UNIX: `set epmusertarget="John.Doe"`
`epmpwdtarget`	Contraseña del usuario o la ruta absoluta del archivo de contraseñas cifradas. **Ejemplos:** Windows: `set epmpwdtarget="Example1"` Linux/UNIX: `epmpwdtarget="Example1"`
`epmurltarget`	URL del entorno en el que se van a crear los usuarios. **Ejemplos:** Windows: `set epmurltarget="https://example.oraclecloud.com"` Linux/UNIX: `epmurltarget="https://example.oraclecloud.com"`
`epmidentitydomaintarget`	Nombre del dominio de identidad que usa el entorno de destino. **Ejemplos:** Windows: `set epmidentitydomaintarget="example_target_dom"` Linux/UNIX: `epmidentitydomaintarget="example_target_dom"`
`proxyserverusername`	Nombre de usuario utilizado para autentificar una sesión segura con el servidor proxy que controla el acceso a Internet. Borre todas las apariciones de esta propiedad si no se utiliza. **Ejemplos:** Windows: `set proxyserverusername="Example"` Linux/UNIX: `proxyserverusername="Example"`
`proxyserverpassword`	Contraseña para autentificar el usuario con el servidor proxy. Borre todas las apariciones de esta propiedad si no se utiliza. **Ejemplos:** Windows: `set proxyserverpassword="examplePwd"` Linux/UNIX: `proxyserverpassword="examplePwd"`
`proxyserverdomain`	Nombre del dominio definido para el servidor proxy. Borre todas las apariciones de esta propiedad si no se utiliza. **Ejemplos:** Windows: `set proxyserverdomain="exampleDom"` Linux/UNIX: `proxyserverdomain="exampleDom"`

Parámetro	Descripción
`emailtoaddress`	Opcionalmente, dirección de correo electrónico a la que se va a enviar el informe de asignación de roles. El informe se envía por correo electrónico solo si se especifica este valor. **Ejemplo:** `emailtoaddress=john.doe@example.com`

Linux/UNIX

1. Cree `replicatepredefineroles.sh` copiando el siguiente script.

```
#!/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)

        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
        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. Actualice `replicatepredefineroles.sh`. Consulte la tabla anterior para obtener información sobre los valores que debe especificar. Además, debe especificar los valores de estas propiedades:
   - `javahome`: la ruta absoluta al directorio donde está instalado Java.
   - `epmautomatescript`: ubicación de `epmautomatescript.sh`, por ejemplo, `epmautomatescript="/home/user1/epmautomate/bin/epmautomate.sh"`

Creación de una cadencia de cambio de versión trimestral de Cloud EPM

Utilice estos scripts para crear una solución de autoservicio y omitir las actualizaciones para que los entornos de Cloud EPM se actualicen trimestralmente con un ciclo de prueba de dos semanas. En este caso, los entornos de producción se actualizan dos semanas más tarde que los entornos de prueba.

Este script también se puede usar para omitir cada dos actualizaciones mensuales si es necesario. De forma predeterminada, Cloud EPM aplica una actualización mensual a sus entornos. Utilice el comando skipUpdate para omitir la aplicación de las actualizaciones mensuales en un entorno o para ver las solicitudes de omisión de actualización actuales. Puede automatizar la ejecución manual de los comandos `skipUpdate` con los scripts incluidos en esta sección. Estos scripts automatizan el proceso de omisión de actualización para que las actualizaciones se apliquen de forma trimestral o cada dos meses.

> **Note:**
>
> 1. No puede omitir las actualizaciones durante más de dos meses consecutivos. Por ejemplo, el script devuelve un error si intenta que un entorno de Cloud EPM se actualice solo en febrero, junio y noviembre.
> 2. Todas las actualizaciones que se hayan producido durante el periodo intermedio se aplicarán al entorno durante la siguiente actualización. Por ejemplo, asuma que usa este script para programar actualizaciones trimestrales para que solo se produzcan en febrero, mayo, agosto y noviembre. En este caso, la actualización de mayo, por ejemplo, aplicará todas las actualizaciones y parches mensuales de Cloud EPM aplicables que se hayan publicado después de la actualización de febrero en su entorno. El proceso de mantenimiento puede tardar más tiempo de lo normal cuando se aplique la actualización.
> 3. Este script configura la cadencia de actualización solo para un trimestre. Ejecute este script mensualmente para garantizar que la cadencia de actualización esté configurada para todo el año.

- Instrucciones y script para Windows
- Instrucciones y script para UNIX/Linux
- Script de Groovy

Ejecución del script

1. Para ejecutar los scripts de Windows y Linux/UNIX:
   a. Cree el archivo `input.properties` y actualícelo con información de su entorno. Guarde el archivo en un directorio local. El contenido de este archivo difiere según su sistema operativo.
   Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar el script.
   b. Cree el script de bash `skip_update.ps1` (Windows PowerShell) o `skip_update.sh` (Linux/UNIX) y guárdelo en el directorio donde se encuentre `input.properties`.
   c. Inicie el script.
      - Linux/UNIX: ejecute `./skip_update.sh`.
      - Windows PowerShell: ejecute `skip_update.ps1`.
2. Para ejecutar el script de Groovy, use la pantalla correspondiente en un proceso de negocio de Cloud EPM o automatice la ejecución del script con runBusinessRule. Para obtener información sobre la ejecución del script de Groovy con EPM Automate, consulte Ejecución de comandos sin instalar EPM Automate.

Instrucciones y script para Windows

Cree `input.properties` y `skip_update.ps1` copiando los scripts de esta sección.

1. Cree `input.properties` copiando el siguiente script:

```
username=exampleAdmin
password=examplePassword.epw
```

```
url=exampleURL
updatemonths=02,05,08,11
```

2. Para actualizar `input.properties` especifique valores de parámetros.

Table 3-12 Parámetros de input.properties

Parámetro	Descripción
`username`	Nombre de usuario de un administrador del servicio.
`password`	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.
`url`	URL del entorno en el que desea establecer una cadencia de actualización que no sea mensual.
`updatemonths`	Lista de meses separados por comas cuando se deben aplicar actualizaciones de Oracle Fusion Cloud Enterprise Performance Management al entorno identificado con el parámetro `url`. Por ejemplo, `updatemonths=02,05,08,11`. Los meses se deben especificar con dos dígitos: 01 para enero hasta 12 para diciembre. Asegúrese de incluir un cero antes de los meses que van de enero a septiembre. El script intenta ejecutar el comando skipUpdate para los meses no incluidos en el valor del parámetro `updatemonths`. Por ejemplo, si especifica `updatemonths=02,05,08,11`, el script intenta establecer indicadores de actualización para enero, marzo, abrir, junio, julio, septiembre, octubre y diciembre de forma que las actualizaciones se realicen solo en febrero, mayo, agosto y noviembre.

3. Cree `skip_updates.ps1` copiando el siguiente script:

```
# 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
        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
    }
    }
```

```
    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 3) {
        echo "There are more than 2 months skipped from month ${thismonth}
to month ${nextmonthorig}. Please correct updatemonths in input.properties
so that there are not more than two 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
```

Instrucciones y script para UNIX/Linux

Cree `input.properties` y `skip_update.sh` copiando los scripts de esta sección.

1. Cree `input.properties` copiando el siguiente script:

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
username=exampleAdmin
```

```
password=examplePassword.epw
url=exampleURL
updatemonths=02,05,08,11
```

2. Para actualizar `input.properties` especifique valores de parámetros.

Table 3-13 Parámetros de input.properties

Parámetro	Descripción
`javahome`	Ubicación `JAVA_HOME`.
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`).
`username`	Nombre de usuario de un administrador del servicio.
`password`	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.
`url`	URL del entorno en el que desea establecer una cadencia de actualización que no sea mensual.
`updatemonths`	Lista de meses separados por comas cuando se deben aplicar actualizaciones de Oracle Fusion Cloud Enterprise Performance Management al entorno identificado con el parámetro `url`. Por ejemplo, `updatemonths=02,05,08,11`. Los meses se deben especificar con dos dígitos; incluya un cero antes de los meses que van de enero a septiembre. El script intenta ejecutar el comando skipUpdate para los meses no incluidos en el valor del parámetro `updatemonths`. Por ejemplo, si especifica `updatemonths=02,05,08,11`, el script intenta establecer indicadores de actualización para enero, marzo, abrir, junio, julio, septiembre, octubre y diciembre de forma que las actualizaciones se realicen solo en febrero, mayo, agosto y noviembre.

3. Cree `skip_updates.sh` copiando el siguiente script:

```
#!/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
        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[@]}
        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 3 ]]
    then
        echo "There are more than 2 months skipped from month ${thismonth}
to month ${nextmonthorig}. Please correct updatemonths in input.properties
so that there are not more than two 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 [[ ${nextupdatemonth} == "" ]]
        then
            nextupdatemonth="${updatemonthsarr[0]}"
```

```
                nextupdatemonthint=${nextupdatemonth#0}
            fi

            compareUpdateMonths ${thisupdatemonthint} ${nextupdatemonthint}
        done
    }

    validateUpdateMonths
    populateFromCurrentArrays
    processSkipUpdates
```

Script de Groovy

Si las contraseñas contienen caracteres especiales, consulte Manejo de caracteres especiales Asegúrese también de reemplazar estos valores de parámetros para que se ajusten a su entorno:

Table 3-14 Parámetros que cambiar

Parámetro	Descripción
user	Nombre de usuario de un administrador del servicio.
password	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.
url	URL del entorno en el que desea establecer una cadencia de actualización que no sea mensual.
updatemonths	Lista de meses separados por comas cuando se deben aplicar actualizaciones de Oracle Fusion Cloud Enterprise Performance Management al entorno identificado con el parámetro `url`. Por ejemplo, `updatemonths=02,05,08,11`. Los meses se deben especificar con dos dígitos: 01 para enero hasta 12 para diciembre. Asegúrese de incluir un cero antes de los meses que van de enero a septiembre. El script intenta ejecutar el comando skipUpdate para los meses no incluidos en el valor del parámetro `updatemonths`. Por ejemplo, si especifica `updatemonths=02,05,08,11`, el script intenta establecer indicadores de actualización para enero, marzo, abrir, junio, julio, septiembre, octubre y diciembre de forma que las actualizaciones se realicen solo en febrero, mayo, agosto y noviembre.

```
import java.text.SimpleDateFormat

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<>()
```

```
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 > 3) {
        LogMessage('There are more than 2 months skipped from month ' +
thismonth + ' to month ' + nextmonthorig + '. Please correct updatemonths so
that there are not more than two 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) {
```

```
            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) {
```

```
        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')
```

Creación de una cadencia de cambio de versión trimestral de Cloud EPM con ciclos de prueba de seis semanas

Utilice el script de esta sección para crear una solución de autoservicio y omitir las actualizaciones para que los entornos de Oracle Fusion Cloud Enterprise Performance Management se actualicen trimestralmente con un ciclo de prueba de seis semanas. En este caso, los entornos de producción se actualizan seis semanas más tarde que los entornos de prueba.

De forma predeterminada, Cloud EPM aplica una actualización mensual a sus entornos. Utilice el comando skipUpdate para omitir la aplicación de las actualizaciones mensuales en un entorno o para ver las solicitudes de omisión de actualización actuales. Puede automatizar la ejecución manual de los comandos `skipUpdate` con los scripts incluidos en esta sección. Estos scripts automatizan el proceso de omisión de actualización para que las actualizaciones se apliquen de forma trimestral con un ciclo de prueba de seis semanas.

Note:

1. No puede omitir las actualizaciones durante más de tres meses consecutivos. Este script devuelve un error si intenta que un entorno de Cloud EPM se actualice solo en febrero y octubre.
2. Todas las actualizaciones que se hayan producido durante el periodo intermedio se aplicarán al entorno durante la siguiente actualización. Por ejemplo, suponga que utiliza este script para programar las actualizaciones trimestrales de modo que se realicen solo para las actualizaciones de febrero, mayo, agosto y noviembre. En este caso, la actualización de mayo, por ejemplo, aplicará todas las actualizaciones y parches mensuales de Cloud EPM aplicables que se hayan publicado después de la actualización de febrero en su entorno. El proceso de mantenimiento puede tardar más tiempo de lo normal cuando se aplique la actualización.
3. Este script configura la cadencia de actualización solo para un trimestre. Escenario de ejemplo: el ciclo de actualización del entorno de prueba se establece en el primer viernes de febrero (actualización 24.02), mayo (actualización 24.05), agosto (actualización 24.08) y noviembre (actualización 24.11). El entorno de producción se actualizará el tercer viernes de marzo (actualización 24.02) con la versión que se ha utilizado para actualizar el entorno de prueba el primer viernes de febrero (actualización 24.02). Una actualización similar en el entorno de producción tendrá lugar en la tercera semana de junio (actualización 24.05), septiembre (actualización 24.08) y diciembre (actualización 24.11). En este escenario, el entorno de producción no se actualiza a la fecha actual, sino con la fecha actual del entorno de prueba.

Script de Windows de ejemplo

Cree `skip_update.ps1` copiando el siguiente script. Almacénelo en un directorio local. Consulte Ejecución del script para obtener información sobre la ejecución de este script:

```
# Skip Update PowerShell script

$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]
            $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
        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
```

Script de Linux/UNIX de ejemplo

Cree `skip_update.sh` copiando el siguiente script. Almacénelo en un directorio local. Consulte Ejecución del script para obtener información sobre la ejecución de este script:

```
#!/bin/sh

. ./input.properties

echo "Starting skip_update.sh script."

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
updateversionsarr=( $(echo "${updateversions}" | sed 's/,/ /g') )
currentyear=$(date +%y)
nextyear=$((currentyear+1))
currentmonth=$(date +%m)

populateFromCurrentArrays() {
    local startposition=0

    for i in ${!monthsarr[@]}
    do
        if [[ "${currentmonth}" == "${monthsarr[$i]}" ]]
        then
            if [[ "${podtype}" == "prod" ]]
            then
                if [[ ${updateversionsarr[@]} =~ ${currentmonth} ]]
                then
                    startposition=$((i-2))
                else
                    startposition=$((i-1))
                fi
                break
            else
                if [[ ${updateversionsarr[@]} =~ ${currentmonth} ]]
                then
                    startposition=$i
                else
                    startposition=$((i-1))
                fi
                break
            fi
        fi
    done

    if [[ ${startposition} -lt 0 ]]
    then
        startposition=$((startposition+12))
    fi

    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
    local countlimit=0

    if [[ "${podtype}" == "prod" ]]
    then
        countlimit=3
    else
        countlimit=2
    fi

    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: ${epmautomatescript} skipUpdate remove"
```

```
    ${epmautomatescript} skipUpdate remove

    for i in ${!monthsarrfromcurrent[@]}
    do
        local match=1

        if [[ ${addcount} -eq ${countlimit} ]]
        then
            echo "Update add calls are completed. No more will be attempted."
            break
        fi

        for j in ${!updateversionsarr[@]}
        do
            if [[ "${currentmonth}" == "${updateversionsarr[$j]}" ]] && [[ $
{addcount} -gt 0 ]]
            then
                match=1
                break
            fi

            if [[ "${monthsarrfromcurrent[$i]}" == "$
{updateversionsarr[$j]}" ]] && [[ ${addcount} -eq 0 ]]
            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 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
}

validateUpdateVersions
populateFromCurrentArrays
processSkipUpdates
```

Script de Groovy

Cree el script de Groovy `skip_update.groovy` copiando el siguiente script y actualizándolo a continuación. Consulte Ejecución del script para obtener información sobre la ejecución de este script:
Actualice las siguientes variables en este script de Groovy:

- `username` Nombre de usuario de un Administrador del servicio en el entorno en el que desea definir la cadencia de actualización no mensual.
- `password` Contraseña del administrador de servicio o nombre y ubicación del archivo de contraseña cifrado.
- `url` URL del entorno en el que desea definir una cadencia de actualización que no sea mensual.
- `updateversions` Lista separada por comas de actualizaciones de Cloud EPM que se deben aplicar al entorno identificado por el parámetro `url`. Por ejemplo, `updateversions=02,05,08,11`.
  Las versiones se deben especificar con dos dígitos; incluya un cero delante para las actualizaciones 01 a 09. El script intenta ejecutar el comando skipUpdate para las actualizaciones no incluidas en el valor de parámetro `updateversions`. Por ejemplo, si especifica `updateversions=02,05,08,11`, el script intenta definir indicadores de omisión de actualización para las actualizaciones 01 (enero), 03 (marzo), 04 (april), 06 (junio), 07

(jullio), 09 (septiembre), 10 (octubre) y 12 (diciembre). En este caso, se aplican al entorno las actualizaciones de Cloud EPM 02 (febrero), 05 (mayo), 08 (agosto) y 11 (noviembre).

- `podtype` Tipo de entorno de Cloud EPM. Los valores válidos son `test` y `prod`.
- `proxyserverusername` Nombre de usuario para autentificar una sesión segura con el servidor proxy que controla el acceso a Internet.
- `proxyserverpassword` Contraseña para autenticar el usuario con el servidor proxy.
- `proxyserverdomain` Nombre del dominio definido para el servidor proxy.

> **Note:**
>
> Si no utiliza un servidor proxy, no especifique ningún valor para los parámetros `proxyserverusername`, `proxyserverpassword` y `proxyserverdomain`.

```
import java.text.SimpleDateFormat

String username = 'service_administrator'
String password = 'examplePWD'
String url = 'example_EPM_URL'
String updateversions = '01,04,07,10'
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) {
    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 {
            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)

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')
```

Creación del archivo input.properties para ejecutar scripts `skip_update` en Windows y Linux/UNIX

Para ejecutar `skip_update.ps1` o `skip_update.sh`, cree el archivo `input.properties` y actualícelo con información para el entorno. Guarde el archivo en un directorio local. El contenido de este archivo difiere en función del sistema operativo.

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 Parámetros de input.properties

Parámetro	Descripción
`javahome`	Ubicación `JAVA_HOME`. Solo para Linux/UNIX.
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`). Solo para Linux/UNIX.
`username`	Nombre de usuario de un administrador del servicio.

Table 3-15 (Cont.) Parámetros de input.properties

Parámetro	Descripción
`password`	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.
`url`	URL del entorno en el que desea establecer una cadencia de actualización que no sea mensual.
`updateversions`	`updateversions` Lista separada por comas de actualizaciones de Cloud EPM que se deben aplicar al entorno identificado por el parámetro `url`. Por ejemplo, `updateversions=02,05,08,11`. Las versiones se deben especificar con dos dígitos; incluya un cero delante para las actualizaciones 01 a 09. El script intenta ejecutar el comando skipUpdate para las actualizaciones no incluidas en el valor de parámetro `updateversions`. Por ejemplo, si especifica `updateversions=02,05,08,11`, el script intenta definir indicadores de omisión de actualización para las actualizaciones 01 (enero), 03 (marzo), 04 (april), 06 (junio), 07 (jullio), 09 (septiembre), 10 (octubre) y 12 (diciembre). En este caso, se aplican al entorno las actualizaciones de Cloud EPM 02 (febrero), 05 (mayo), 08 (agosto) y 11 (noviembre).
`podtype`	Tipo de entorno de Cloud EPM. Los valores válidos son `test` y `prod`.

Ejecución del script

1. **Solo para Windows y Linux/UNIX:**
   - Cree `skip_update.ps1` o `skip_update.sh` copiando el script de una sección anterior.
   - Cree el archivo `input.properties` y guárdelo en el directorio en el que está ubicado el script `skip_update`. El contenido de este archivo difiere según su sistema operativo. Consulte Creación del archivo input.properties para ejecutar scripts skip_update en Windows y Linux/UNIX.
     Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar el script.
   - Inicie el script.
     - **Windows PowerShell:** ejecute `skip_update.ps1`.
     - **Linux/UNIX:** ejecute `./skip_update.sh`.
2. **Groovy:**
   - Cree el script de Groovy `skip_update.groovy` y actualícelo según sea necesario.
   - Utilice la pantalla de Groovy en un proceso de negocio de Cloud EPM o automatice la ejecución del script mediante runBusinessRule. Para obtener información sobre la ejecución del script de Groovy con EPM Automate, consulte Ejecución de comandos sin instalar EPM Automate.

Escenarios de ejemplo para Planning, Consolidation, Tax Reporting y Enterprise Profitability and Cost Management

Los scripts disponibles en esta sección le ayudan a automatizar las tareas en los entornos de Planning (incluido Planning Modules), Financial Consolidation and Close, Tax Reporting y Enterprise Profitability and Cost Management.

Consulte también:

- Automatización de la exportación de un gran número de celdas desde un cubo de almacenamiento agregado
  Utilice el script de PowerShell o de Bash de esta sección para exportar un gran número de celdas desde un cubo de almacenamiento agregado (ASO).
- Importación de metadatos a una aplicación
  Utilice estos scripts para importar metadatos de aplicación desde un archivo de forma manual.
- Importación de datos, ejecución de un script de cálculo y copia de datos desde una base de datos de almacenamiento de bloques a una base de datos de almacenamiento agregado
  Utilice estos scripts para importar datos de un archivo, refrescar el cubo, ejecutar una regla de negocio para calcular el cubo y, a continuación, enviar los datos a un cubo de ASO.
- Exportación y descarga de metadatos y datos
  Utilice estos scripts para exportar metadatos y datos de aplicación y, a continuación, descargar los archivos de exportación en un directorio local.
- Exportación y descarga de datos de aplicación
  Utilice estos scripts para exportar y datos de aplicación y, a continuación, descargarlos en un directorio local.
- Automatización del archivado de los registros de auditoría de aplicación
  Utilice los scripts de Windows y Linux de esta sección para automatizar el proceso de exportación y archivado de datos de auditoría de aplicación en un equipo local.
- Carga de un archivo de datos en un entorno y ejecución de una regla de carga de datos
  Utilice estos scripts para cargar un archivo en un entorno y, a continuación, ejecutar una regla de datos para importar datos del archivo en una aplicación.
- Automatización de la integración de datos diaria
  En este escenario se explora el uso de un script de ejemplo para automatizar la integración de datos periódica.

Automatización de la exportación de un gran número de celdas desde un cubo de almacenamiento agregado

Utilice el script de PowerShell o de Bash de esta sección para exportar un gran número de celdas desde un cubo de almacenamiento agregado (ASO).

Debido a los límites impuestos por `QUERYRESULTLIMIT` de Oracle Essbase, es imposible exportar una gran cantidad de datos desde la interfaz de usuario. Con el script de PowerShell disponible en esta sección se divide la operación de exportación en un número especificado de trabajos, se ejecuta cada trabajo, se descargan los datos exportados y se concatenan los archivos de exportación en un archivo de exportación, lo que garantiza que solo haya una cabecera.

> **Nota:**
>
> Con estos scripts se ejecuta un trabajo existente del tipo Exportar datos. Para obtener instrucciones detalladas sobre la creación de trabajos, consulte "Gestión de trabajos" en *Administración de Planning*.

Script de 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 += $_.name
        }
        getLevel0($_)
    })
}

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 ..."
    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
    }

    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 de Bash

```
#!/bin/bash

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>

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
    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 ]]
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}")
        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"
```

Para exportar un gran número de celdas desde un cubo de almacenamiento agregado (ASO):

1. Copie el script de PowerShell o Bash y guárdelo en el sistema de archivos, por ejemplo, como `ASOCellExport.ps1` o `ASOCellExport.sh`.
2. Modifique el archivo de script y establezca los valores de parámetro. Consulte la tabla a continuación para ver información detallada.

Tabla 3-16 Valores de variables que se deben incluir en los scripts de PowerShell y Bash

Variable	Descripción
`user`	Dominio y nombre de usuario del Administrador del servicio en formato *`DOMAIN.USER`*. **Ejemplos:** **Windows:** `$user = 'exampleDomain.jDoe'` **Linux/UNIX:** `user = 'exampleDomain.jDoe'`
`pass`	Contraseña del administrador de servicio o ubicación del archivo de contraseña cifrado. Consulte el comando encrypt para obtener información sobre la creación de un archivo de contraseñas cifradas. **Ejemplos:** **Windows:** `$pass = 'Example'` **Linux/UNIX:** `pass = 'Example'`
`serverURL`	URL del entorno de Oracle Fusion Cloud Enterprise Performance Management. **Ejemplos:** **Windows:** `$serverURL = 'https://example .oraclecloud.com'` **Linux/UNIX:** `serverURL = 'https://example .oraclecloud.com'`
`applicationName`	Nombre de una aplicación de Planning, Financial Consolidation and Close, Tax Reporting o Enterprise Profitability and Cost Management. **Ejemplos:** **Windows:** `$applicationName = 'Vision'` **Linux/UNIX:** `applicationName = 'Vision'`
`cubeName`	Nombre de un cubo de la aplicación. **Ejemplos:** **Windows:** `$cubeName = 'VisASO'` **Linux/UNIX:** `cubeName = 'VisASO'`
`splitDimension`	Nombre de una dimensión cuyos miembros se usan para dividir la exportación en grupos. **Ejemplos:** **Windows:** `$splitDimension = 'Account'` **Linux/UNIX:** `splitDimension = 'Account'`

Tabla 3-16 (Continuación) Valores de variables que se deben incluir en los scripts de PowerShell y Bash

Variable	Descripción
`topLevelMemberForExport`	Nombre de un miembro de la subjerarquía de dimensiones en la que se crea una lista de miembros de nivel 0. **Ejemplos:** **Windows:** `$topLevelMemberForExport = 'Total Cash Flow'` **Linux/UNIX:** `topLevelMemberForExport = 'Total Cash Flow'`
`exportJobName`	Nombre de un trabajo existente del tipo Exportar datos. La configuración especifica en este trabajo se sobrescribirá con los parámetros que establezca en el script. **Ejemplos:** **Windows:** `$exportJobName = 'ASO Cell Export'` **Linux/UNIX:** `exportJobName = 'ASO Cell Export'`
`exportFilePrefix`	Prefijo de nombre de archivo para identificar de forma exclusiva los archivos generados por el trabajo de exportación. **Ejemplos:** **Windows:** `$exportFilePrefix = 'cashflow'` **Linux/UNIX:** `exportFilePrefix = 'cashflow'`
`columnMembers`	Columnas de miembros que incluir en la exportación. **Ejemplos:** **Windows:** `$columnMembers = 'Period'` **Linux/UNIX:** `columnMembers = 'Period'`

`povMembers`	Punto de vista que incluir en la exportación. Los miembros de PDV deben incluir el resto de dimensiones y pueden incluir funciones como se muestra a continuación:
	`ILvl0Descendants(YearTotal),`
	`ILvl0Descendants(Year),`
	`ILvl0Descendants(Scenario),`
	`ILvl0Descendants(Version),`
	`ILvl0Descendants(P_TP), ILvl0Descendants(AltYear)`
	Ejemplos:
	Windows: `$povMembers = 'YTD'`
	Linux/UNIX: `povMembers = 'YTD'`

`numberOfExportFiles`	Número de trabajos que ejecutar para esta operación de exportación. Si la exportación sigue fallando debido a limitaciones de la consulta, aumente este número. **Ejemplos:** **Windows:** `$numberOfExportFiles = 3` **Linux/UNIX:** `numberOfExportFiles = 3`

3. Con el programador de Windows o un trabajo cron, programe el script para que se ejecute a una hora adecuada. Consulte Automatización de la ejecución de scripts para obtener pasos detallados.

Importación de metadatos a una aplicación

Utilice estos scripts para importar metadatos de aplicación desde un archivo de forma manual.

Estos scripts realizan las siguientes actividades:

- Iniciar sesión en un entorno.
- Cargar un archivo de metadatos.
- Importar metadatos del archivo cargado en una aplicación mediante un trabajo.
- Refrescar el cubo.
- Cerrar sesión.

Script de Windows de ejemplo

Cree `importMetadata.ps1` copiando el siguiente script. Almacénelo en un directorio 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}
epmautomate refreshcube
epmautomate logout
```

Script de Linux/UNIX de ejemplo

Cree `importMetadata.sh` copiando el siguiente script. Almacénelo en un directorio 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
```

Creación del archivo input.properties

Cree el archivo `input.properties` copiando uno de los siguientes scripts y actualizándolo con información para su entorno. Guarde el archivo en el directorio en el que está almacenado `importMetadata.ps1` o `importMetadata.sh`.

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
```

Tabla 3-17 Parámetros de input.properties

Parámetro	Descripción
`javahome`	Ubicación `JAVA_HOME`. Solo para Linux/UNIX.
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`). Solo para Linux/UNIX.
`username`	Nombre de usuario de un administrador del servicio que también tiene asignado el rol de administrador del dominio de identidad.
`password`	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.
`serviceURL`	URL del entorno a partir del cual desea generar la instantánea.
`File1`	Nombre del archivo ZIP que contiene los metadatos que se van a importar.
`JobName`	Trabajo que se utilizará para importar los metadatos.

Ejecución de los scripts

1. Cree `importMetadata.ps1` o `importMetadata.sh` copiando el script de una sección anterior.
2. Cree el archivo `input.properties` y guárdelo en el directorio en el que está ubicado el script `importMetadata`. El contenido de este archivo difiere según su sistema operativo. Consulte Creación del archivo input.properties.
   Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar el script.
3. Inicie el script.
   - **Windows PowerShell:** ejecute `importMetadata.ps1`.
   - **Linux/UNIX:** ejecute `./importMetadata.sh`.

Importación de datos, ejecución de un script de cálculo y copia de datos desde una base de datos de almacenamiento de bloques a una base de datos de almacenamiento agregado

Utilice estos scripts para importar datos de un archivo, refrescar el cubo, ejecutar una regla de negocio para calcular el cubo y, a continuación, enviar los datos a un cubo de ASO.

Estos scripts realizan las siguientes acciones:

- Iniciar sesión en un entorno.
- Cargar un archivo `data.csv`.
- Importar datos de `data.csv` en la aplicación mediante el trabajo `loadingq1data`.
- Refrescar el cubo.
- Ejecutar reglas de negocio para transformar datos.
- Enviar datos a una base de datos de almacenamiento agregado mediante un trabajo.
- Cerrar sesión.

Script de Windows de ejemplo

Cree `importDataPlus.ps1` copiando este script. Guárdelo en un directorio local.

```
$inputproperties = ConvertFrom-StringData(Get-Content ./input.properties -raw)
$username="$($inputproperties.username)"
$passwordfile="$($inputproperties.passwordfile)"
$serviceURL="$($inputproperties.serviceURL)"
$importDataJobName="$($inputproperties.importDataJobName)"
$businessRuleName="$($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
```

Script de Linux/UNIX de ejemplo

Cree `importDataPlus.ps1` copiando este script. Guárdelo en un directorio 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
```

Creación del archivo 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
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
```

Tabla 3-18 Parámetros de input.properties

Parámetro	Descripción
`javahome`	Ubicación `JAVA_HOME`. Solo para Linux/UNIX.
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`). Solo para Linux/UNIX.
`username`	Nombre de usuario de un administrador del servicio que también tiene asignado el rol de administrador del dominio de identidad.

Tabla 3-18 (Continuación) Parámetros de input.properties

Parámetro	Descripción
`password`	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.
`serviceURL`	URL del entorno a partir del cual desea generar la instantánea.
`File1`	Archivo de importación desde el que se cargarán los datos en la aplicación.
`iimportDataJobName`	Nombre del trabajo que se utilizará para importar datos.
`businessRuleName`	Regla de negocio que se ejecutará en los datos importados
`planTypeMapName`	Asignación de tipo de plan que se utilizará para copiar datos de una base de datos de BSO en una base de datos de ASO o de una base de datos de BSO a otra base de datos de BSO.
`param1Key`	Petición de datos en tiempo de ejecución 1 para ejecutar la regla de negocio.
`param1Value`	Valor de la petición de datos en tiempo de ejecución 1.
`param2Key`	Petición de datos en tiempo de ejecución 2 para ejecutar la regla de negocio.
`param2Value`	Valor de la petición de datos en tiempo de ejecución 2.
`clearData`	Indica si se van a eliminar los datos en la base de datos de recepción. Especifique `false` para conservar los datos.

Ejecución de los scripts

1. Cree `importDataPlus.ps1` o `importDataPlus.sh` copiando el script de una sección anterior.
2. Cree el archivo `input.properties` y guárdelo en el directorio en el que está ubicado el script `importDataPlus`. El contenido de este archivo difiere según su sistema operativo. Consulte Creación del archivo input.properties.
   Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar el script.
3. Inicie el script.
   - **Windows PowerShell:** ejecute `importDataPlus.ps1`.
   - **Linux/UNIX:** ejecute `./importDataPlus.sh`.

Exportación y descarga de metadatos y datos

Utilice estos scripts para exportar metadatos y datos de aplicación y, a continuación, descargar los archivos de exportación en un directorio local.

Estos scripts completan las siguientes actividades:

- Iniciar sesión en un entorno.
- Exporta los metadatos en un archivo .zip utilizando un trabajo especificado.
- Exporta los datos de aplicación en un archivo .zip utilizando un trabajo especificado.
- Muestra el contenido de la bandeja de entrada o del buzón de salida.

- Descarga los archivos de datos exportados al equipo local.
- Cierre la sesión.

Script de Windows de ejemplo

Cree `exportDownloadMetadataAndData.ps1` copiando el siguiente script. Almacénelo en un directorio 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
```

Script de Linux/UNIX de ejemplo

Cree `exportDownloadMetadataAndData.sh` copiando el siguiente script. Almacénelo en un directorio 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}"
${epmautomatescript} downloadfile "${exportFile2}"
${epmautomatescript} logout
```

Creación del archivo de propiedades

Cree el archivo `input.properties` copiando uno de los siguientes scripts y actualizándolo con información para su entorno. Guarde el archivo en el directorio en el que está almacenado `exportDownloadMetadataAndData.ps1` o `exportDownloadMetadataAndData.sh`.

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
```

Tabla 3-19 Parámetros de input.properties

Parámetro	Descripción
`javahome`	Ubicación `JAVA_HOME`. Solo para Linux/UNIX.
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`). Solo para Linux/UNIX.
`username`	Nombre de usuario de un administrador del servicio que también tiene asignado el rol de administrador del dominio de identidad.
`password`	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.
`serviceURL`	URL del entorno a partir del cual desea generar la instantánea.
`exportFile1`	Nombre del archivo al que se exportarán los metadatos.
`exportFile2`	Nombre del archivo al que se exportarán los datos.
`exportDataJobName1`	Trabajo que se utilizará para exportar los metadatos.
`exportDataJobName2`	Trabajo que se utilizará para exportar los datos.

Ejecución de los scripts

1. Cree `exportDownloadMetadataAndData.ps1` o `exportDownloadMetadataAndData.sh` copiando el script de una sección anterior.
2. Cree el archivo `input.properties` y guárdelo en el directorio en el que está ubicado el script `exportDownloadMetadataAndData`. El contenido de este archivo difiere según su sistema operativo. Consulte Creación del archivo de propiedades.
   Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar el script.
3. Inicie el script.
   - **Windows PowerShell:** ejecute `exportDownloadMetadataAndData.ps1`.
   - **Linux/UNIX:** ejecute `./exportDownloadMetadataAndData.sh`.

Exportación y descarga de datos de aplicación

Utilice estos scripts para exportar y datos de aplicación y, a continuación, descargarlos en un directorio local.

Estos scripts realizan las siguientes operaciones:

- Iniciar sesión en un entorno.
- Realizar una copia de seguridad de los datos utilizando los trabajos que especifique.
- Descargar los archivos de datos exportados.
- Cerrar sesión.

Script de Windows de ejemplo

Cree `exportDownloadData.ps1` copiando este script. Guárdelo en un directorio 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
```

Script de Linux/UNIX de ejemplo

Cree `exportDownloadData.sh` copiando este script. Guárdelo en un directorio local.

```
#!/bin/bash
. ./input.properties
export JAVA_HOME=${javahome}
${epmautomatescript} login "${username}" "${passwordfile}" "${serviceURL}"
${epmautomatescript} exportdata "${exportDataJobName1}" "${exportFile1}"
${epmautomatescript} exportdata "${exportDataJobName2}" "${exportFile2}"
${epmautomatescript} listfiles
${epmautomatescript} downloadfile "${exportFile1}"
${epmautomatescript} downloadfile "${exportFile2}"
${epmautomatescript} logout
```

Creación del archivo input.properties

Cree el archivo `input.properties` copiando uno de los siguientes scripts y actualizándolo con información para su entorno. Guarde el archivo en el directorio en el que está almacenado `exportDownloadData.ps1` o `exportDownloadData.sh`.

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
```

Tabla 3-20 Parámetros de input.properties

Parámetro	Descripción
`javahome`	Ubicación `JAVA_HOME`. Solo para Linux/UNIX.
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`). Solo para Linux/UNIX.
`username`	Nombre de usuario de un administrador del servicio que también tiene asignado el rol de administrador del dominio de identidad.
`password`	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.
`serviceURL`	URL del entorno a partir del cual desea generar la instantánea.
`exportFile1` y `exportFile2`	Nombre del archivo al que se exportarán los datos.
`exportDataJobName1` y `exportDataJobName2`	Trabajo que se utilizará para exportar los datos.

Ejecución de los scripts

1. Cree `exportDownloadData.ps1` o `exportDownloadData.sh` copiando el script de una sección anterior.
2. Cree el archivo `input.properties` y guárdelo en el directorio en el que está ubicado el script `exportDownloadData`. El contenido de este archivo difiere según su sistema operativo. Consulte Tabla 1.
   Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar el script.
3. Inicie el script.

- **Windows PowerShell:** ejecute `exportDownloadData.ps1`.
- **Linux/UNIX:** ejecute `./exportDownloadData.sh`.

Automatización del archivado de los registros de auditoría de aplicación

Utilice los scripts de Windows y Linux de esta sección para automatizar el proceso de exportación y archivado de datos de auditoría de aplicación en un equipo local.

Los datos de auditoría de aplicaciones se conservan solo durante 365 días. Personalice estos scripts y ejecútelos una vez cada 180 días, o según lo estipulen sus normativas de retención de datos, para evitar la pérdida de datos históricos de auditoría con más de 365 días de antigüedad.

> **Note:**
>
> Estos scripts se han diseñado para archivar datos de forma local. Puede modificarlos para guardar los archivos de datos de auditoría exportados en la red o en la nube (por ejemplo, Oracle Object Storage).

Table 3-21 Parámetros y sus valores

Parámetro	Valor
`url`	La URL del entorno. **Ejemplos:** • **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`	El nombre de usuario de un Administrador del servicio para iniciar sesión en el entorno y descargar datos de auditoría. **Ejemplos:** • **Windows:** `set user=ExampleAdmin` • **Linux:** `user=ExampleAdmin`
`password`	Contraseña del Administrador del servicio (no recomendado) o el nombre y la ubicación del archivo de contraseña cifrado. Consulte el comando encrypt para obtener información sobre la creación de archivos de contraseña cifrada. **Ejemplos:** • **Windows:** `set password="C:\mySecuredir\password.epw"` • **Linux:** `password="/home/user1/mySecuredir/password.epw"`
`AuditFileName`	Nombre del archivo de datos de auditoría. Para que este archivo sea único, el script agrega el registro de hora de la exportación de datos de auditoría al nombre de este archivo. **Ejemplo:** • **Windows:** `set AuditFileName=AuditData` • **Linux:** `AuditFileName=AuditData`

Table 3-21 (Cont.) Parámetros y sus valores

Parámetro	Valor
`NumberOfBackups`	Número de archivos de copia de seguridad que se van a retener en el almacenamiento. El número predeterminado son 10, a partir de esa cantidad la copia de seguridad más antigua se sustituye según sea necesario. **Ejemplo:** • **Windows:** `set NumberOfBackups=20` • **Linux:** `NumberOfBackups=20`
Solo para el script de Linux	
`epmautomatescript`	La ubicación donde se instala EPM Automate. **Ejemplo:** `/home/user1/epmautomate/bin/epmautomate.sh`
`javahome`	Ubicación de *JAVA_HOME*. **Ejemplo:** `/home/user1/jdk1.8.0_191/`

Script de Windows

Cree un archivo por lotes (por ejemplo, `AuditExport.bat`) que contenga un script similar al siguiente para automatizar la exportación y descarga de datos de auditoría en un equipo 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
       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 de Linux

Cree un script de shell (por ejemplo, `AuditExports.sh`) que contenga un script similar al siguiente para automatizar la exportación y descarga de datos de auditoría en un equipo 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/jdk1.8.0_191/

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`

    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 --
```

Carga de un archivo de datos en un entorno y ejecución de una regla de carga de datos

Utilice estos scripts para cargar un archivo en un entorno y, a continuación, ejecutar una regla de datos para importar datos del archivo en una aplicación.

Requisitos previos

- Las siguientes definiciones en Data Management:
  - Definición de regla de carga de datos denominada `VisionActual`. Se supone que la regla de datos no especifica una ruta de archivo para el archivo de entrada.
  - Definiciones de periodos del `Mar-15` al `Jun-15`
- Archivo de datos con el formato correcto (`GLActual.dat`) que contenga datos.

Para importar datos y ejecutar una regla de carga de datos, ejecute comandos para realizar estos pasos:

- Inicie sesión en el entorno.
- Cargue un archivo `GLActual.dat` que contenga datos para los periodos del `Mar-15` al `Jun-15` en la carpeta `inbox/Vision` de Data Management.
- Importe los datos de `GLActual.dat` a Data Management utilizando la regla de carga de datos `VisionActual`, el periodo inicial `Mar-15`, el periodo final `Jun-15` y el modo de importación `REPLACE`.
- Exporte los datos con la opción `STORE_DATA` para fusionar los datos de la tabla temporal de Data Management con los datos de la aplicación existentes.
- Cierre la sesión.

Script de Windows de ejemplo

Cree `runDataLoadRule.ps1` copiando el siguiente script. Almacénelo en un directorio 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
```

Script de Linux/UNIX de ejemplo

Cree `runDataLoadRule.sh` copiando el siguiente script. Almacénelo en un directorio 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
```

Creación del archivo input.properties

Cree el archivo `input.properties` copiando uno de los siguientes scripts y actualizándolo con información para su entorno. Guarde el archivo en el directorio en el que está almacenado `runDataLoadRule.ps1` o `runDataLoadRule.sh`.

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
```

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
```

Tabla 3-22 Parámetros de input.properties

Parámetro	Descripción
`javahome`	Ubicación `JAVA_HOME`. Solo para Linux/UNIX.
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`). Solo para Linux/UNIX.
`username`	Nombre de usuario de un administrador del servicio que también tiene asignado el rol de administrador del dominio de identidad.
`password`	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.
`serviceURL`	URL del entorno a partir del cual desea generar la instantánea.
`dataFile`	Archivo que contiene los datos que se importarán utilizando la regla de datos.
`dataFileUploadLocation`	Ubicación en la que se cargarán los archivos de datos.
`dataRuleName`	Nombre de una regla de carga de datos definida en Data Integration.
`startPeriod`	Primer período para el que se van a cargar los datos. Este nombre de período debe estar definido en la asignación de períodos de Data Integration.
`endPeriod`	Para una carga de datos de varios períodos, el último período para el que cargarán los datos. Para una carga de un único período, introduzca el mismo período como período inicial. Este nombre de período debe estar definido en la asignación de períodos de Data Integration.
`importMode`	Modo para importar datos en Data Integration. Utilice `APPEND`, `REPLACE` o `RECALCULATE`. Utilice `NONE` para omitir la importación de datos en las tablas temporales.
`exportMode`	Modo para exportar datos a la aplicación. Utilice Data Integration. Utilice `STORE_DATA`, `ADD_DATA`, `SUBTRACT_DATA` o `REPLACE_DATA`. Utilice `NONE` para omitir la exportación de datos de Data Integration a la aplicación. **Nota:** Financial Consolidation and Close solo soporta los modos `MERGE` y `NONE`.

Ejecución del script

1. Cree `runDataLoadRule.ps1` o `runDataLoadRule.sh` copiando el script de una sección anterior.
2. Cree el archivo `input.properties` y guárdelo en el directorio en el que está ubicado el script `runDataLoadRule`. El contenido de este archivo difiere según su sistema operativo. Consulte Creación del archivo input.properties.
   Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar el script.
3. Inicie el script.

- **Windows PowerShell:** ejecute `runDataLoadRule.ps1`.
- **Linux/UNIX:** ejecute `./runDataLoadRule.sh`.

Automatización de la integración de datos diaria

En este escenario se explora el uso de un script de ejemplo para automatizar la integración de datos periódica.

Cree un archivo de lote (`.bat`) o shell (`.sh`) que contenga un script similar al siguiente para automatizar las actividades relacionadas con la integración de datos. El siguiente script de ejemplo para Windows automatiza la integración diaria de datos de las aplicaciones completando estas actividades:

- Inicie sesión en un entorno.
- Suprima `DailyPlanData` si está presente.
- Cargue `DailyPlanData` en el servicio.
- Ejecute la regla de negocio `Clear Plan Targets` en el tipo de plan `Plan1`.
- Importe los datos con el nombre del trabajo `LoadDailyPlan`.
- Ejecute la regla de negocio `Balance Sheet - Plan`.
- Ejecute la regla de negocio `Allocate Plan Targets`.
- Suprima `DailyTarget.zip` si está presente.
- Exporte datos a `DailyTarget.zip` con el nombre de trabajo `ExportDailyTarget`.
- Descargue `DailyTarget.zip` en el servidor y agregue la marca de tiempo.
- Cierre la sesión en el entorno.

> **Nota:**
>
> Si vuelve a planificar este script para su uso, asegúrese de modificar los valores de los parámetros `SET url` y `SET user`. Además, puede modificar los valores de los parámetros `dataimportfilename`, `dataexportfilename`, `importdatajobname`, `exportdatajobname`, `br_clear`, `br_calculatebalancesheet` y `br_allocatetarget` para ajustar los requisitos

Consulte Automatización de la ejecución de scripts para obtener información sobre la programación del script con Programador de tareas de 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

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%
```

Escenarios de ejemplo para Account Reconciliation

Consulte también:

- Carga de balances preformateados en un periodo
  Utilice estos scripts para importar los datos asignados de un archivo cargado a un entorno de Account Reconciliation.
- Carga e importación de una instantánea de copia de seguridad
  Utilice estos scripts para cargar e importar una instantánea de copia de seguridad en un entorno de Account Reconciliation.
- Archivado de transacciones confrontadas antiguas y depuración de transacciones archivadas
  Utilice los scripts de esta sección para archivar las transacciones confrontadas, incluidos los detalles de ajuste y soporte, que son iguales o anteriores a una antigüedad especificada y, a continuación, depure las transacciones archivadas de Account Reconciliation. Las transacciones confrontadas archivadas se registran en un archivo ZIP.

Carga de balances preformateados en un periodo

Utilice estos scripts para importar los datos asignados de un archivo cargado a un entorno de Account Reconciliation.

Script de Windows de ejemplo

Cree un archivo denominado `runPreformattedBalances.ps1` copiando el siguiente script. Almacénelo en un directorio 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
```

Script de Linux/UNIX de ejemplo

Cree un archivo denominado `runPreformattedBalances.sh` copiando el siguiente script. Almacénelo en un directorio 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
```

Archivo input.properties de ejemplo

Para ejecutar los scripts `runPreformattedBalances`, cree el archivo `input.properties` y actualícelo con información para el entorno. Guarde el archivo en el directorio en el que está almacenado `runPreformattedBalances.sh` o `runPreformattedBalances.ps1`.

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
```

Tabla 3-23 Parámetros de input.properties

Parámetro	Descripción
`javahome`	Ubicación `JAVA_HOME`. Solo para Linux/UNIX.

Tabla 3-23 (Continuación) Parámetros de input.properties

Parámetro	Descripción
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`). Solo para Linux/UNIX.
`username`	Nombre de usuario de un administrador del servicio.
`password`	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.
`serviceURL`	URL del entorno que aloja la aplicación en la que desea cargar los balances preformateados
`dataFile`	Archivo CSV que contiene los balances preformateados (normalmente creados a partir de un libro mayor) que desea cargar en la aplicación. Este archivo se debe haber cargado en el entorno utilizando el comando uploadFile.
`period`	Período de conciliación en el que se van a cargar los balances preformateados.
`balanceType`	Tipo de balances preformateados contenidos en `dataFile`.
`currencyBucket`	Depósito de monedas para los balances preformateados.

Ejecución del script

1. Cree `runPreformattedBalances.ps1` o `runPreformattedBalances.sh` copiando el script de una sección anterior.
2. **Solo para Windows y Linux/UNIX:**
   - Cree el archivo `input.properties` y guárdelo en el directorio en el que está ubicado el script `runPreformattedBalances`. El contenido de este archivo difiere según su sistema operativo. Consulte Tabla 1.
     Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar el script.
   - Inicie el script.
     - **Windows PowerShell:** ejecute `runPreformattedBalances.ps1`.
     - **Linux/UNIX:** ejecute `./runPreformattedBalances.sh`.

Carga e importación de una instantánea de copia de seguridad

Utilice estos scripts para cargar e importar una instantánea de copia de seguridad en un entorno de Account Reconciliation.

Script de Windows de ejemplo

Cree un archivo denominado `importBackupSnapshot.ps1` copiando el siguiente script. Almacénelo en un directorio 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
```

Script de Linux/UNIX de ejemplo

Cree un archivo denominado `importBackupSnapshot.sh` copiando el siguiente script. Almacénelo en un directorio 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
```

Archivo input.properties de ejemplo

Para ejecutar los scripts `importBackupSnapshot`, cree el archivo `input.properties` y actualícelo con información para el entorno. Guarde el archivo en el directorio en el que está almacenado `importBackupSnapshot.sh` o `importBackupSnapshot.ps1`.

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
```

Tabla 3-24 Parámetros de input.properties

Parámetro	Descripción
`javahome`	Ubicación `JAVA_HOME`. Solo para Linux/UNIX.
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`). Solo para Linux/UNIX.

Tabla 3-24 (Continuación) Parámetros de input.properties

Parámetro	Descripción
`username`	Nombre de usuario de un administrador del servicio.
`password`	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.
`serviceURL`	URL del entorno en el que desea importar la instantánea.
`snapshotName`	Nombre de la instantánea desde la que se van a importar los artefactos y los datos. Esta instantánea se debe haber cargado en el entorno utilizando el comando uploadFile.
`userPassword`	Contraseña predeterminada que se debe asignar a todos los nuevos usuarios creados en el dominio de identidad como resultado de esta importación de instantánea.

Ejecución del script

1. Cree `importBackupSnapshot.ps1` o `importBackupSnapshot.sh` copiando el script de una sección anterior.
2. Cree el archivo `input.properties` y guárdelo en el directorio en el que está ubicado el script `runPreformattedBalances`. El contenido de este archivo difiere según su sistema operativo. Consulte Archivo input.properties de ejemplo.
   Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar el script.
3. Inicie el script.
   - **Windows PowerShell:** ejecute `importBackupSnapshot.ps1`.
   - **Linux/UNIX:** ejecute `./importBackupSnapshot.sh`.

Archivado de transacciones confrontadas antiguas y depuración de transacciones archivadas

Utilice los scripts de esta sección para archivar las transacciones confrontadas, incluidos los detalles de ajuste y soporte, que son iguales o anteriores a una antigüedad especificada y, a continuación, depure las transacciones archivadas de Account Reconciliation. Las transacciones confrontadas archivadas se registran en un archivo ZIP.

Funcionamiento del script

1. Utilizando la información del archivo `input.properties`, inicia sesión en un entorno.
2. Ejecuta el siguiente comando de `archiveTmTransactions` para crear un archivo. El archivo ZIP y el archivo de registro resultantes utilizan los nombres predeterminados `Archive_Transactions_INTERCO_JOB_ID.zip` y `Archive_Transactions_INTERCO_JOB_ID.log`
   ```
   epmautomate archiveTmTransactions INTERCO 365 filterOperator=contains
   filterValue=14001
   ```
   Puede cambiar los parámetros del comando modificando el archivo `input.properties`.
3. Descargue el archivo de registro y el archivo ZIP que contiene las transacciones archivadas en el equipo local. El script muestra un mensaje de error si no se encuentra ninguna transacción de confrontación.

4. Copia el archivo ZIP que contiene las transacciones archivadas en Oracle Object Store.
5. Ejecuta el comando `purgeArchivedTmTransactions` (con el identificador de trabajo del trabajo archiveTmTransactions) para suprimir las transacciones confrontadas archivadas de la aplicación.

Ejecución del script

1. Cree el archivo `input.properties` y actualícelo con información de su entorno. Guarde el archivo en un directorio local. Este directorio se denomina `parentsnapshotdirectory` en esta discusión. El contenido de este archivo difiere según su sistema operativo. Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar los scripts.
2. Cree el script `transaction_match.ps1` (Windows PowerShell) o `transaction_match.sh` (Linux/UNIX) y guárdelo en `parentsnapshotdirectory`, donde se ubica `input.properties`.
3. Inicie el script.
   - Linux/UNIX: ejecute `./transaction_match.sh`.
   - Windows PowerShell: ejecute `transaction_match.ps1`.

Creación del script input.properties

Cree `input.properties` copiando y actualizando el siguiente script.

```
javahome=JAVA_HOME
epmautomatescript=EPM_AUTOMATE_LOCATION
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
```

Nota:

Solo para Windows: elimine estas propiedades del archivo `input.properties`:

- `javahome=JAVA_HOME`
- `epmautomatescript=EPM_AUTOMATE_LOCATION`

Si `authentication at proxy server` no está activado para el entorno de red de Windows, elimine estas propiedades del archivo `input.properties`.

- `proxyserverusername`
- `proxyserverpassword`
- `proxyserverdomain`

Tabla 3-25 Parámetros de input.properties

Parámetro	Descripción
`javahome`	Directorio en el que se instala la instancia de JDK que utiliza EPM Automate. Suprima esta entrada de la versión para Windows de `input.properties`. **Ejemplo:** `javahome=./home/JDK/bin`
`epmautomatescript`	Directorio donde se instala EPM Automate. Suprima esta entrada de la versión para Windows de `input.properties`. **Ejemplo:**`epmautomatescript=./home/utils/EPMAutomate/bin`
`epmusername`	Nombre de usuario de un administrador del servicio o un usuario avanzado, usuario o visor que está autorizado para archivar transacciones confrontadas. **Ejemplo:** `epmusername=ServiceAdmin`
`epmuserpassword`	Archivo de contraseña cifrado del usuario identificado como `epmusername`. **Ejemplo:** `epmpassword=myPwd.epw`
`epmurl`	URL del entorno donde se archivarán las transacciones confrontadas. **Ejemplo:** `epmurl=https://test-cloudpln.pbcs.us1.oraclecloud.com`
`objectstorageusername`	Identificador de un usuario que tiene los derechos de acceso necesarios para escribir en Oracle Object Storage Cloud. Para los usuarios creados en un proveedor de identidad federado, especifique el nombre completo del usuario (por ejemplo, `exampleIdP/jdoe` o `exampleIdP/john.doe@example.com`, donde `exampleIdP` es el nombre del proveedor de identidad federado). Para otros usuarios, especifique el ID de usuario. **Ejemplo:** `epmusername=myIdP/jdoe`

Tabla 3-25 (Continuación) Parámetros de input.properties

Parámetro	Descripción
`objectstoragepassword`	La contraseña Swift o el símbolo de autenticación asociado al usuario identificado en `objectstorageusername`. Esta contraseña no es la misma que la que utiliza el usuario para iniciar sesión en la consola de Object Storage. El símbolo de autentificación es un símbolo generado por Oracle que usa para autentificarse en las API de terceros, por ejemplo, para autentificarse en un cliente de Swift. Para obtener instrucciones para crear este símbolo, consulte Para crear un símbolo de autentificación en *Documentación de Oracle Cloud Infrastructure*. **Ejemplo:** `objectstoragepassword=jDoe_PWD`
`objectstorageurl`	URL del cubo de Oracle Object Storage Cloud con un nombre de objeto opcional añadido. **Ejemplo:** `objectstorageurl=https://swiftobjectstorage.region_identifier.oraclecloud.com/v1/namespace/MT_Archives/2023_archives`
`matchtype`	Identificador (TextID) del tipo de confrontación desde el que se deben archivar las transacciones confrontadas. **Ejemplo:** `matchtype=cashrecon`
`age`	Número de días desde que se confrontó la transacción. Se archivará una transacción coincidente anterior o igual a este valor. **Ejemplo:** `age=180`
`filteroperator`	Condiciones de filtro para identificar las cuentas que contienen transacciones confrontadas para su archivado. Debe ser una de las siguientes: `equals`, `not_equals`, `starts_with`, `ends_with`, `contains` o `not_contains`. Este valor se combina con `filterValue` para identificar las cuentas de las que deben archivarse las transacciones confrontadas: **Ejemplo:** `filteroperator=not_equals`
`filtervalues`	Uno o más valores de filtro para identificar las transacciones que se van a archivar. Si se especifica `equals` o `not_equals` como `filterOperator`, puede utilizar una lista separada por espacio para especificar varios valores. Si se especifican varios valores, se seleccionan las transacciones de las cuentas que coincidan con cualquier combinación de valor de filtro y operador de filtro para su archivado. **Ejemplo:** `filterValue=101-120 filterValue=140-202`
`proxyserverusername`	Nombre de usuario utilizado para autentificar una sesión segura con el servidor proxy que controla el acceso a Internet. **Ejemplo:** `proxyserverusername=myProxyserver`
`proxyserverpassword`	Contraseña para autentificar el usuario con el servidor proxy. **Ejemplo:** `proxyserverpassword=myProxyserver_pwd`
`proxyserverdomain`	Nombre del dominio definido para el servidor proxy. **Ejemplo:** `proxyserverdomain=myProxyDomain`

Script de Windows

Cree `transaction_match.ps1` copiando el siguiente script. Almacénelo en la carpeta en la que está almacenado `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
$transactionslogfound=Select-String -Path "./files.txt.tmp" -Pattern "$
{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."
```

Script de Linux/UNIX

Cree `transaction_match.sh` copiando el siguiente script. Almacénelo en la carpeta en la que está almacenado `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 ]
        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."
```

Escenarios de ejemplo para Profitability and Cost Management

En estos escenarios se analiza el uso de comandos de EPM Automate para realizar algunas tareas comunes de Profitability and Cost Management.

Consulte también:

- Importación de metadatos en la aplicación
  Utilice estos scripts para cargar un archivo de metadatos e importar metadatos de dimensión de este en la aplicación de Profitability and Cost Management.
- Importación de datos y ejecución de reglas del programa
  Utilice estos scripts para cargar archivos de datos e importar datos de los archivos cargados en un proceso de negocio de Profitability and Cost Management.

Importación de metadatos en la aplicación

Utilice estos scripts para cargar un archivo de metadatos e importar metadatos de dimensión de este en la aplicación de Profitability and Cost Management.

Estos scripts realizan las siguientes operaciones:

- Iniciar sesión en el entorno.
- Cargar un archivo de metadatos.
- Importar metadatos del archivo cargado en una aplicación.
- Activar la aplicación.
- Cerrar sesión.

Script de Windows

Cree `importMetadata.ps1` copiando este script.

```
$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 de Linux/UNIX

Cree `importMetadata.sh` copiando este script.

```
#!/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
```

Creación del archivo input.properties

Para ejecutar los scripts `importMetadata`, cree el archivo `input.properties` y actualícelo con información para el entorno. Guarde el archivo en el directorio en el que está almacenado `importMetadata.ps1` o `importMetadata.sh`.

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
serviceURL=exampleURL
applicationName=APPLICATION_NAME
dataFileName=DATA_FILE.txt
dataFileNameDestination=profitinbox
```

Tabla 3-26 Parámetros de input.properties

Parámetro	Descripción
`javahome`	Ubicación `JAVA_HOME`. Solo para Linux/UNIX.
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`). Solo para Linux/UNIX.
`username`	Nombre de usuario de un administrador del servicio que también tiene asignado el rol de administrador del dominio de identidad.

Tabla 3-26 (Continuación) Parámetros de input.properties

Parámetro	Descripción
`password`	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.
`serviceURL`	URL del entorno a partir del cual desea generar la instantánea.
`applicationName`	Nombre de la instancia de Profitability and Cost Management en la que se cargarán los datos.
`dataFileName`	Nombre del archivo que contiene los metadatos que se van a importar.
`dataFileNameDestination`	Ubicación de carga del archivo de metadatos.

Ejecución de los scripts

1. Cree `importMetadata.ps1` o `importMetadata.sh` copiando el script de una sección anterior.
2. Cree el archivo `input.properties` y guárdelo en el directorio en el que está ubicado el script `importMetadata`. El contenido de este archivo difiere según su sistema operativo. Consulte Creación del archivo input.properties.
   Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar el script.
3. Inicie el script.
   - **Windows PowerShell:** ejecute `importMetadata.ps1`.
   - **Linux/UNIX:** ejecute `./importMetadata.sh`.

Importación de datos y ejecución de reglas del programa

Utilice estos scripts para cargar archivos de datos e importar datos de los archivos cargados en un proceso de negocio de Profitability and Cost Management.

Estos scripts completan los siguientes pasos:

- Inicie sesión en el entorno.
- Cargar un archivo de datos que contiene los datos que se van a cargar.
- Cargar un archivo de reglas que contiene reglas de datos.
- Cargar datos del archivo de datos en la aplicación para sobrescribir los valores existentes.
- Ejecutar todas las reglas en el archivo de reglas.
- Cerrar sesión.

Script de Windows

Cree un archivo denominado `importData.ps1` copiando el siguiente script. Almacénelo en un directorio 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 de Linux/UNIX

Cree un archivo denominado `importData.sh` copiando el siguiente script. Almacénelo en un directorio 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
```

Creación del archivo input.properties

Para ejecutar los scripts `importData`, cree el archivo `input.properties` y actualícelo con información para el entorno. Guarde el archivo en el directorio en el que está almacenado `importData.ps1` o `importData.sh`.

Windows

```
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
```

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
```

Tabla 3-27 Parámetros de input.properties

Parámetro	Descripción
`javahome`	Ubicación `JAVA_HOME`. Solo para Linux/UNIX.
`epmautomatescript`	Ruta de acceso absoluta del ejecutable de EPM Automate (`epmautomate.sh`). Solo para Linux/UNIX.
`username`	Nombre de usuario de un administrador del servicio que también tiene asignado el rol de administrador del dominio de identidad.
`password`	Contraseña del administrador de servicio o el nombre y la ubicación del archivo de contraseña cifrado.
`serviceURL`	URL del entorno a partir del cual desea generar la instantánea.
`applicationName`	Nombre de la instancia de Profitability and Cost Management en la que se cargarán los datos.
`dataFileName`	Nombre del archivo que contiene los datos que se van a importar.
`rulesFileName`	Nombre del archivo que contiene las reglas que se van a importar.
`fileDestination`	Ubicación en la que se cargarán los archivos de datos y de reglas.
`clearAllDataFlag`	Especifica si se deben borrar los datos existentes en el cubo de aplicación. Defínalo en `false` si no desea borrar los datos existentes.
`dataLoadValue`	Especifica cómo gestionar los datos existentes. Especifique `ADD_TO_EXISTING` si desea conservar los datos existentes en el cubo.

Ejecución de los scripts

1. Cree `importData.ps1` o `importData.sh` copiando el script de una sección anterior.
2. Cree el archivo `input.properties` y guárdelo en el directorio en el que está ubicado el script `importData`. El contenido de este archivo difiere según su sistema operativo. Consulte Creación del archivo input.properties.
   Asegúrese de que tiene privilegios de escritura en este directorio. Para Windows, puede que necesite iniciar PowerShell mediante la opción **Ejecutar como administrador** para poder ejecutar el script.
3. Inicie el script.
   - **Windows PowerShell:** ejecute `importData.ps1`.
   - **Linux/UNIX:** ejecute `./importData.sh`.

Escenarios de ejemplo para Oracle Enterprise Data Management Cloud

En estos escenarios de ejemplo se explora el uso de comandos de EPM Automate para sincronizar dimensiones de aplicación entre Oracle Fusion Cloud Enterprise Data Management y Oracle Fusion Cloud Enterprise Performance Management.

Consulte también:

- Sincronización de dimensiones y asignaciones de Oracle Enterprise Data Management Cloud con aplicaciones de Cloud EPM
  En este escenario de ejemplo se analiza la sincronización de una dimensión entre una aplicación de Oracle Fusion Cloud Enterprise Data Management y una aplicación de Oracle Fusion Cloud Enterprise Performance Management.
- Sincronización de dimensiones de Cloud EPM con aplicaciones de Oracle Enterprise Data Management Cloud
  En este escenario de ejemplo se analiza la sincronización de una dimensión entre una aplicación de Oracle Fusion Cloud Enterprise Performance Management y una aplicación de Oracle Fusion Cloud Enterprise Data Management.

Sincronización de dimensiones y asignaciones de Oracle Enterprise Data Management Cloud con aplicaciones de Cloud EPM

En este escenario de ejemplo se analiza la sincronización de una dimensión entre una aplicación de Oracle Fusion Cloud Enterprise Data Management y una aplicación de Oracle Fusion Cloud Enterprise Performance Management.

Utilice los scripts de esta sección para completar las siguientes tareas:

- Exportar una dimensión desde una aplicación de Oracle Enterprise Data Management Cloud
- Exportar asignaciones desde una dimensión de aplicación de Oracle Enterprise Data Management Cloud
- Copiar archivos de exportación a un entorno de EPM Cloud
- Importar metadatos y asignaciones de dimensión en la aplicación de Cloud EPM

Para sincronizar una dimensión y asignaciones entre una aplicación de Oracle Enterprise Data Management Cloud y una aplicación de Cloud EPM:

1. Cree un archivo de script copiando el siguiente script:

```
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
```

```
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. Modifique el archivo de script y defina los valores de parámetro necesarios. Consulte Parámetros para la ejecución de scripts para obtener una explicación y un ejemplo de los parámetros.
3. Ejecute el script de forma manual o prográmelo para que se ejecute según sea necesario. Consulte Automatización de la ejecución de scripts.

Parámetros para la ejecución de scripts

Los archivos de script de esta sección requieren que se especifiquen algunos de los valores de parámetro que se explican en la siguiente tabla. No todos estos parámetros se utilizan en todos los scripts.

Tabla 3-28 Valores de parámetro para archivos de script

Parámetro	Descripción
EDMUSER	Identificador de inicio de sesión de usuario de un administrador del servicio de Oracle Enterprise Data Management Cloud. **Ejemplo:** EDMUSER=jdoe@example.com
EDMSVR	URL del entorno de Oracle Enterprise Data Management Cloud. **Ejemplo:** EDMSVR=https:// example.oraclecloud.com

Tabla 3-28 (Continuación) Valores de parámetro para archivos de script

Parámetro	Descripción
EDMPWDFILE	Nombre y ubicación del archivo de contraseña cifrado (EPW) del administrador del servicio de Oracle Enterprise Data Management Cloud. **Ejemplo:** EDMPWDFILE=edm_jdoe.epw
EDMAPP	Nombre de una dimensión de aplicación de Oracle Enterprise Data Management Cloud. **Ejemplo:** EDMAPP=USOperations
EDMDIM	Nombre de la dimensión que se va a exportar o importar. **Ejemplo:** EDMDIM=entity
EDMLOC	Nombre de la ubicación que se va a exportar. **Ejemplo:** EDMLOC=Loc1
EPMUSER	Nombre de inicio de sesión de un administrador del servicio de Cloud EPM. **Ejemplo:** EPMUSER=john.doe@example.com
EPMSVR	URL del entorno de Cloud EPM. **Ejemplo:** EPMSVR=https://example.oraclecloud.com
EPMIMPJOB	Nombre de un trabajo de importación existente del tipo import metadata en el entorno de Cloud EPM. **Ejemplo:** EPMIMPJOB=imp_DIMMetadata
EPMEXPJOB	Nombre de un trabajo existente del tipo export metadata en el entorno de Cloud EPM. **Ejemplo:**EPMEXPJOB=Exp_DIMMetadata
PWDFILE	Nombre y ubicación del archivo de contraseña cifrado (EPW) para el administrador del servicio de Cloud EPM. Consulte el comando encrypt. **Ejemplo:** PWDFILE=pwd_jdoe.epw
DIMFILE	Nombre del archivo que va a contener los datos de dimensión exportados. **Ejemplo:** DIMFILE=entity_file.CSV
MAPFILE	Nombre del archivo que va a contener los datos de asignación exportados. **Ejemplo:** MAPFILE=map_file.CSV

Sincronización de dimensiones de Cloud EPM con aplicaciones de Oracle Enterprise Data Management Cloud

En este escenario de ejemplo se analiza la sincronización de una dimensión entre una aplicación de Oracle Fusion Cloud Enterprise Performance Management y una aplicación de Oracle Fusion Cloud Enterprise Data Management.

Utilice los scripts de esta sección para completar las siguientes tareas:

- Exportar metadatos (dimensiones) desde una aplicación de Cloud EPM
- Copiar los archivos de exportación que contienen datos de dimensión en un entorno de Oracle Enterprise Data Management Cloud

- Importar metadatos de dimensión en la aplicación de Oracle Enterprise Data Management Cloud

Para sincronizar una dimensión entre una aplicación de Cloud EPM y una aplicación de Oracle Enterprise Data Management Cloud:

1. Cree un archivo de script de Windows copiando el siguiente script:

```
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
```

   Modifique el archivo de script y defina los valores de parámetro necesarios. Consulte Parámetros para la ejecución de scripts para obtener una explicación y un ejemplo de los parámetros.

2. Ejecute el script de forma manual o prográmelo para que se ejecute según sea necesario. Consulte Automatización de la ejecución de scripts.

Automatización de la ejecución de scripts

Un administrador de servicio programa scripts en el programador de tareas de Windows o utiliza un trabajo cron para automatizar actividades con EPM Automate.

Para programar la ejecución de scripts de EPM Automate con el Programador de tareas de Windows:

1. Haga clic en **Inicio**, en **Panel de control** y, a continuación, en **Herramientas administrativas**.
2. Abra **Programador de tareas**.
3. Seleccione **Acción** y, a continuación, **Crear tarea básica**.
4. Introduzca un nombre de tarea y una descripción opcional y, a continuación, haga clic en **Siguiente**.
5. En **Disparador de tarea**, seleccione un programa para ejecutar el script y, a continuación, haga clic en **Siguiente**.
6. En la siguiente pantalla, especifique otros parámetros de programación y, a continuación, haga clic en **Siguiente**.
7. En Acción, asegúrese de que está seleccionada la opción **Iniciar un programa**.
8. En **Iniciar un programa**, realice los siguientes pasos:
   a. En **Programa/script** seleccione el script que desea programar.
   b. En **Agregar argumentos (opcional)**, introduzca la contraseña del administrador de servicio identificado por el parámetro de script `SET user`.
   c. En **Empezar en (opcional)**, introduzca la ubicación en la que está instalado EPM Automate, normalmente, en `C:/Oracle/EPMAutomate/bin`.
   d. Haga clic en **Siguiente**.
9. En **Resumen**, seleccione **Abrir el diálogo Propiedades para esta tarea al hacer clic en Finalizar** y, a continuación, haga clic en **Finalizar**.
10. En **General**, seleccione estas opciones de seguridad y, a continuación, haga clic en **Aceptar**.
    - **Ejecutar si el usuario está o no conectado**
    - **Ejecutar con los privilegios más altos**

Supervisión de las actividades de EPM Automate

Para ayudarle a identificar el estado de la operación que ha inicializado, EPM Automate muestra los códigos de estado en la consola desde la que la ejecuta.

Consulte Códigos de salida.

Utilice la consola de trabajos para supervisar los trabajos que ejecute mediante el uso de EPM Automate. Consulte Gestión de trabajos en *Administración de Planning* para obtener más información.

4

Ejecución de comandos sin instalar EPM Automate

Con Groovy, puede ejecutar comandos específicos directamente en Oracle Fusion Cloud Enterprise Performance Management. No necesita una instalación de EPM Automate para ejecutar comandos en el servidor.

> **Note:**
>
> En este escenario, el script de Groovy se escribe para ejecutarse directamente en Cloud EPM, no en ninguna máquina cliente.

Contenido del capítulo:

- Entornos que soportan la ejecución de comandos en el servidor
- Fuentes de información
- Comandos soportados
- Métodos que usar para ejecutar EPM Automate con un script Groovy del servidor
- Clonación de un entorno con un script de Groovy del servidor
- Envío por correo electrónico de un informe de actividad con un script de Groovy del servidor

Entornos que soportan la ejecución de comandos en el servidor

En los scripts de Groovy está soportada la ejecución en el servidor de comandos de EPM Automate solo en los siguientes entornos:

- Procesos de negocio de Planning y Planning Modules desplegados en entornos de EPM Enterprise Cloud Service.
- Enterprise Planning and Budgeting Cloud
- Planning and Budgeting Cloud con opción Plus One
- Formato libre
- Enterprise Profitability and Cost Management
- Financial Consolidation and Close en entornos de EPM Enterprise Cloud Service.
- Tax Reporting en entornos de EPM Enterprise Cloud Service.
- Strategic Workforce Planning
- Sales Planning

Los scripts de Groovy que incorporen comandos de EPM Automate se pueden escribir y ejecutar solo en entornos anteriores de Oracle Fusion Cloud Enterprise Performance Management. Los scripts escritos en esos entornos pueden, no obstante, emitir comandos de

EPM Automate en cualquier entorno de Cloud EPM. Por ejemplo, puede crear el script en un entorno de Planning EPM Enterprise Cloud Service y hacer que emita comandos en un entorno de Narrative Reporting que no soporte scripts de Groovy.

Fuentes de información

Consulte estas fuentes de *Diseño con Calculation Manager* para obtener información detallada:

- Acerca de las reglas de negocio de Groovy
- Creación de una regla de negocio de Groovy

Comandos soportados

Excepto en el caso de los siguientes, todos los comandos de EPM Automate se pueden ejecutar mediante Groovy:

- downloadFile
- upgrade
- uploadFile

Los siguientes comandos no se pueden ejecutar en el entorno en el que se ejecute el script de Groovy:

- recreate
- replay
- resetService
- restructureCube

> **Note:**
>
> - Al ejecutar el comando encrypt, el archivo de contraseñas cifradas se crea en el servidor y se conserva para su uso a largo plazo. El archivo de contraseñas cifradas no se muestra si ejecuta el comando listFiles . No puede descargarlo del servidor mediante el comando downloadFile .
> - Para que el comando feedback funcione, todos los archivos y las capturas de pantalla que se usen como datos adjuntos deben estar disponibles en la ubicación de carga predeterminada. Consulte Ubicación de carga predeterminada. Esta es la ubicación donde se almacenan los archivos si no especifica una en el comando uploadFile.

Métodos que usar para ejecutar EPM Automate con un script Groovy del servidor

- `getEPMAutomate ()` Este método estático proporciona una instancia de la clase `EpmAutomate`, que a continuación se puede usar para llamar a los comandos de EPM Automate.

- `execute ()` Este método de la clase `EpmAutomate` se usa para ejecutar un comando EPM Automate. Transfiere el nombre del comando de EPM Automate como primer parámetro y las opciones de comando como parámetros posteriores. Este método devuelve una instancia de clase `EpmAutomateStatus`.
- `getStatus ()` Este método de la clase `EPMAutomateStatus` devuelve el estado de ejecución que devuelve el comando. El valor de retorno 0 indica un resultado correcto, mientras que un valor que no sea cero indica un fallo del comando.
- `getOutput ()` Este método de la clase `EPMAutomateStatus` devuelve la salida en formato cadena del comando. Por ejemplo, puede usar este método para devolver la salida del comando `getApplicationAdminMode` y `getDailyMaintenanceStartTime`. Si el estado de devolución del comando no es cero, este método devuelve el mensaje de error.
- `getItemsList ()` Este método de la clase `EPMAutomateStatus` devuelve la salida en formato lista del comando. Por ejemplo, puede usar este método para devolver la salida de los comandos `getSubstVar`, `listBackups` y `listFiles`.

Clonación de un entorno con un script de Groovy del servidor

Puede incluir comandos de EPM Automate en scripts de Groovy del servidor para clonar entornos para la recuperación ante desastres. Esto permite configurar la recuperación ante desastres sin ninguna huella local.

Si las contraseñas contienen caracteres especiales, consulte Manejo de caracteres especiales Asegúrese también de reemplazar estos valores de parámetros para que se ajusten a su entorno:

Table 4-1 Parámetros que cambiar

Parámetro	Descripción
`password`	Contraseña del administrador del servicio que está realizando la operación de clonación en el entorno de origen.
`targetpassword`	Contraseña del administrador del servicio que está realizando la operación de clonación en el entorno de destino.
`username`	ID de usuario del administrador del servicio en el entorno de origen.
`targetusername`	ID de usuario del administrador del servicio en el entorno de destino. Este usuario también tiene que tener asignado el rol de administrador de dominio en el entorno de destino.
`email_id`	Dirección de correo electrónico a la que tiene previsto enviar información sobre el proceso de clonación.

Script para cifrar la contraseña

```
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())
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 para clonar el entorno

Este script usa los archivos de contraseñas cifradas creados con el script anterior.

```
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())
```

Envío por correo electrónico de un informe de actividad con un script de Groovy del servidor

Este script se puede usar para enviar por correo electrónico el informe de actividad a una lista de destinatarios. Tras esto, el script se puede programar para que se ejecute a diario con el fin de obtener el informe de actividad diario. Con este script se realizan las siguientes funciones:

- Cifra la contraseña del administrador del servicio que ejecuta el script de Groovy.
- Se conecta al entorno de Oracle Fusion Cloud Enterprise Performance Management con la contraseña cifrada.
- Envía por correo electrónico el informe de actividad disponible en el entorno a una lista de destinatarios, generalmente, administradores del servicio
- Cierra la sesión en el entorno.

Si las contraseñas contienen caracteres especiales, consulte Manejo de caracteres especiales Asegúrese también de reemplazar estos valores de parámetros para que se ajusten a su entorno:

Table 4-2 Parámetros que cambiar

Parámetro	Descripción
user	ID de usuario del administrador del servicio que inicia sesión en el entorno.
password	Contraseña del administrador del servicio.
url	URL del entorno de Cloud EPM desde el que se va a enviar por correo electrónico el informe de actividad.
emailaddresses	Una lista de direcciones de correo electrónico separada por puntos y comas a la que se va a enviar el informe de actividad.

Para obtener información detallada sobre el uso de reglas de Groovy, consulte Uso de reglas Groovy en *Administración de Planning*.

```
/*RTPS: {user} {password} {url} {emailaddresses}*/
import java.text.SimpleDateFormat

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)
}

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)
```

5
Replicación de un entorno de Cloud EPM

Estos pasos están implicados en la configuración de un entorno secundario de Oracle Fusion Cloud Enterprise Performance Management para garantizar la disponibilidad del servicio, si el centro de datos principal de Oracle deja de estar disponible debido a circunstancias imprevistas.

> **Nota:**
>
> Los procedimientos descritos en este apéndice no se pueden aplicar a Narrative Reporting.

- Configuración de la replicación de artefactos diaria
- Configuración de replicaciones a petición
- Configuración del entorno secundario

Configuración de la replicación diaria

Para replicar un entorno, use EPM Automate para copiar la instantánea del artefacto creada durante el mantenimiento diario del entorno principal al entorno secundario.

Oracle realiza un mantenimiento rutinario en cada entorno de forma diaria. Durante este mantenimiento del servicio, Oracle crea una instantánea de mantenimiento realizando una copia de seguridad del contenido del entorno (datos y artefactos existentes, incluidas las asignaciones de roles y usuarios del dominio de identidad).

Para configurar la replicación del servicio diaria:

1. Cree un archivo de script que contenga los siguientes comandos de EPM Automate. Este script replica la instantánea de la aplicación del entorno principal en el entorno secundario.

   > **Nota:**
   >
   > Asegúrese de cambiar el nombre de usuario, el archivo de contraseñas, los nombres de dominio de identidad y las URL de servicio. Para obtener más información sobre la creación de un archivo de contraseña cifrado, consulte el comando 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
epmautomate copysnapshotfrominstance "Artifact Snapshot"
```

```
primaryPassword.epw primary_URL primaryDomain
REM Sign out of the secondary instance
epmautomate logout
```

2. Con un programador, por ejemplo, el programador de tareas de Windows, programe la ejecución del archivo de script para que se ejecute dos horas después del inicio del periodo de mantenimiento.
3. Establezca una hora de inicio del periodo de mantenimiento idéntica en los entornos principal y secundario. Consulte Establecimiento del tiempo de mantenimiento de servicio en *Guía de introducción para administradores* para obtener más información.

Configuración de replicaciones a petición

Para reducir el RPO, puede crear instantáneas del entorno principal a petición y, a continuación, copiarlas en el entorno secundario.

Por ejemplo, puede crear y programar un script de EPM Automate que se ejecute cada seis horas entre las replicaciones diarias para reducir el RPO de 24 a seis horas.

> **Nota:**
>
> Durante la creación de instantáneas a petición, el entorno principal se coloca en modo de solo lectura durante unos minutos.

Para configurar la replicación a petición:

1. Cree un archivo de script que contenga los siguientes comandos de EPM Automate. Este script replica la instantánea de la aplicación del entorno principal en el entorno secundario.

   > **Nota:**
   >
   > Asegúrese de cambiar el nombre de usuario, el archivo de contraseñas, los nombres de dominio de identidad y las URL de servicio. Para obtener más información sobre la creación de un archivo de contraseña cifrado, consulte el comando 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
```

```
REM Sign out of the secondary instance
epmautomate logout
```

2. Con un programador, por ejemplo, el programador de tareas de Windows, programe la ejecución del archivo de script para que se ejecute como sea necesario para cumplir con el RPO que desee.

Configuración del entorno secundario

Configure el entorno secundario para activarlo.

Realice este procedimiento solo si necesita activar el entorno secundario cuando el entorno principal no esté disponible durante un largo periodo. Antes de configurar el entorno secundario, consulte estos temas en *Guía de introducción para administradores*:

- Rutas de migración para instantáneas heredadas de EPM Cloud
- Rutas de migración para instantáneas de EPM Standard Cloud Service y de EPM Enterprise Cloud Service

Para configurar el entorno secundario:

1. Inicie una sesión de EPM Automate y realice estas actividades.
   - Inicie sesión en el entorno secundario con una cuenta que tenga los roles de administrador del servicio y administrador de dominio de identidad. Asegúrese de especificar el nombre de usuario, la contraseña, el nombre de dominio y la URL de servicio.
   - Vuelva a crear el entorno secundario.
     - Si el entorno principal es un entorno de Planning, Tax Reporting, Financial Consolidation and Close o Enterprise Profitability and Cost Management, use: `epmautomate recreate -f`
     - Si el entorno principal no es un entorno de Planning, Tax Reporting, Financial Consolidation and Close o Enterprise Profitability and Cost Management, use: `epmautomate recreate -f TempServiceType=`*`PRIMARY_APPLICATION_TYPE`*, donde *`PRIMARY_APPLICATION_TYPE`* es `ARCS`, `EDMCS`, `PCMCS` o `EPRCS`.
   - Importe artefactos de aplicación y dominio de identidad de la instantánea.
   - Cierre la sesión.

   Puede realizar las actividades anteriores ejecutando los siguientes comandos. Consulte estos temas:

   - Comando login
   - Comando recreate
   - Comando importSnapshot

   ```
   epmautomate login serviceAdmin secondaryPassword.epw secondary_URL
   epmautomate recreate -f
   epmautomate importsnapshot "Artifact Snapshot" importUsers=true
   epmautomate logout
   ```

2. Conéctese al entorno secundario y verifique que todos los datos están disponibles.
3. Envíe la URL del entorno secundario a todos los usuarios.

A

Preparación para ejecutar el comando simulateConcurrentUsage

El comando simulateConcurrentUsage soporta estas operaciones para simular la carga en un entorno:

- Abrir formularios
- Guardar formularios
- Ejecutar reglas de negocio
- Ejecutar conjuntos de reglas de negocio
- Ejecutar reglas de datos
- Abrir cuadrículas ad hoc
- Ejecutar libros de informes de gestión
- Ejecutar informes de informes de gestión

Pasos involucrados

1. Cree el archivo `requirement.csv`. Consulte Creación del archivo requirement.csv
2. Cree archivos de entrada especificando los detalles de las operaciones incluidas en `requirement.csv`. Consulte:
   - Archivo de entrada de apertura de formulario
   - Archivo de entrada de almacenamiento de formulario
   - Archivo de entrada de ejecución de regla de negocio
   - Archivo de entrada de ejecución de conjunto de reglas de negocio
   - Archivo de entrada de ejecución de regla de datos
   - Archivo de entrada de cuadrícula ad hoc
   - Archivo de entrada de ejecución de libro
   - Archivo de entrada de ejecución de informe
3. Cree `UserVarMemberMapping.csv` que contenga los detalles de variable de usuario si es necesario establecer la asignación de miembros de variable de usuario. Consulte Creación del archivo UserVarMemberMapping.csv
4. Opcionalmente, exporte las opciones de Oracle Smart View for Office en `options.xml` si es necesario utilizar las opciones de Smart View. Consulte Creación del archivo options.xml para obtener más información.
5. Opcionalmente, cree el archivo `users.csv` para ejecutar simulaciones con los usuarios existentes. Consulte Creación del archivo users.csv para obtener más información.
6. Cree un archivo ZIP que contenga los archivos que ha creado en los pasos anteriores y cárguelo en el entorno. Consulte Creación y carga del archivo ZIP de entrada en el entorno

7. Ejecute el comando simulateConcurrentUsage utilizando el archivo ZIP cargado.

Creación del archivo requirement.csv

Empiece por crear el archivo `requirement.csv` que muestra los detalles de los casos de uso que desea probar. Cada línea de este archivo CSV identifica el tipo de operación que se debe realizar, el nombre del artefacto, el número de usuarios simultáneos, el archivo de entrada que especifica los detalles de la operación y, si la hubiera, información adicional relacionada con la operación. Por ejemplo, para abrir 2 formularios, guardar 2 formularios y ejecutar 2 reglas de negocio, debe especificar 6 líneas en el archivo CSV de entrada. La primera línea de `requirement.csv` debe contener esta información:

```
#Type of Operation,Artifact Name,Number of Users,Input File,Additional Info
```

Cada una de las líneas siguientes del archivo contiene una única operación y sus parámetros. Es posible que algunas operaciones no requieran todos estos valores de parámetro. Las entradas de archivo esperadas se explican en la siguiente tabla.

> **Note:**
>
> Todos los valores son necesarios salvo que se indique lo contrario en la tabla.

Table A-1 Formato de requirement.csv

Campo	Descripción
Tipo de operación	Debe ser uno de los siguientes: • `Open Form` • `Save Form` • `Run Business Rule` • `Run Business Ruleset` • `Run Data Rule` • `Ad Hoc Grid` • `Execute Report` • `Execute Book`
Nombre de artefacto	Este valor depende del tipo de operación: • Open Form: nombre y ubicación del formulario que se va a abrir. • Save Form: nombre y ubicación del formulario que se va a guardar. • Run Business Rule: nombre de la regla de negocio. • Ejecutar conjunto de reglas de negocio: nombre del conjunto de reglas de negocio. • Data Rule: nombre de la regla de datos. • Ad Hoc Grid: no aplicable (dejar en blanco). • Execute Report: nombre y ubicación del informe. • Execute Book: nombre y ubicación del libro.
Número de usuarios	Número de usuarios para simular el usuario simultáneo.
Archivo de entrada	Nombre del archivo CSV que contiene los valores de PDV, las peticiones de datos en tiempo de ejecución u otros valores específicos del caso de uso que se van a utilizar
Información adicional	Parámetros adicionales necesarios para la operación. Aplicable solo a la cuadrícula ad hoc. Déjelo en blanco para otros casos de uso.

Table A-1 (Cont.) Formato de requirement.csv

Campo	Descripción
Notas: Los nombres de artefactos deben coincidir con los de la aplicación y tener las mismas mayúsculas/minúsculas.	

Ejemplo de un archivo `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,
```

Creación de archivos de entrada

Cada caso de uso identificado en `requirement.csv` debe tener un archivo de entrada coincidente que proporcione los parámetros necesarios para ejecutarlo.

El archivo de entrada debe contener normalmente una entrada por el número de usuarios especificado para este caso de uso en `requirement.csv`.

Si el número de entradas del archivo de entrada es inferior al número de usuarios simultáneos para ese caso de uso de `requirement.csv`, EPM Automate repite algunas entradas del archivo de entrada para ejecutar los casos de uso hasta que la operación se ejecute para el número de usuarios identificados en `requirement.csv`.

Por ejemplo, si la entrada del caso de uso en `requirement.csv` para la operación Ejecutar regla de negocio es la siguiente:

```
Run Business Rule, Copy Budget,10,br_input_file.csv,
```

`br_input_file.csv` también debe contener 10 entradas. Si `br_input_file.csv` contiene solo seis entradas, EPM Automate las utiliza para los 6 primeros usuarios. Para los siguientes 4 usuarios, EPM Automate reutiliza las primeras 4 entradas de `br_input_file.csv`.

Si el número de entradas del archivo de entrada es superior al número de usuarios especificado para el caso de uso, EPM Automate ignora las últimas entradas adicionales del archivo de entrada.

- Archivo de entrada de apertura de formulario
- Archivo de entrada de almacenamiento de formulario
- Archivo de entrada de ejecución de regla de negocio
- Archivo de entrada de ejecución de conjunto de reglas de negocio
- Archivo de entrada de ejecución de regla de datos
- Archivo de entrada de cuadrícula ad hoc

- Archivo de entrada de ejecución de informe
- Archivo de entrada de ejecución de libro

Archivo de entrada de apertura de formulario

Este archivo, al que se hace referencia en `requirement.csv` para soportar la apertura de formulario, incluye entrada de PDV con el formato: `pov=[DIM 1:MEMBER 1],[DIM 2:MEMBER 2],[DIM 3:MEMBER 3],`, etc.

En este análisis, DIM 1, DIM 2, etc. son los nombres de dimensión, y MEMBER 1, MEMBER 2, etc. son sus valores de miembro de dimensión para el PDV.

Archivo de entrada de ejemplo:

```
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:**
>
> También debe crear `UserVarMemberMapping.csv` si los formularios que ha especificado en `requirement.csv` requieren que se definan variables de usuario. Consulte Creación del archivo UserVarMemberMapping.csv.

Archivo de entrada de almacenamiento de formulario

Este archivo, al que se hace referencia en `requirement.csv` para soportar el almacenamiento de formularios, incluye PDV y valores de entrada de celda con el siguiente formato. Si se asocian reglas de negocio a este formulario, el archivo incluye también las peticiones de datos en tiempo de ejecución para dichas reglas de negocio:

Archivo de entrada de ejemplo:

```
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]]..
```

En este ejemplo:

- DIM indica el nombre de una dimensión, y MEMBER indica un valor de miembro de dimensión.
- CELL COLUMN HEADER identifica el nombre de la cabecera de columna, y CELL ROW HEADER identifica el nombre de una cabecera de fila.
- BUSINESS RULE NAME indica el nombre de la regla de negocio, RTP indica el nombre de la petición de datos en tiempo de ejecución y VALUE indica el valor. RTP no es

necesario si no hay ninguna regla de negocio asociada al formulario o si desea utilizar el valor de la petición de datos en tiempo de ejecución predeterminado presente en el formulario.

Por ejemplo:

```
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]]
```

> **Note:**
>
> También debe crear `UserVarMemberMapping.csv` si los formularios que especifique en `requirement.csv` requieren que se definan variables de usuario. Consulte Creación del archivo UserVarMemberMapping.csv.

Los envíos inteligentes, si los hay, asociados al formulario se ejecutan de forma automática.

Archivo de entrada de ejecución de regla de negocio

Este archivo, al que se hace referencia en `requirement.csv` para soportar la ejecución de reglas de negocio, incluye valores de parámetro de tiempo de ejecución con el formato: `rtp=[PARAMETER:VALUE],[PARAMETER:VALUE]`, etc. En este formato, *`PARAMETER`* identifica un nombre de petición de datos en tiempo de ejecución y *`VALUE`* identifica su valor.

Asegúrese de agregar tantas peticiones de datos en tiempo de ejecución como sea necesario para ejecutar la regla. Se utilizan los valores predeterminados si no se proporcionan peticiones de datos en tiempo de ejecución y sus valores. El comando ignora las peticiones de datos en tiempo de ejecución que no coincidan exactamente con las definidas para regla. Si no es necesario ningún parámetro de tiempo de ejecución para la regla de negocio, incluya `rtp=[]`.

Archivo de entrada de ejemplo:

```
rtp=[Period:Q1],[Entity:USA]
rtp=[Period:Q2],[Entity:USA]
rtp=[Period:Q3],[Entity:USA]
rtp=[Period:Q4],[Entity:USA]
```

Archivo de entrada de ejecución de conjunto de reglas de negocio

Este archivo, al que se hace referencia en `requirement.csv` para soportar la ejecución de conjuntos de reglas de negocio, incluye valores de parámetro de tiempo de ejecución con el formato: `rtp=[PARAMETER:VALUE],[PARAMETER:VALUE]`, etc. En este formato, *`PARAMETER:VALUE`* identifica un nombre de parámetro de tiempo de ejecución y su valor.

Utilice el par *`PARAMETER:VALUE`* para especificar tantas peticiones de datos en tiempo de ejecución como requiera el conjunto de reglas. Se utilizan los valores predeterminados si no se proporcionan valores para los parámetros de tiempo de ejecución. El comando ignora las peticiones de datos en tiempo de ejecución que no coincidan exactamente con las definidas en el conjunto de reglas. Si no es necesario ningún valor de parámetro de tiempo de ejecución para el conjunto de reglas de negocio, incluya `rtp=[]`.

Archivo de entrada de ejemplo:

```
rpt=[Period:Q1],[Entity:USA]
rtp=[Period:Q2],[Entity:USA]
rtp=[Period:Q3],[Entity:USA]
rtp=[Period:Q4],[Entity:USA]
```

Archivo de entrada de ejecución de regla de datos

Este archivo, al que se hace referencia en `requirement.csv` para soportar la ejecución de reglas de datos, debe especificar el período de inicio, el período de finalización, el modo de importación, el modo de exportación y un nombre de archivo de importación opcional disponible en el entorno. Si no se especifica un nombre de archivo, se utiliza el nombre de archivo especificado en la regla de datos. Formato de cada línea:`startperiod=`*`START PERIOD`*`;endperiod=`*`END PERIOD`*`;importmode=IMPORT_MODE;exportmode=`*`EXPORT_MODE`*`;filename=`*`FILE NAME`*

Archivo de entrada de ejemplo:

```
startperiod=Dec-15;endperiod=Dec-15;importmode=REPLACE;exportmode=STORE_DATA;f
ilename=comma_delim_file1.csv
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
```

Archivo de entrada de cuadrícula ad hoc

El comando simulateConcurrentUsage soporta cuadrículas ad hoc tanto en modo nativo como en modo estándar. En una cuadrícula en modo nativo, los PDV se muestran como parte de la barra de herramientas del complemento Oracle Smart View for Office. En el modo estándar, los PDV forman parte de la propia hoja de cálculo y ocupan la primera fila de dicha hoja.

El archivo de entrada de cuadrícula ad hoc, al que se hace referencia en `requirement.csv` para soportar la apertura de cuadrículas ad hoc, debe especificar la cuadrícula que se debe abrir. Cada línea del archivo debe tener el siguiente formato.

```
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,..]
```

Archivo de entrada de ejemplo:

```
fileName=dropdown.xlsx#sheet4;pov=[HSP_View:BaseData],[Scenario:Forecast],
[Product:No Product],[Entity:Sales Mid-Atlantic];rows = [ Account ]; cols=
[Year, Period, Version]
```

> **Note:**
>
> El archivo identificado por `fileName` en el archivo CSV de entrada debe contener la definición de cuadrícula ad hoc en la hoja especificada. Por ejemplo, el archivo de entrada de ejemplo anterior especifica `sheet4` de `dropdown.xlsx` como origen de la definición de cuadrícula ad hoc. El archivo de Excel, junto con `requirement.csv` y el archivo CSV de entrada, debe estar disponible en el archivo *INPUT_FILE.zip* que se utiliza para ejecutar el comando simulateConcurrentUsage.

Archivo de entrada de ejecución de informe

Este archivo, al que se hace referencia en `requirement.csv` para soportar la apertura de informes de gestión, debe especificar el informe que se debe abrir. Cada línea del archivo debe tener el siguiente formato.

```
format=[REPORT_FORMAT];globalPov=[DIM 1:MEMBER 1],[DIM 2:MEMBER
2],..;prompts=[PROMPT 1:VALUE 1],[PROMPT 2:VALUE 2],..
```

`globalPov` y `prompts` son opcionales.

> **Note:**
>
> - Los formatos de informe soportados son `pdf` y `embedded`.
> - Si el nombre de la dimensión globalPov o sus miembros contienen dos puntos (:) o punto y coma (;), utilice el carácter de escape \\ delante. Por ejemplo, el nombre de dimensión `Version:View` se debe especificar como `Version\\:View`

Archivo de entrada de ejemplo para generar un .pdf a partir de un informe con el PDV global `[Version View:Working],[Sales Entity:International Sales]` y las peticiones de datos `[Actual;Budget],[Year:2018]`:

```
format=pdf;globalPov=[Version View:Working],[Sales Entity:International
Sales];prompts=[Actual:Budget],[Year:2018]
```

Archivo de entrada de ejecución de libro

Este archivo, al que se hace referencia en `requirement.csv` para soportar la apertura de libros en Informes, debe especificar el libro que se debe abrir. Cada línea del archivo debe tener el siguiente formato.

`format=BOOK_FORMAT` o

`format=BOOK_FORMAT;globalPov=[DIM 1:MEMBER 1],[DIM 2:MEMBER 2],..`

Archivo de entrada de ejemplo para generar un .pdf a partir de un libro con el PDV global *[Version View:Working],[Sales Entity:International Sales]*:

```
format=pdf;globalPov=[Version View:Working],[Sales Entity:International Sales]
```

> **Note:**
>
> - Los formatos de libro soportados son PDF y XLSX.
> - Si el nombre de la dimensión globalPov o sus miembros contienen dos puntos (:) o punto y coma (;), utilice el carácter de escape \\ delante. Por ejemplo, el nombre de dimensión `Version:View` se debe especificar como `Version\\:View`

Creación del archivo UserVarMemberMapping.csv

Este archivo es necesario si los formularios que ha especificado en el archivo de entrada de los casos de uso Abrir formulario o Guardar formulario requieren que se definan las variables de usuario. Este archivo no es necesario para otros casos de uso.

La primera línea de este archivo es la cabecera `#Dimension,User Variable,Member`

Las entradas siguientes contienen la asignación de dimensión, la variable de usuario y el miembro de dimensión.

Archivo `UserVarMemberMapping.csv` de ejemplo:

```
#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
```

Creación del archivo options.xml

Las opciones de Oracle Smart View for Office, como la supresión de filas, la supresión de bloques que faltan y la sangría de miembros de página, se pueden utilizar al simular los casos de uso Abrir formulario, Guardar formulario y Cuadrícula ad hoc. Cada una de estas opciones ejerce diferentes niveles de carga en el entorno.

Si desea simular la carga con las opciones de Smart View, puede incluir el archivo `options.xml` en el archivo ZIP de entrada.

Para crear `options.xml`, se exportan las opciones de Smart View. Para generar `options.xml`, en el separador **SmartView**, haga clic en **Opciones**, en **Aceptar** y, a continuación, en **Opciones de exportación**.

El uso de `options.xml` no es obligatorio. Si este archivo no está presente en el archivo ZIP de entrada, se utilizan las opciones predeterminadas para la simulación.

Creación del archivo users.csv

El modo 4 del comando simulateConcurrentUsage le permite ejecutar el comando con los usuarios existentes identificados en el archivo `users.csv` incluido en el archivo ZIP de entrada. En este modo no se crean usuarios para la simulación.

El formato del archivo `users.csv` es `loginname,password`:

```
jdoe,jdoe_pwd
john.doe@example.com,john_doe_pwd
```

El valor de `loginname` en `users.csv` debe coincidir con un nombre de usuario definido en el dominio de identidad. Además, el usuario identificado debe tener los roles predefinidos y de aplicación necesarios para ejecutar la operación.

Creación y carga del archivo ZIP de entrada en el entorno

Utilizando una herramienta como 7 Zip, cree un archivo ZIP que contenga `requirement.csv`, los archivos de entrada de caso de uso correspondientes y, opcionalmente, `UserVarMemberMapping.csv`, `users.csv` y `options.xml`, si es necesario.

Utilice el comando uploadFile para cargar el archivo ZIP resultante en la bandeja de entrada del entorno (ejemplo de sintaxis de comandos `epmautomate uploadFile "C:/uploads/INPUT_FILE.zip" inbox`) donde desee ejecutar la simulación.

Informe Simulación de uso simultáneo de ejemplo

Simulación de uso simultáneo se envía de forma predeterminada al usuario que ejecuta el comando simulateConcurrentUsage. Si especifica destinatarios de correo electrónico, el informe se envía por correo electrónico solo a esos destinatarios.

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

En este informe se identifica lo siguiente:

Columna	Descripción
N.º de operación	Número de secuencia del caso de uso en `requirement.csv`
Operación	Tipo de operación según se especifica en `requirement.csv`
Nombre de artefacto	Nombre de artefacto según se especifica en `requirement.csv`
Usuarios	Número de usuarios según se especifica en `requirement.csv`
Iteraciones	Número de veces que se ha ejecutado el caso de uso según se especifica en el parámetro `iterations`
Duración mínima	Tiempo mínimo que tarda un usuario en ejecutar este caso de uso
Duración máxima	Tiempo máximo que tarda un usuario en ejecutar este caso de uso
Duración media	Tiempo medio que tarda un usuario en ejecutar este caso de uso

Columna	Descripción
Estado de devolución	Estado del caso de uso. Se muestra `Failed` si la ejecución del caso de uso no es correcta

B

Preparación para ejecutar el comando replay

El comando replay se utiliza para probar el rendimiento de un entorno sometido a una carga, con objeto de verificar que la experiencia de usuario es aceptable cuando el servicio tiene una carga especificada. Debe completar algunos pasos antes de realizar pruebas de carga en entornos.

En este apéndice se describen los pasos que los administradores de servicio deben realizar antes de ejecutar el comando replay de EPM Automate.

- Acerca del comando replay
- Requisitos previos
- Creación de archivos HAR
- Creación de archivos de reproducción
- Generación de archivos de rastreo
- Sesión de reproducción de ejemplo

Acerca del comando replay

El comando replay reproduce la carga de Oracle Smart View for Office, Oracle Fusion Cloud Enterprise Performance Management, la API de REST de o la carga de EPM Automate en un entorno con el fin de activar las pruebas de rendimiento si hay una carga muy elevada para verificar que la experiencia de usuario es aceptable cuando el entorno tiene una carga especificada.

Por ejemplo, puede probar la experiencia de usuario en un entorno de prueba si hay una carga muy elevada para asegurarse de que actuará correctamente después de migrar la aplicación del entorno de prueba al entorno de producción.

Requisitos previos

Al ejecutar el comando con un archivo de reproducción, EPM Automate ejecuta cada fila del archivo de reproducción en paralelo para ejercer una carga en el servicio para que pueda realizar pruebas y verificar que la experiencia del usuario es aceptable cuando el servicio tiene una carga.

- Identifique los formularios que necesitan un mayor procesamiento en el entorno. Los formularios que manejan grandes cantidades de datos, o los formularios que incluyen cálculos complejos, son buenos candidatos. Por ejemplo, los formularios que se utilizan para enviar previsiones, procesos implicados en la creación de informes y ad hoc y estáticos, pueden ejercer grandes cargas en el servicio. Del mismo modo, actividades como la ejecución de reglas de negocio, la ejecución de informes, la ejecución de API de REST que ejecuten muchos recursos y comandos de EPM Automate (por ejemplo, la regla runBusiness, runDataRule, exportData, exportMetadata, restructureCube) podrían provocar que el entorno se someta a una carga muy elevada y pueden ser aptas para pruebas de carga.

- Instale Fiddler si es necesario. En EPM Automate se necesita un archivo de formato HTTP Archive (HAR) 1.1 que contenga registros de Oracle Smart View for Office, la API de REST de Oracle Fusion Cloud Enterprise Performance Management, o bien la interacción de EPM Automate con el entorno de Cloud EPM. Normalmente, se usa Fiddler para generar el archivo HAR que captura el registro de la interacción con Cloud EPM.
- Ejecute las actividades principales que haya identificado anteriormente. Utilice Smart View para ejecutar las actividades, como la apertura y el almacenamiento de formularios, la ejecución de reglas de negocio y la creación de informes, y Fiddler para capturar los detalles de las actividades y exportarlos a archivos HAR. De la misma forma, ejecute las API de REST y los comandos de EPM Automate y haga que Fiddler capture los detalles. Consulte Creación de archivos HAR para obtener información.
- Cree un archivo CSV de reproducción que muestre las credenciales (nombres de usuario y contraseñas) y el nombre de los archivos HAR que se van a ejecutar. Cada fila del archivo puede contener el nombre de usuario y la contraseña de un usuario único para simular varias sesiones de usuario simultáneas. Consulte Creación de archivos de reproducción para obtener información
  El usuario cuyas credenciales se hayan especificado en una fila para ejecutar un archivo HAR no tiene que ser el usuario que haya ejecutado la sesión que se haya utilizado para crear el archivo HAR. Sin embargo, este usuario debe tener los derechos para ejecutar estas actividades en el entorno.

Consulte Sesión de reproducción de ejemplo para obtener los pasos detallados para ejecutar el comando de reproducción.

Creación de archivos HAR

El archivo HAR captura los rastreos de la interacción de Oracle Smart View for Office, la API de REST o de EPM Automate con Oracle Fusion Cloud Enterprise Performance Management.

Debido a que Fiddler captura información sobre todo el tráfico de HTTP(S), al crear los archivos HAR, absténgase de realizar actividades que puedan agregar rastreos innecesarios a Fiddler.

Para crear un archivo HAR:

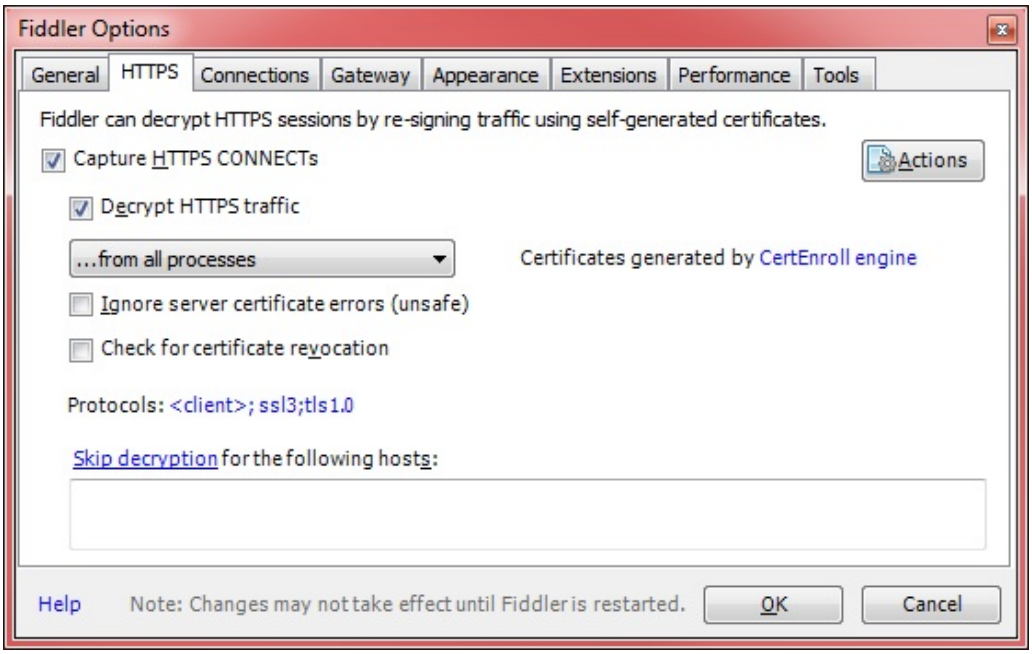
1. Inicie Fiddler.
2. Asegúrese de que Fiddler está configurado para descifrar el tráfico de HTTPS de todos los procesos.
   a. Seleccione **Herramientas**, a continuación **Opciones** y, por último, **HTTPS**.
   b. Seleccione **Descifrar tráfico HTTPS**, si no está seleccionada.

      Fiddler muestra información sobre el certificado raíz que utiliza para interceptar el tráfico de HTTPS. En general, es seguro confiar en este certificado.

c. Haga clic en **Sí** si desea agregar el certificado raíz a la lista de CA de confianza; si no, seleccione **No**.

d. **Opcional:** si ha seleccionado **No** en el paso anterior, puede seleccionar **Omitir errores de certificado de servidor** para suprimir las advertencias de seguridad de Fiddler relacionadas con el descifrado del tráfico de HTTPS.

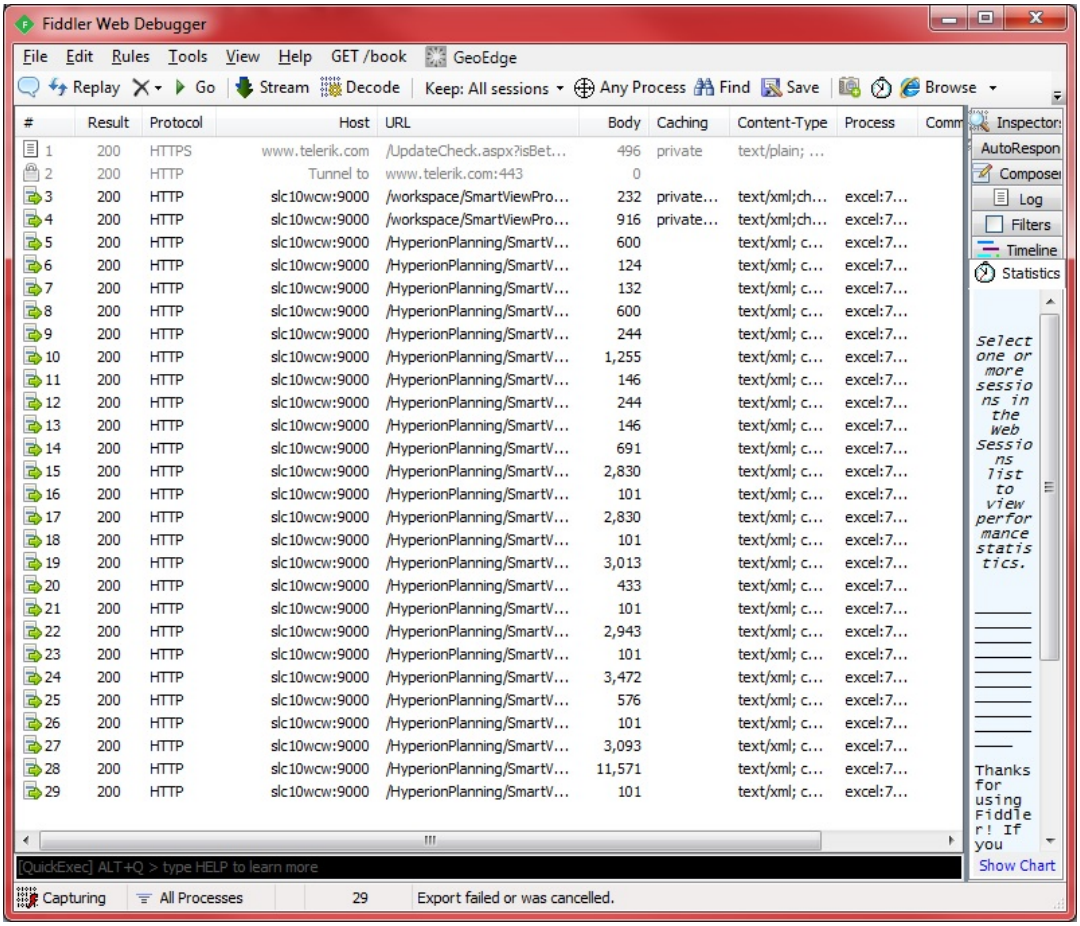
e. Haga clic en **Aceptar**.



3. Inicie Smart View y acceda al entorno para el que desea capturar el rastreo.

4. Con Smart View, la API de REST o EPM Automate, ejecute las actividades que ejercen una gran carga de procesamiento en el entorno. Por ejemplo, abra los formularios en Smart View para que Fiddler pueda registrar su actividad.

   Fiddler registra los procesos de que ha iniciado.

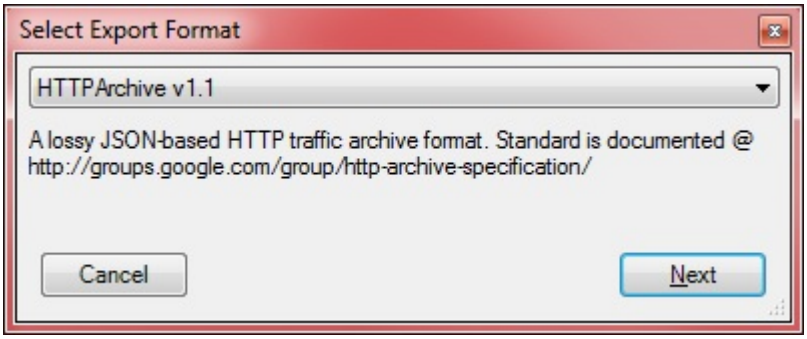


5. En Fiddler, realice estos pasos:

   a. Seleccione `Archivo`, luego `Exportar sesiones` y, a continuación, `Todas las sesiones` o `Sesiones seleccionadas`. Si se ha conectado a otros sitios web al ejecutar Fiddler, seleccione `Sesiones seleccionadas` para elegir las sesiones relevantes para el entorno.

   b. En `Seleccionar formato de exportación`, seleccione `HTTPArchive v1.1` como formato de exportación.

   c. Haga clic en `Siguiente`.



   d. En `Exportar como HTTPArchive v1.1`, seleccione el directorio donde desea almacenar el archivo y especifique un nombre de archivo.

e. Haga clic en `Guardar`.

Creación de archivos de reproducción

Un archivo de reproducción es un archivo CSV que muestra las credenciales (nombre de usuario y contraseña) y el nombre de los archivos HAR que se van a ejecutar para cargar el sistema mediante el comando replay de EPM Automate.

Asegúrese de que el nombre de usuario y la contraseña que especifique tienen los derechos necesarios para ejecutar las actividades incluidas en el archivo HAR.

Al ejecutar el comando de reproducción, EPM Automate ejecuta cada fila en el archivo de reproducción en paralelo para ejercer una carga en el servicio. Por ejemplo, si el archivo de reproducción contiene 10 filas, EPM Automate reproduce 10 sesiones para que pueda realizar pruebas y verificar que la experiencia del usuario es aceptable cuando el servicio tiene una carga especificada. Cada actividad incluida en el archivo HAR se ejecuta en serie.

Consulte replay para obtener más información sobre la ejecución del comando de reproducción.

Para crear un archivo de reproducción:

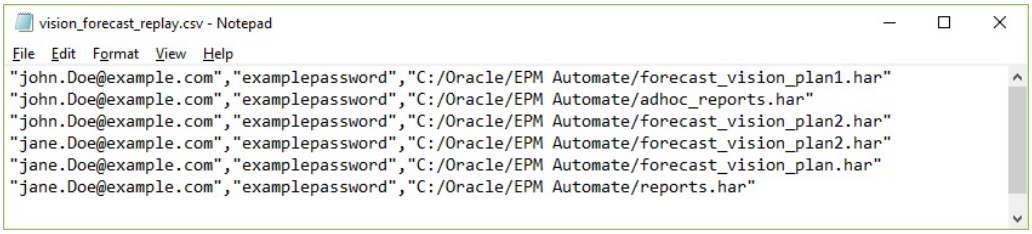
1. Abra Microsoft Office Excel e inicie una nueva hoja de trabajo.
2. Introduzca un nombre de usuario, la contraseña y la ubicación de un archivo HAR en las columnas A, B y C respectivamente de la fila 1.

   Repita este paso para crear filas adicionales.

   > **Nota:**
   >
   > Debe especificar la ruta absoluta a la ubicación del archivo HAR. Utilice la barra (/) como separador de directorio en rutas de archivo. No utilice barras diagonales inversas (\).

3. Guarde el archivo.
4. En `Guardar como`, realice estos pasos:
   a. Seleccione el directorio donde desea almacenar el archivo de reproducción.
   b. En `Nombre de archivo`, especifique un nombre y, en `Guardar como tipo`, seleccione `CSV (separado por comas) (*.csv)`.
   c. Haga clic en `Guardar`.

      Un archivo de reproducción de ejemplo puede ser el siguiente:



Generación de archivos de rastreo

Al ejecutar el comando replay, puede generar archivos de rastreo para compartir con los Servicios de Soporte Oracle para solucionar los problemas. Los Servicios de Soporte Oracle utilizan los archivos de rastreo para comprender cómo el entorno ha manejado una actividad de Oracle Smart View for Office.

Utilice el parámetro `trace=true` opcional con el comando replay para generar los archivos de rastreo en formato XML. Si utiliza este parámetro, por cada actividad en el archivo HAR, EPM Automate crea un archivo de rastreo que contiene la respuesta de Smart View a la actividad.

Los archivos de rastreo se denominan `trace-N.xml`; por ejemplo, `trace-1.xml`, donde `N` es un contador que comienza en 1. Si se especifican varios archivos HAR con nombres idénticos en el archivo de reproducción, EPM Automate consolida los archivos de rastreo en una carpeta.

Los archivos de rastreo relacionados con un archivo HAR están almacenados en una carpeta en el directorio desde el que ejecuta EPM Automate. EPM Automate crea una carpeta para cada archivo HAR que aparece en el archivo de reproducción. EPM Automate usa una combinación de hora del sistema del servidor actual y nombre del archivo HAR con formato `AAAA_MM_DD_HH_MM_SS_NOMBRE_ARCHIVO_HAR` para asignar un nombre a las carpetas. Por ejemplo, si el nombre del archivo HAR es `forecast1.har`, el nombre de la carpeta puede ser `2016_06_08_10_21_42_forecast1`.

Sesión de reproducción de ejemplo

Describe cómo ejecutar el comando replay utilizando varios archivos HAR.

En esta sección se da por hecho lo siguiente:

- Ha creado los siguientes archivos HAR. Cada archivo HAR puede contener el mismo conjunto de actividades. Consulte Creación de archivos HAR para obtener información detallada.
  - `C:\Oracle\EPM Automate\forecast_vision_plan1.har`
  - `C:\Oracle\EPM Automate\forecast_vision_plan2.har`
  - `C:\Oracle\EPM Automate\forecast_plan2.har`
- Ha creado un archivo de reproducción `C:/Oracle/EPM Automate/vision_forecast_replay.csv` con el siguiente contenido (consulte Creación de archivos de reproducción para obtener más información):

> **Nota:**
>
> Utilice la barra (/) como separador de directorio en rutas de archivo del archivo de reproducción. No utilice barras diagonales inversas (\).

```
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
```

Para ejecutar el comando replay:

1. En la ventana del símbolo del sistema, navegue al directorio; por ejemplo, `C:\Oracle\EPM Automate\bin`, donde está instalado EPM Automate.
2. Conéctese a un entorno como administrador del servicio y, a continuación, ejecute el comando replay:

   ```
   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 muestra información de reproducción en la consola y termina el procesamiento después de la duración especificada (12 minutos en el ejemplo anterior). También crea los archivos y carpetas de rastreo porque el comando anterior incluye el parámetro `trace=true`.
   Debido a que el comando se ha ejecutado desde `C:\Oracle\EPM Automate\bin`, EPM Automate ha almacenado los archivos de rastreo en las siguientes carpetas. Tenga en cuenta que estas carpetas se denominan según los nombres de los archivos 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. Desconéctese del entorno:

   ```
   epmautomate logout
   ```

C

Manejo de caracteres especiales

Los valores de contraseñas, contraseñas de proxy y parámetros de comando de Oracle Fusion Cloud Enterprise Performance Management pueden contener caracteres especiales. Es necesario aplicar un tratamiento especial a EPM Automate para que admita estos caracteres.

En los ejemplos de esta sección se usa la contraseña de ejemplo para demostrar el uso de los caracteres especiales.

Oracle recomienda que incluya los pares de valores y parámetros entre comillas dobles.

Windows

Estos caracteres especiales se deben identificar con comillas dobles (") junto al carácter especial o junto al valor de parámetro que contiene el carácter especial.

> **Nota:**
>
> EPM Automate no se puede ejecutar desde una carpeta que contiene `&` en su nombre; por ejemplo, `C:\Oracle\A&B`.

Tabla C-1 Manejo de caracteres especiales: Windows

Carácter	Descripción	Ejemplo con carácter de escape
)	Paréntesis de cierre	• `Example")"pwd1` or • `"Example)pwd1"`
<	Menor que	• `Example"<"pwd1` o • `"Example<pwd1"`
>	Mayor que	• `Example">"pwd1` o • `"Example>pwd1"`
&	Ampersand	• `Example"&"pwd1` or • `"Example&pwd1"`
|	Pleca	• `Example"|"pwd1` or • `"Example|pwd1"`
"	Comillas	• `Example""pwd1` o • `"Example"pwd1"`

Uso del signo de exclamación en contraseñas de texto sin formato en archivos por lotes de Windows

El uso del signo de exclamación (!) en contraseñas de texto sin formato en archivos por lotes de Windows utilizados con EPM Automate se debe gestionar como se indica a continuación:

1. Utilice dos símbolos de intercalación (^^) delante del signo de exclamación como carácter de escape. Por ejemplo, si la contraseña es `Welc0me!`, codifíquela como `Welc0me^^!`

2. Actualice el archivo por lotes para definir DisableDelayedExpansion al principio del archivo mediante la inclusión de la siguiente declaración:
   `setlocal DisableDelayedExpansion`
3. Elimine la declaración `setlocal EnableExtensions EnableDelayedExpansion`, si está presente, del script.

UNIX/Linux

En los sistemas operativos UNIX y Linux, en los caracteres especiales se debe usar una barra invertida (\).

> **Nota:**
>
> - Para salir `!` (signo de exclamación), utilice una sola comilla alrededor de la contraseña o use la barra invertida (\) como carácter de escape.
> - Para identificar `\`, `$`, `'` y `"`, utilice comillas dobles alrededor de la contraseña o use la barra invertida (\) como carácter de escape.

Tabla C-2 Manejo de caracteres especiales: UNIX/Linux

Carácter	Descripción	Ejemplo con carácter de escape
(	Paréntesis de apertura	`Example\(pwd1`
)	Paréntesis de cierre	`Example\)pwd1`
<	Menor que	`Example\<pwd1`
>	Mayor que	`Example\>pwd1`
`	Apóstrofe	`Example\'pwd1`
!	signo de exclamación	• `'Example!pwd1'` or • `Example\!pwd1`
#	Hash	`Example\#pwd1`
&	Ampersand	`Example\&pwd1`
|	Pleca	`Example\\|pwd1`
;	Punto y coma	`Example\;pwd1`
.	Punto	`Example\.pwd1`
"	Comillas	• `Example\"pwd1` o • `"Example\"pwd1"`
'	Comillas simples	• `Example\'pwd1` o • `"Example\'pwd1"`
$	Signo de dólar	• `Example\$pwd1` o • `"Example\$pwd1"`
\	Barra diagonal inversa	• `Example\\pwd1` o • `"Example\\pwd1"`

Uso del signo de exclamación en contraseñas de texto sin formato en scripts de UNIX o Linux

En scripts de UNIX/Linux, si una contraseña de EPM Automate almacenada en una variable de shell contiene caracteres especiales, use tres barras invertidas como secuencia de escape y, a continuación, incluya la cadena entre comillas dobles. Por ejemplo, la contraseña `lzi[ACO(e*7Qd)jE` incluida en la variable del shell `password` se debe incluir en el script de la siguiente forma:

```
password="lzi[ACO\\\(e*7Qd\\\)jE"
```

D

Comandos específicos de cada servicio de Cloud EPM

- Comandos de Account Reconciliation
- Comandos de Financial Consolidation and Close
- Comandos de Narrative Reporting
- Comandos de Oracle Enterprise Data Management Cloud
- Comandos de Planning, Planning Modules, FreeForm, Strategic Workforce Planning y Sales Planning Commands
- Comandos de Profitability and Cost Management
- Comandos de Enterprise Profitability and Cost Management
- Comandos de Tax Reporting

Comandos de Account Reconciliation

Comandos de EPM Automate para Account Reconciliation		
addUsers	groupAssignmentAuditReport	renameSnapshot
addUsersToGroup	help	replay
addUsersToTeam	importARApplicationProperties	resetService
addUserToGroups	importBackgroundImage	restoreBackup
archiveTmTransactions	importLogoImage	roleAssignmentAuditReport
assignRole	importBalances	roleAssignmentReport
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	sendMail
deleteFile	importTMAttributeValues	setApplicationAdminMode
deleteGroups	importTmPremappedTransactions	setDailyMaintenanceStartTime
downloadFile	invalidLoginReport	setDemoDates
encrypt	listBackups	setEncryptionKey
exportAccessControl	listFiles	setIdleSessionTimeout
exportARApplicationProperties	login	setIPAllowlist
exportBackgroundImage	logout	setRestrictedDataAccess
exportDataManagement	provisionReport	setManualDataAccess
exportLogoImage	purgeArchivedTmTransactions	setPeriodStatus
exportMapping	purgeTmTransactions	setVirusScanOnFileUploads
exportSnapshot	recreate	skipUpdate
feedback	refreshCube	unassignRole
getApplicationAdminMode	removeUserFromGroups	updateUsers
getDailyMaintenanceStartTime	removeUsers	upgrade
getIdleSessionTimeout	removeUsersFromGroup	uploadFile
getIPAllowlist	removeUsersFromTeam	userAuditReport
getRestrictedDataAccess		userGroupReport
getVirusScanOnFileUploads		

Comandos de Financial Consolidation and Close

Comandos de EPM Automate para 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
getRestrictedDataAccess
getSubstVar
getVirusScanOnFileUploads
groupAssignmentAuditReport
help
importAppSecurity
importConsolidationJournals
importData
importDataManagement
importJobConsole
importMapping
importMetadata
importOwnershipData
importSnapshot
importSupplementalCollectionData
importSupplementalData
importValidIntersections
invalidLoginReport
listBackups
listFiles
login
logout
maskData
provisionReport
recomputeOwnershipData
recreate
refreshCube
removeUserFromGroups
removeUsers
removeUsersFromGroup
removeUsersFromTeam
renameSnapshot
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
setRestrictedDataAccess
setVirusScanOnFileUploads
setManualDataAccess
setSubstVars
simulateConcurrentUsage
skipUpdate
snapshotCompareReport
unassignRole
updateGuidedLearningSettings
updateUsers
upgrade
uploadFile
userAuditReport
userGroupReport

Comandos de EPM Automate para Financial Consolidation and Close	
exportSnapshot	validateConsolidationMetadata

Comandos de Narrative Reporting

Comandos de EPM Automate para Narrative Reporting		
addUsers	getIdleSessionTimeout	restoreBackup
addUsersToGroup	getIPAllowlist	roleAssignmentAuditReport
addUserToGroups	getRestrictedDataAccess	roleAssignmentReport
assignRole	getVirusScanOnFileUploads	runDailyMaintenance
cloneEnvironment	groupAssignmentAuditReport	sendMail
copyFileFromInstance	help	setDailyMaintenanceStartTime
copyFromObjectStorage	importLibraryArtifact	setEncryptionKey
copyFromSFTP	invalidLoginReport	setIdleSessionTimeout
copyToObjectStorage	listBackups	setIPAllowlist
copyToSFTP	listFiles	setManualDataAccess
createGroups	login	setRestrictedDataAccess
createNRSnapshot	logout	setVirusScanOnFileUploads
deleteFile	provisionReport	skipUpdate
deleteGroups	recreate	unassignRole
downloadFile	removeUserFromGroups	updateUsers
encrypt	removeUsers	upgrade
executeBurstDefinition	removeUsersFromGroup	uploadFile
exportLibraryArtifact	replay	userAuditReport
feedback	resetService	userGroupReport
getDailyMaintenanceStartTime		

Comandos de Oracle Enterprise Data Management Cloud

Comandos de EPM Automate para Oracle Fusion Cloud Enterprise Data Management		
addUsers	getIdleSessionTimeout	replay
addUsersToGroup	getIPAllowlist	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
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
extractPackage	renameSnapshot	userGroupReport
feedback		
getDailyMaintenanceStartTime		

Comandos de Planning, Planning Modules, FreeForm, Strategic Workforce Planning y Sales Planning

Comandos de EPM Automate para Planning, Planning Modules, FreeForm, Strategic Workforce Planning y Sales Planning

addUsers
addUsersToGroup
addUserToGroups
applicationAdminMode
assignRole
autoPredict * Véase la nota al pie
clearCube
cloneEnvironment
compactCube
copyFileFromInstance
copyFromObjectStorage
copyFromSFTP
copySnapshotFromInstance
copyToObjectStorage
copyToSFTP
createGroups
deleteFile
deleteGroups
dismissIPMInsights**
downloadFile
enableQueryTracking
encrypt
essbaseBlockAnalysisReport
executeAggregationProcess
executeReportBurstingDefinition
exportAppAudit
exportAppSecurity
exportCellLevelSecurity
exportData
exportDataManagement
exportEssbaseData
exportJobConsole
exportLibraryDocument
exportMapping
exportMetadata
exportSnapshot
exportValidIntersections
feedback
getApplicationAdminMode
getDailyMaintenanceStartTime
getEssbaseQryGovExecTime
getIdleSessionTimeout
getIPAllowlist
getRestrictedDataAccess
getSubstVar
getVirusScanOnFileUploads
groupAssignmentAuditReport
help
importAppAudit
importAppSecurity
importCellLevelSecurity
importData
importDataManagement
importJobConsole
importMapping
importMetadata
importSnapshot
importValidIntersections
invalidLoginReport
listBackups
listFiles
login
logout
maskData
mergeDataSlices
provisionReport
recreate
refreshCube
removeUserFromGroups
removeUsers
removeUsersFromGroup
renameSnapshot
replay
resetService
restoreBackup
restructureCube
roleAssignmentAuditReport
roleAssignmentReport
runBatch
runBusinessRule
runDailyMaintenance
runDataRule
runDMReport
runIntegration
runPipeline
runPlanTypeMap
runRuleSet
sendMail
setApplicationAdminMode
setDailyMaintenanceStartTime
setDemoDates
setEncryptionKey
setEssbaseQryGovExecTime
setIdleSessionTimeout
setIPAllowlist
setManualDataAccess
setRestrictedDataAccess
setSubstVars
setVirusScanOnFileUploads
simulateConcurrentUsage
skipUpdate
snapshotCompareReport
sortMember
unassignRole
updateGuidedLearningSettings
updateUsers
upgrade
uploadFile
userAuditReport
userGroupReport

Comandos de EPM Automate para Planning, Planning Modules, FreeForm, Strategic Workforce Planning y Sales Planning
* Este comando no está soportado para FreeForm, Strategic Workforce Planning y Sales Planning. ** Este comando no está soportado para Formato libre.

Comandos de Profitability and Cost Management

Comandos de EPM Automate para Profitability and Cost Management		
addUsers	getEssbaseQryGovExecTime	renameSnapshot
addUsersToGroup	getIdleSessionTimeout	replay
addUserToGroups	getIPAllowlist	resetService
applyDataGrants	getRestrictedDataAccess	restoreBackup
assignRole	getVirusScanOnFileUploads	roleAssignmentAuditReport
clearPOV	groupAssignmentAuditReport	roleAssignmentReport
cloneEnvironment	help	runBatch
copyFileFromInstance	importDataManagement	runCalc
copyFromObjectStorage	importMapping	runDailyMaintenance
copyFromSFTP	importSnapshot	runDataRule
copyPOV	importTemplate	runDMReport
copySnapshotFromInstance	invalidLoginReport	runIntegration
copyToObjectStorage	listBackups	sendMail
copyToSFTP	listFiles	setDailyMaintenanceStartTime
createGroups	loadData	setEncryptionKey
deleteFile	loadDimData	setEssbaseQryGovExecTime
deleteGroups	login	setIdleSessionTimeout
deletePOV	logout	setIPAllowlist
deployCube	mergeSlices	setRestrictedDataAccess
downloadFile	optimizeASOCube	setManualDataAccess
enableApp	programDocumentationReport	skipUpdate
encrypt	provisionReport	setVirusScanOnFileUploads
exportDataManagement	recreate	unassignRole
exportMapping	removeUserFromGroups	updateUsers
exportQueryResults	removeUsers	upgrade
exportSnapshot	removeUsersFromGroup	uploadFile
exportTemplate		userAuditReport
feedback		userGroupReport
getDailyMaintenanceStartTime		

Comandos de Enterprise Profitability and Cost Management

Comandos de EPM Automate para Enterprise Profitability and Cost Management		
addUsers	exportTaskManagerAccessControl	removeUsers
addUsersToGroup	exportValidIntersections	removeUsersFromGroup
addUserToGroups	feedback	renameSnapshot
applicationAdminMode	getApplicationAdminMode	replay
assignRole	getDailyMaintenanceStartTime	resetService
calculateModel	getEssbaseQryGovExecTime	restoreBackup
clearCube	getIdleSessionTimeout	roleAssignmentAuditReport
compactCube	getIPAllowlist	roleAssignmentReport
copyDataByPointOfView	getRestrictedDataAccess	runBatch
cloneEnvironment	getSubstVar	runBusinessRule
copyFileFromInstance	getVirusScanOnFileUploads	runDailyMaintenance
copyFromObjectStorage	groupAssignmentAuditReport	runDataRule
copyFromSFTP	help	runDMReport
clearDataByPointOfView	importAppAudit	runIntegration
copySnapshotFromInstance	importAppSecurity	runPipeline
copyToObjectStorage	importCellLevelSecurity	sendMail
copyToSFTP	importData	runRuleSet
createGroups	importDataManagement	runTaskManagerReport
deleteFile	importJobConsole	setApplicationAdminMode
deleteGroups	importMapping	setDailyMaintenanceStartTime
downloadFile	importMetadata	setDemoDates
deletePointOfView	importSnapshot	setEncryptionKey
deployTaskManagerTemplate	importValidIntersections	setEssbaseQryGovExecTime
enableQueryTracking	invalidLoginReport	setIdleSessionTimeout
encrypt	listBackups	setIPAllowlist
executeAggregationProcess	listFiles	setManualDataAccess
executeReportBurstingDefinition	login	setRestrictedDataAccess
exportAppAudit	logout	setSubstVars
exportAppSecurity	maskData	setVirusScanOnFileUploads
exportCellLevelSecurity	mergeDataSlices	skipUpdate
exportData	provisionReport	snapshotCompareReport
exportDataManagement	recreate	sortMember
exportEssbaseData	refreshCube	unassignRole
exportJobConsole	removeUserFromGroups	updateGuidedLearningSettings
exportLibraryDocument		updateUsers
exportMapping		upgrade
exportMetadata		uploadFile
exportMetadata		userAuditReport
exportSnapshot		userGroupReport
		validateModel

Comandos de Tax Reporting

Comandos de EPM Automate para Tax Reporting

addUsers
addUsersToGroup
addUsersToTeam
addUserToGroups
applicationAdminMode
assignRole
clearDataByProfile
copyDataByProfile
copyFileFromInstance
copyFromObjectStorage
copyFromSFTP
copyOwnershipDataToNextYear
copySnapshotFromInstance
copyToObjectStorage
copyToSFTP
createGroups
deleteFile
deleteGroups
deployFormTemplates
deployTaskManagerTemplate
downloadFile
encrypt
essbaseBlockAnalysisReport
executeReportBurstingDefinition
exportAppAudit
exportCellLevelSecurity
exportData
exportDataManagement
exportEssbaseData
exportJobConsole
exportLibraryDocument
exportMapping
exportMetadata
exportOwnershipData
exportSnapshot
exportTaskManagerAccessControl
exportValidIntersections
feedback
getApplicationAdminMode
getDailyMaintenanceStartTime
getEssbaseQryGovExecTime
getIdleSessionTimeout
getIPAllowlist
getRestrictedDataAccess
getSubstVar
getVirusScanOnFileUploads
groupAssignmentAuditReport
help
importAppSecurity
importCellLevelSecurity
importData
importDataManagement
importJobConsole
importMapping
importMetadata
importOwnershipData
importSnapshot
importSupplementalCollectionData
importSupplementalData
importValidIntersections
invalidLoginReport
listBackups
listFiles
login
logout
maskData
provisionReport
recomputeOwnershipData
recreate
refreshCube
removeUserFromGroups
removeUsers
removeUsersFromGroup
removeUsersFromTeam
renameSnapshot
replay
resetService
restoreBackup
restructureCube
roleAssignmentAuditReport
roleAssignmentReport
runBatch
runBusinessRule
runDailyMaintenance
runDataRule
runDMReport
runIntegration
runPipeline
runRuleSet
runSupplementalDataReport
runTaskManagerReport
sendMail
setApplicationAdminMode
setDailyMaintenanceStartTime
setDemoDates
setEncryptionKey
setEssbaseQryGovExecTime
setIdleSessionTimeout
setIPAllowlist
setManualDataAccess
setRestrictedDataAccess
setSubstVars
setVirusScanOnFileUploads
simulateConcurrentUsage
skipUpdate
snapshotCompareReport
unassignRole
updateGuidedLearningSettings
upgrade
updateUsers
uploadFile
userAuditReport
userGroupReport