Anaplan

Anaplan Connect Guide

Version 1.4

Table of Contents

What's New in Anaplan Connect 1.4

Welcome to Anaplan Connect 1.4. The following are changes in features and behavior in this release:

New and Updated Features

New and updated features in Anaplan Connect 1.4 include:

- Use of the new and improved Anaplan v2.0 integration API.
- Enabling customers to use certificates, obtained from public Certificate Authorities, for Anaplan authentication. For more information, see Certificate Authentication.
- Enabling users to configure retry timeout and number of retries. For more information, see Appendix C: List of all Operator Commands.
- Configurable chunk sizes for imports, ranging from 1 to 50 MB. For more information, see Appendix C: List of all Operator Commands.
- Improved screen logging, including timestamp, classpath for debug lines, Linux process ID, and more. For more information, see Log Files in Anaplan Connect.
- Better security and ease of querying using JDBCparameters moved to ”properties” file.

Changes to Behavior

With Anaplan Connect 1.4, users can run API calls against their default Tenant only. This change provides enhanced security.

Workaround:If you have to access Anaplan models across multiple tenants and want to run Anaplan Connect 1.4 integrations in these tenants, create different user IDs for different Tenants and use these user IDs to run your integrations.

Introduction

Anaplan Connect is an API Client with a command-line interface that supports the following types of Anaplan actions:

- import
- export
- delete - **Delete from List using Selection** to remove specific items from a list.
- process - a combination of the other types of actions
- upload

The advantages include:

- No need for manual work in the Anaplan GUI for each run. You can schedule jobs to run automatically at the interval you want
- No need to code a full-scale software application
- Integration with Anaplan's version 2 API for enhanced security.

This document shows you how to write a file of commands for your operating system.

Windows	Linux
Write the batch (.bat) file using TextPad, SublimeText, Notepad, or similar	TextPad SublimeText Write the script (.sh) file using nano, vi, SublimeText, or similar
Use a command prompt to run a batch file, such as myImport.bat	Use Terminal to run a shell script, such as myImport.sh

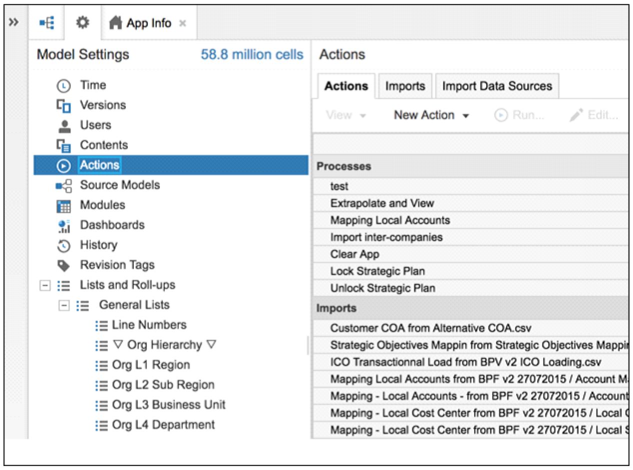
Example: A batch file named RunMyImport.bat that loads a text file, Europe.txt, onto the Anaplan server. If you also use a scheduling tool, the batch file can run itself at any hour and interval you choose (daily, weekly) without you having to log onto Anaplan or be present.

Anaplan Connect compresses the files during upload. Do not refer to zip files in the .bat or .sh file, this is not supported.

Prerequisites

Access to Anaplan model with the actions configured

Open an Anaplan model that already has the actions (Import, Export, Delete, or Process) that you want Anaplan Connect to run.



Processes

- test
- Extrapolate and View
- Mapping Local Accounts
- Import inter-companies
- Clear App
- Lock Strategic Plan
- Unlock Strategic Plan

Imports

- Customer COA from Alternative COA.csv
- Strategic Objectives Mappin from Strategic Objectives Mappi
- ICO Transactionnal Load from BPV v2 ICO Loading.csv
- Mapping Local Accounts from BPF v2 27072015 / Account M
- Mapping - Local Accounts - from BPF v2 27072015 / Account
- Mapping - Local Cost Center from BPF v2 27072015 / Local C
- Mapping - Local Cost Center from BPF v2 27072015 / Local S

If you do not have access to Anaplan, work with someone who has the ability to create actions in Anaplan.

Java version and third-party data sources

- An installation of Java 7 or 8. If you want to use Anaplan Connect to import from an ODBC data source, note that Java 8 does not support the JDBC-ODBC Bridge (see https://blogs.oracle.com/Lance/entry/removal_of_the_jdbc_odbc).
- Anaplan Connect supports Java Database Connectivity (JDBC), which means it is possible to work with many third-party data sources.
  - JDBC for Oracle, Access, Excel
  - Import through a JDBC Connection for a Microsoft SQL Server database

Firewall and Proxy Access

Whitelist api.anaplan.com in your firewall and any proxies your organization uses.

If you’re using an SSO-enabled workspace

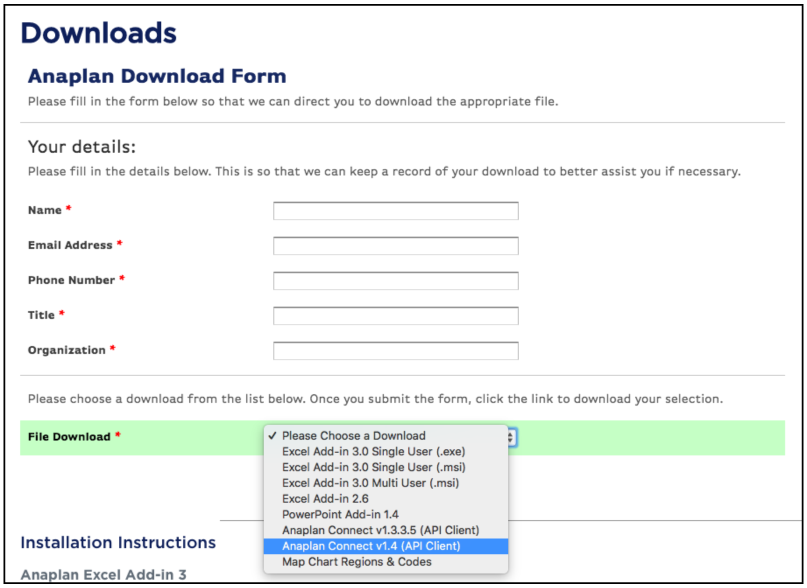
If the actions you want Anaplan Connect to run are for models in a workspace using single sign-on, we recommend using Certificate Authority (CA) Authentication.

If you use Anaplan's Basic authentication, the Anaplan Connect Single Sign-on (SSO) user must be an **Exception User**. An Exception User can authenticate by username and password or by certificate, rather than through SAML.

See: https://help.anaplan.com/anapedia/Content/Administration_and_Security/Security/Single_Sign-on.html.

Download and Setup

1. Go to the Downloads page in Anapedia.
2. Fill in the information on the Anaplan Download Form.
3. On the **File Download** list, select **Anaplan Connect v1.4 (API Client)**, and click **Submit**.



Downloads

Anaplan Download Form

Please fill in the form below so that we can direct you to download the appropriate file.

Your details:

Please fill in the details below. This is so that we can keep a record of your download to better assist you if necessary.

Name *

Email Address *

Phone Number *

Title *

Organization *

Please choose a download from the list below. Once you submit the form, click the link to download your selection.

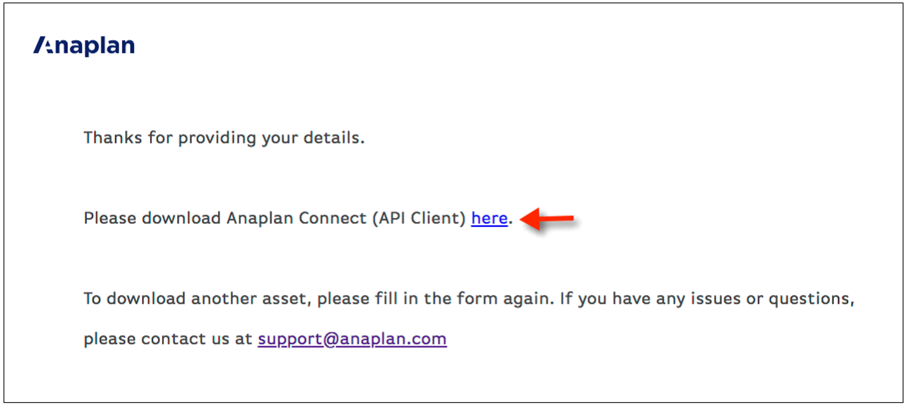
File Download *

✓ Please Choose a Download
Excel Add-in 3.0 Single User (.exe)
Excel Add-in 3.0 Single User (.msi)
Excel Add-in 3.0 Multi User (.msi)
Excel Add-in 2.6
PowerPoint Add-in 1.4
Anaplan Connect v1.3.3.5 (API Client)
Anaplan Connect v1.4 (API Client)
Map Chart Regions & Codes

Installation Instructions

Anaplan Excel Add-in 3

4. Click the **here** link and save the zip file to your hard drive.



Anaplan

Thanks for providing your details.

Please download Anaplan Connect (API Client) here.

To download another asset, please fill in the form again. If you have any issues or questions, please contact us at support@anaplan.com

5. Extract the zip to a directory.

⚠ **Note:** If the folder name contains parentheses, an error might occur when using Anaplan Connect. Do not install Anaplan Connect to a folder that has parentheses in its name. For example, Program Files (x86).

Work with Batch (.bat) Files

In your Anaplan Connect installation folder, at *examples\example.bat*, is the following code:

```
@echo off
rem This example loads a source text file and runs an Anaplan import into
a module.
rem For details of how to configure this script see doc\Anaplan Connect
User Guide.doc
set AnaplanUser="fred.smith@mycompany.com:password"
set WorkspaceId="My Workspace"
set ModelId="My Model"
set Operation=-file "My Source.txt" -put "C:\My Source.txt" -import "My Module
from My Source.txt" -execute
  -output "C:\My Errors.txt"
rem *** End of settings - Do not edit below this line ***
setlocal enableextensions enabledelayedexpansion || exit /b 1
cd %~dp0
if not %AnaplanUser% == "" set Credentials=-user %AnaplanUser%
  set Command=.\AnaplanClient.bat %Credentials% -workspace %WorkspaceId% -
model %ModelId% %Operation%
@echo %Command%
cmd /c %Command%
pause
```

where:

fred.smith@mycompany.com:password	your Anaplan login credentials
My Workspace	your workspace ID or workspace name
My Model	your model ID
My Source.txt	a flat file in your local host
C:\My Source.txt	full path to the flat file in your local host

C:\My Errors.txt	full path to where you want Anaplan Connect to create a dump file in your local host. The dump file contains the records that could not be imported.

Copy the example.bat file up one directory, to the root of your Anaplan Connection installation, and rename it *myFirstTest.bat*. This location guarantees that Anaplan Connect can use any batch file you create.

Configure the Batch File

Every batch file contains the following information:

- User Name and Password - unless you are using Certificate-based Authentication.
- Workspace ID - Unique and will not change
- Model ID - Unique to each model and will not change
- Action from Anaplan, such as a specific Import or Export

When running a batch file or script at an interactive terminal, the user is prompted for the password if it is not included in the script.

Integration jobs can take longer than the 30 minutes allocated to an Auth token, Anaplan Connect allows calls to be made to the API after this expiry limit without re-entering the authentication details.

⚠ **Warning**: A script that runs without user interaction must contain the credentials, so take measures to secure the file, machine, and account.

Locate the Workspace ID and Model ID

1. Log in to Anaplan.
2. Open the model you want to use.
3. In the upper-right corner, click **Help > About**. The **About** dialog displays the values of the workspace and model IDs.
4. Copy the workspace ID to the line of your batch file that begins with *set WorkspaceId=''*.
5. Copy the model ID to the line of your batch file that begins with *set ModelId=''*.

Locate the Actions for Anaplan Connect to Perform

1. Open the Anaplan model that has the actions you want Anaplan Connect to perform.
2. In **Model Settings > Actions**, note the exact names of the actions, including capitalization and file extensions.

Certificate Authentication using KeyStore

Certificate Authority (CA) authentication lets you use certificates from an external public certificate authority with Anaplan APIs and Anaplan Connect. To use CA authentication with Anaplan APIs and Anaplan Connect, your users must submit certificates with a valid, recognized external public CA as the root certificate. For more information, see Administration: Certificates in Anapedia.

Storing sensitive pieces of information like your certificate private keys in the filesystem can be insecure. It's good practice to use a Java KeyStore (JKS) to protect the private key and certificate with a KeyStore password. Anaplan Connect can work with such KeyStores if they have been created for Anaplan Connect to consume.

Create the KeyStore

You need the following command-line tools for this procedure:

Tool	Location
openssl	https://www.openssl.org/source/
keytool	https://sourceforge.net/projects/keytool/

Windows users must run commands through PowerShell as Administrator in Windows.

1. Create a PKCS12 bundle called "*keystore bundle.p12*" from the Private-Key and the CA Certificate.

   Make sure you use a suitable keystore alias to uniquely identify the entry as this name is required for the -*keystorealias* argument when you run Anaplan Connect.

   ```
   $ openssl pkcs12 -export -in <CERTIFICATE.pem> -inkey <PRIVATE_
   KEY.pem> -out keystore_bundle.p12 -name
   <KEYSTORE_ALIAS> -CAfile <CERTIFICATE.pem> -caname root
   ```

   You are prompted for your Keystore password.

2. Use the PKCS12 bundle to create the JKS keystore called "my_keystore.jks".

   ```
   $ keytool -importkeystore -deststorepass <KEYSTORE-PASSWORD> -
   destkeystore my_keystore.jks
   -srckeystore keystore_bundle.p12 -srcstoretype PKCS12
   ```

   You are prompted for your Keystore password. The *.jks* file contains your KeyStore that securely holds the **Private-Key** and the **CA Certificate**.

Use Keystore with Anaplan Connect

Use the Keystore with Anaplan Connect by providing to the *-keystore, -keystorepass* and *-keystorealias* arguments with values used when building the Keystore.

Example Run script

```
#!/bin/sh
# This example runs a Keystore with Anaplan Connect
CACertPath="/path/to/CA.crt"
KeyStorePath="/path/to/keystore.jks"
KeyStorePass="your_password"
KeyStoreAlias="your_test_alias"
WorkspaceId="<WORKSPACE_ID>"
Operation="-debug -service 'https://api.anaplan.com' -auth
'https://auth.anaplan.com' -w ${WorkspaceId} -M"
#________________ Do not edit below this line __________________
if [ "${CACertPath}" ]; then
  Credentials="-keystore ${KeyStorePath} -keystorepass ${KeyStorePass} -
keystorealias ${KeyStoreAlias}"
  #Credentials="-certificate ${CACertPath} -keystore ${KeyStorePath} -
keystorepass ${KeyStorePass}
  -keystorealias ${KeyStoreAlias}"    # THIS IS ANOTHER APPROACH OF
PROVIDING THE RAW CERTIFICATE VIA -certificate
fi
echo cd "`dirname "$0"`"
cd "`dirname "$0"`"
if [ ! -f AnaplanClient.sh ]; then
  echo "Please ensure this script is in the same directory as
AnaplanClient.sh." >&2
  exit 1
elif [ ! -x AnaplanClient.sh ]; then
```

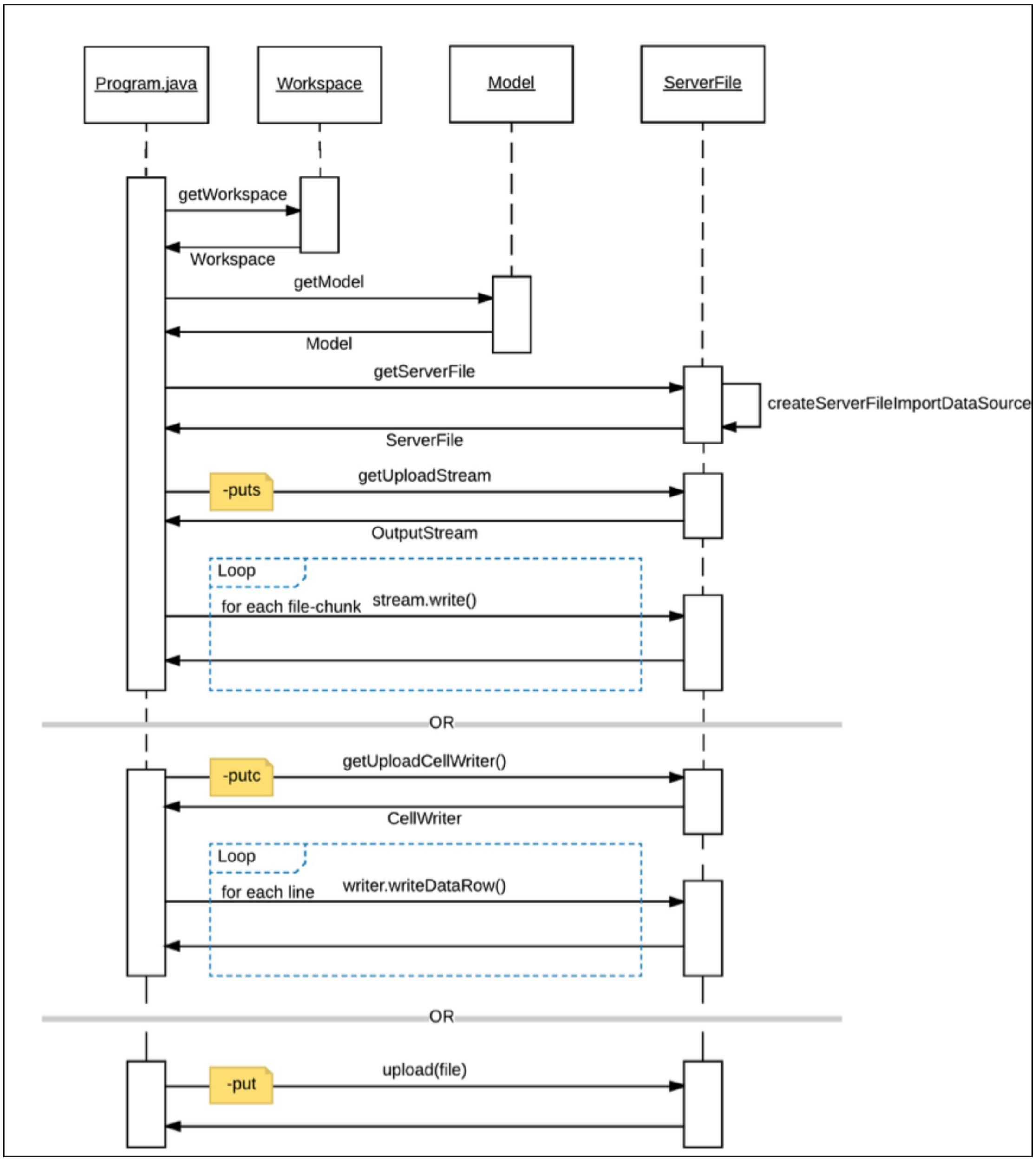
```
  echo "Please ensure you have executable permissions on
AnaplanClient.sh." >&2
  exit 1
fi
Command="./AnaplanClient.sh ${Credentials} ${Operation}"
/bin/echo "${Command}"
exec /bin/sh -c "${Command}"
```

Upload and Download Files

Files are uploaded to a location on a customer-hosted machine, where the import script can access them before they are imported into Anaplan.

The -chunksize parameter allows you to configure the upload chunk size parameter between 1 and 50MB to accommodate large file uploads.

Upload sequence diagram

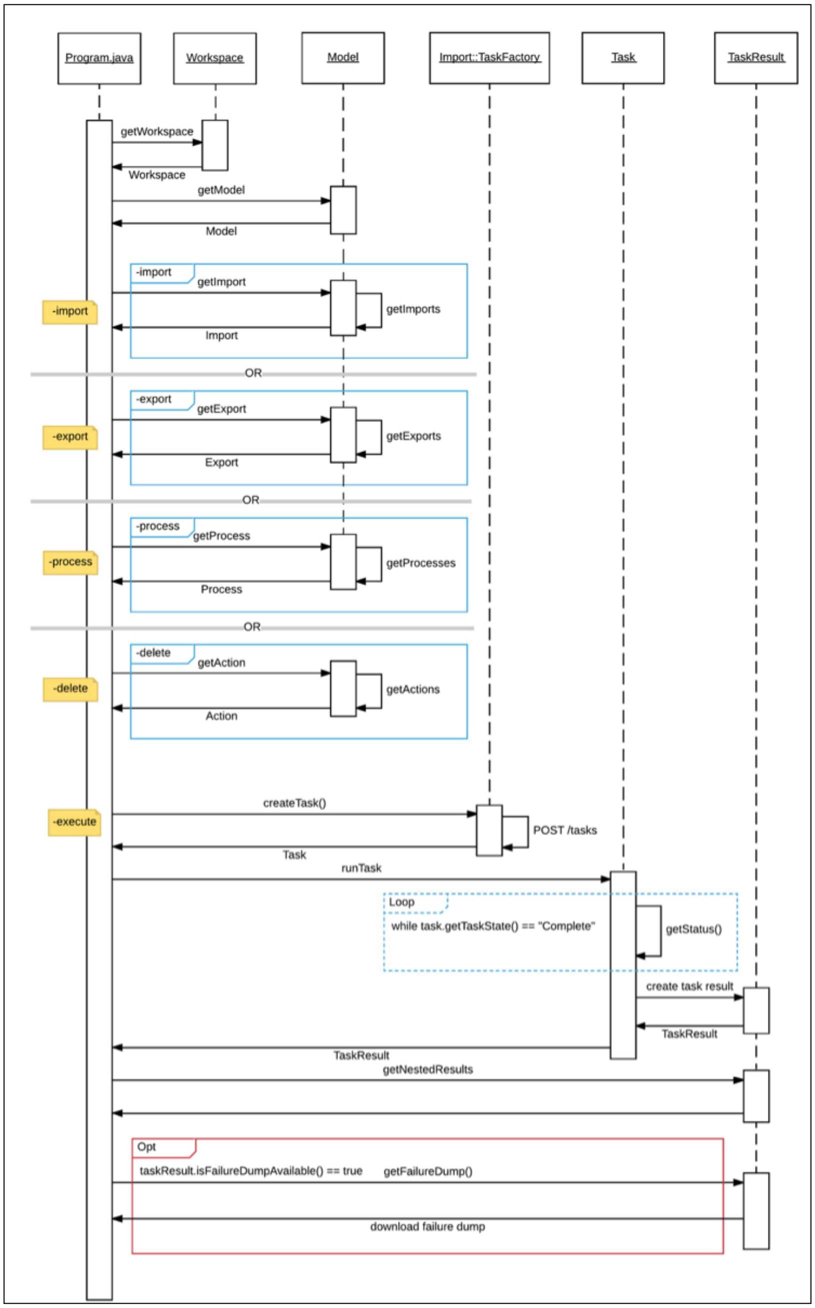


Example batch file for upload

```
#!/bin/sh
# This example uploads a file and runs an import
AnaplanUser="user@anaplan.com"
AnaplanPassword="Password"
set WorkspaceId="8a1234567897c12b014bf01234567890"
set ModelId="CD1234D60CA84E9A123C1C5D061C1234"
Operation="-debug -service 'https://api.anaplan.com' -auth
'https://auth.anaplan.com' -file 'file_to_upload.csv' -chunksize 1 -put
'file_to_upload.csv' -output 'errors.txt'"
#____________________________ Do not edit below this line ________________
______________
if [ "${AnaplanUser}" ]; then
  Credentials="-user ${AnaplanUser}:${AnaplanPassword}"
fi
echo cd "`dirname "$0"`"
cd "`dirname "$0"`"
if [ ! -f AnaplanClient.sh ]; then
  echo "Please ensure this script is in the same directory as
AnaplanClient.sh." >&2
  exit 1
elif [ ! -x AnaplanClient.sh ]; then
  echo "Please ensure you have executable permissions on
AnaplanClient.sh." >&2
  exit 1
fi
Command="./AnaplanClient.sh ${Credentials} -workspace ${WorkspaceId} -
model ${ModelId} ${Operation}"
/bin/echo "${Command}"
exec /bin/sh -c "${Command}"
```

Create an Import Batch Script

Import/Export Sequence Diagram



Example batch file for import

```
@echo off
rem This example loads a source text file and runs an Anaplan import into
a module.
rem For details of how to configure this script see doc\Anaplan Connect
User Guide.doc
set AnaplanUser=anaplan.user@anaplan.com:Password
set WorkspaceId="8a1234567897c12b014bf01234567890"
set ModelId="CD1234D60CA84E9A123C1C5D061C1234"
set Operation=-file "Employee.txt" -put
"C:\AnaplanConnect\Import\Employee.txt"
-import "New Hire from Employee.txt" -execute -output "C:\My Errors.txt"
rem *** End of settings - Do not edit below this line ***
setlocal enableextensions enabledelayedexpansion || exit /b 1 cd %~dp0
if not %AnaplanUser% == "" set Credentials=-user %AnaplanUser%
set Command=.\AnaplanClient.bat %Credentials% -workspace %WorkspaceId% -
model %ModelId% %Operation%
@echo %Command%
cmd /c %Command%
pause
```

Set the Import Operation

As a best practice, the name of the Import action should indicate the name of the file (or other source) from which data will be imported, such as *Import From Employee.txt*.

Example:

```
set Operation=-file "Employee.txt" -put
"C:\AnaplanConnect\Import\Employee.txt"
-import "New Hire from Employee.txt" -execute -output
"C:\ImportDumpFilesDirectory"
```

where:

-file “Employee.txt”	indicates the action uses a file named Employee.txt. Note that .csv format is also supported.
-put “C:\AnaplanConnect\Import\Employee.txt"	upload the file to the specified absolute or relative path to the Anaplan Server.
-import “New Hire from Employee.txt”	an Import action with the specified name. To see this name in Anaplan, in **Settings** click **Actions** and view the list of Imports. As a best practice, name your import (or export) action such that it matches the name of the file.
-execute	runs the action
-output "C:\ImportDumpFilesDirectory"	generates a file at the specified directory that lists any Import errors. This is optional. This file is created only if at least one Import error occurs.

Linux example for Import

In Linux or Mac OS, use single-quotes instead of double quotes. In this example, compare the double-quotes of Windows with the single-quotes of Linux and Mac OS.

Windows	-put "C:\testdata\Europe P&L.txt" or: -p "C:\testdata\Europe P&L.txt"
Linux or Mac OS	-put '/Users/user1/testdata/Europe P&L.txt' or: -p '/Users/user1/testdata/Europe P&L.txt'

Linux Import code

```
#!/bin/sh # This example uploads a file and runs an import
AnaplanUser=firstname.lastname@yourcompany.com
WorkspaceId="yourWorkspaceId"
ModelId="yourModelId"
Operation="-file 'file-to-import.csv' -put
'/path/to/anaplan-connect/file-to-import.csv' -import
'ImportDefinitionName'
-execute -output 'MyImportErrors.txt'"
```

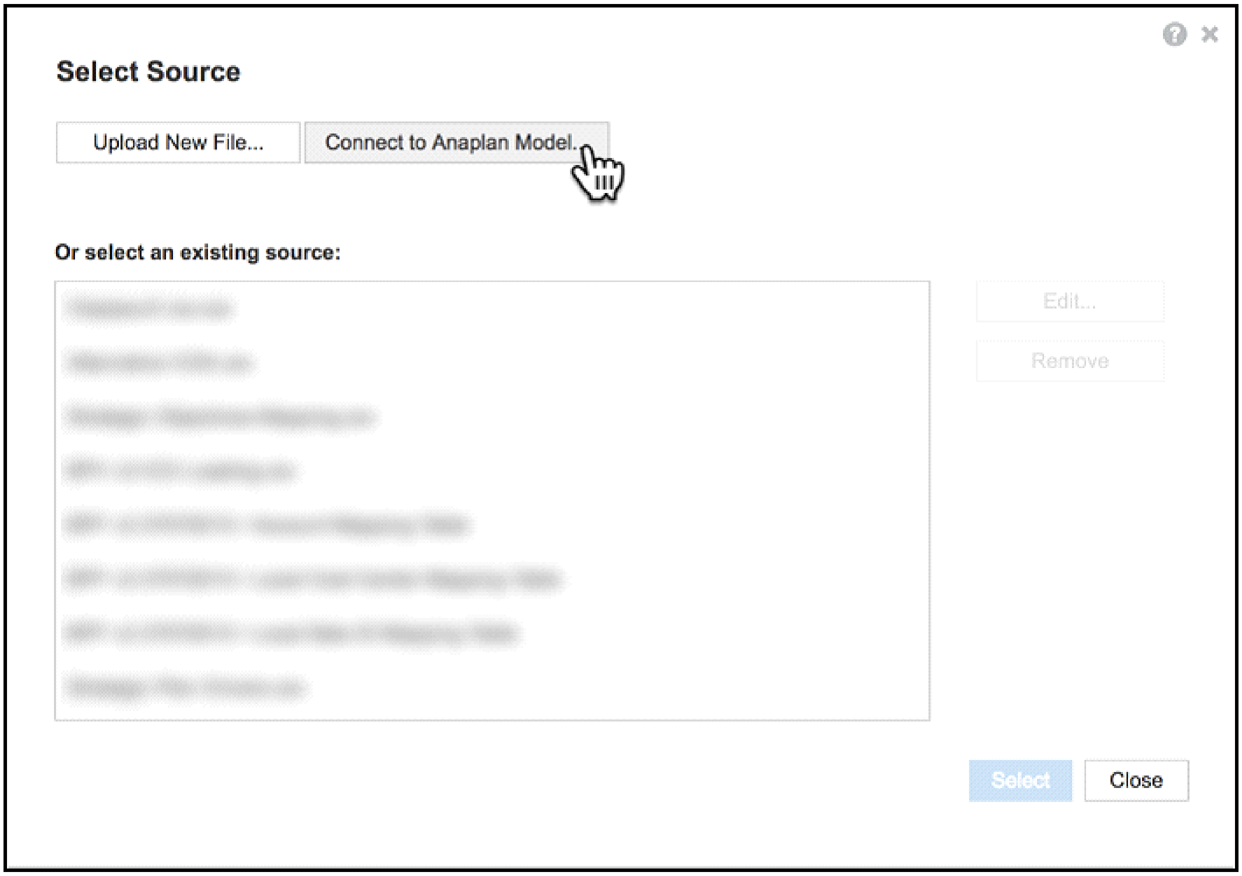
To run the shell script, at the command prompt enter:

./RunMyImport.sh

Model-to-model Import

This example runs a model-to-model import within Anaplan, transferring data from the **Installation Sales** module in Model2 to the **P&L** module in *MyBudgetModel*.

1. Run the import manually. On the **Data** menu, click **Import** then click **Connect to Anaplan Model**.



2. Select a module (or list) as the source of the import.
3. Click **Run Import** then edit and run the batch file.

   In the model, note the **Import ID** that is used in the batch file, which in the example below is *P&L from Model2 / Installation Sales*.

   There is a space before and after the forward slash "/" in the string ‘P&L from Model2 / Installation Sales’.

```
@echo off
rem This example runs a model to model import within Anaplan.
set AnaplanUser="firstname.lastname@company.com"
set WorkspaceId="8a819488459fa63301462b73fe785786"
set ModelId="CB0A5A4D5C5943B5837FF42C5FAA95E1"
set Operation=-import "P&L from Model2 / Installation Sales" -execute
rem *** End of settings - Do not edit below this line ***
```

Create an Export Batch Script

Example Batch file for Export

```
@echo off
set AnaplanUser="Anaplan.User@anaplan.com:Password"
set WorkspaceId="8a1234567897c12b014bf01234567890"
set ModelID="CD1234D60CA84E9A123C1C5D061C1234"
set Operation=-export "Employee by Department.xls" -execute -get
"C:\Employee.xls"
rem *** End of settings - Do not edit below this line ***
setlocal enableextensions enabledelayedexpansion || exit /b 1
cd %~dp0
if not %AnaplanUser% == "" set Credentials=-user %AnaplanUser%
set Command=.\AnaplanClient.bat %Credentials% -workspace %WorkspaceId% -
model%ModelId% %Operation%
@echo %Command%
cmd /c %Command%
pause
```

Set the Export Operation

Example:

```
set Operation=-export "Employee by Department.xls" -execute -get
"C:\Employee.xls"
```

where:

-export "Employee by Department.xls"	indicates that an Export action exists and is named "Employee by Department.xls"
-execute	runs the Export
-get "C:\Employee.xls"	creates a new file with the exported data at the specified path

An Export action cannot generate an error file, so we do not specify a path for it.

Linux Example for Export

```
# This example runs an export, then downloads the file to the client.
# In the Operation, specify the name of the Export definition before
# giving the path to client file the server will create.
AnaplanUser=firstname.lastname@company.com
WorkspaceId="8a819488459fa63301462b73fe785786"
ModelId="85EFA3E719AF49E183118A58C644D802"
Operation="-export 'company-summary-export-definition' -execute -get
'/Users/username/Downloads/Company Summary.csv'"
```

Create a File to Run Other Actions

Example Batch file for Delete

```
@echo off
rem This example deletes obsolete customers from a list
set AnaplanUser=firstname.lastname@company.com
set WorkspaceId="8a1234567897c12b014bf01234567890"
set ModelId="CB0A5A4D5C5943B5837FF42C5FAA95E1"
set Operation=-action "Delete from Customers Using Obsolete" -execute
rem *** End of settings - Do not edit below this line ***
```

Set the Delete operation to delete items from a list

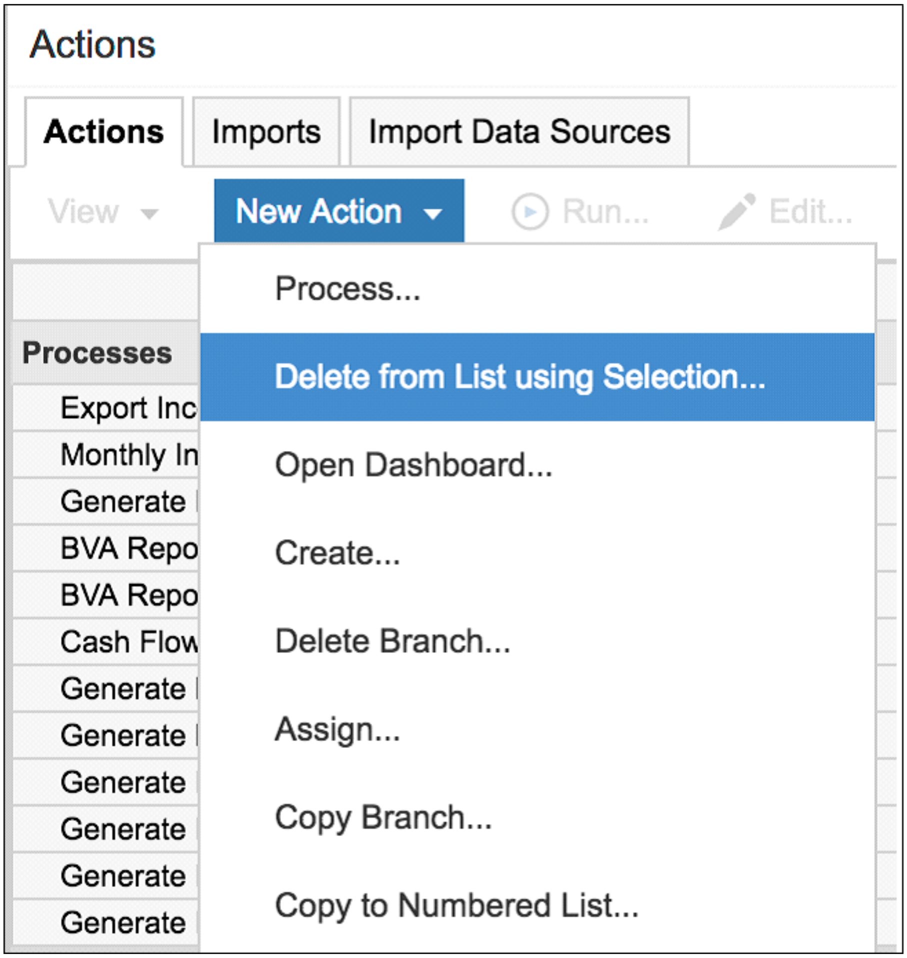
This example shows how to automate deleting items from a list based on Boolean criteria.

For example, you can automatically delete items in a list that are now obsolete, such as customers with a rating less than, or equal to, 2. The Obsolete line item has Boolean data type and must be set up to be a formula:

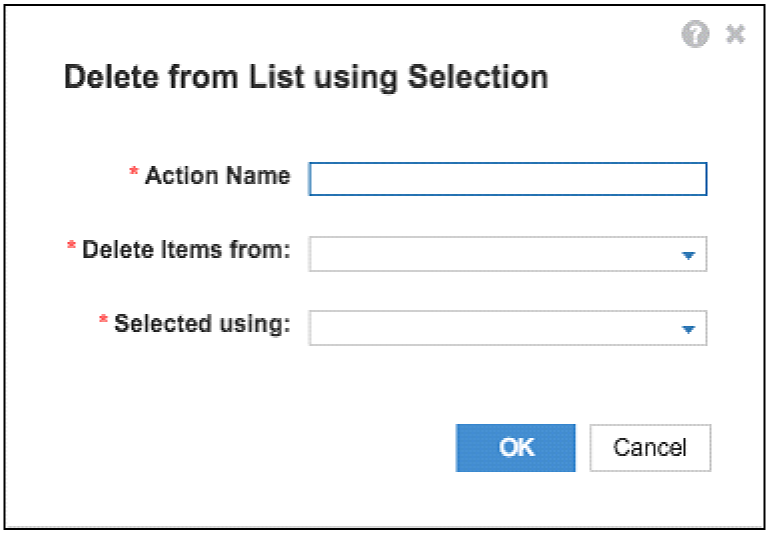
```
Obsolete = Rating <= 2
```

Line items of Boolean data type that only have the dimensionality of the list can be used as the criteria to determine which items to delete.

1. In Anaplan, go to **Model Settings > Actions**.
2. On the **New Action** list, click **Delete from List using Selection**.



The Delete from list using Selection dialog appears.



3. In the **Action Name** box, type the text that you want to appear on the button.
4. On the **Delete Items from** list, select the list (for example *Customers*) from which you are deleting items.
5. On the **Selected using** list, select the line item that contains the selection criteria (for example *Customer Rating.Obsolete*).
6. Click **OK**. The action appears under **Other Actions**. Use the name of this action in your batch script.

See Delete from List using Selection in Anapedia.

Example:

```
set Operation=-action "Delete from Customers Using Obsolete" -execute
```

where:

-action "Delete from Customers Using Obsolete"	an Other Action (neither Import nor Export) named Delete from Customers Using Obsolete
-execute	runs the Delete

Create a Batch Script to run a Process

A Process is a combination of Imports, Exports, and/or Deletes.

The process must exist within Anaplan before calling it in the Anaplan Connect script.

An example batch file for a process that contains *Import* and *Export* actions shown below . Note that the process can also contain *Delete* actions:

```
@echo off
rem This example runs a Process that contains multiple Export actions
set AnaplanUser="firstname.lastname@company.com"
set WorkspaceId="8a1234567897c12b014bf01234567890"
set ModelId="CB0A5A4D5C5943B5837FF42C5FAA95E1"
set Operation= -file "file1.txt" -put"C:\Files\ImportModule.csv" -file
"file2.txt" -put "C:\Files\example.csv"
-process "myprocess" -execute -file "ExportList" -get
"C:\Files\ExportList.csv" -file "ExportModule"
-get "C:\Files\ExportModule.csv" -output
"C:\MyDirectoryForImportDumpFiles"
rem *** End of settings -Do not edit below this line ***
```

Set the Process Operation

```
set Operation=-file "file1.csv" -put "C:\Files\ImportModule.csv" -file
"file2.csv" -put "C:\Files\example.csv"
-process "myprocess" -execute -file "ExportList" -get
"C:\Files\ExportList.csv"
-file "ExportModule" -get "C:\Files\ExportModule.csv" -output
"C:\MyDirectoryForImportDumpFiles"
```

Where:

-file "file1.csv" -put "C:\Files\ImportModule.csv"	Uploads the local file *C:\Files\ImportModule.csv* to the Anaplan server so that its data can be stored into the Import data source named *file1.csv*.

-file "file2.csv" -put "C:\Files\ImportModule.csv"	Uploads the local file *C:\Files\ImportModule.csv* to the Anaplan server so that its data can be stored into the Import data source named *file2.csv*.
-process "myprocess" -execute	Runs the process named *myprocess*
-file "ExportList" -get "C:\Files\ExportList.csv"	Downloads the data last exported by the export *ExportList* to the local file *C:\Files\ExportList.csv*
-file "ExportModule" -get "C:\Files\ExportModule.csv”	Downloads the data last exported by the export *ExportModule* to the local file *export/ProcessExportModule.csv*
-output "C:\MyDirectoryForImportDumpFiles"	Generates a file for each import action within the specified directory that lists one or more import errors

End Users versus Workspace Administrators

An end user can run the same actions through Anaplan Connect that the end user can run manually.

	End user can run..	Workspace Admin can create and run..
Import	• model-to-model import • list imports - requires write access to the target list • module imports - requires write access to the target module	any type of import, including imports that involve uploading external files or data
Export	Requires read access to the module or list.	export
Delete	Requires write access to the list.	delete
Process	Requires access to the actions in the process	process

	End user can run..	Workspace Admin can create and run..
Info		• Can change the **model**, unless the role is *No Access* to a particular model, which also prevents the Workspace Admin from finding the model. • Can have a role that has access to no **module** but can grant self rights to the module.

Scheduling an import or export

A batch file that runs the import or export can be scheduled to run at a specific time, as a one-time operation, or recurring at the interval you choose, such as daily, weekly, or monthly. The scheduler is not part of Anaplan Connect, and the scheduling program and set-up depends on your operating system; the computer must be running at the scheduled time.

In the batch file, the password needs to be appended to the Anaplan user name and enclosed in double quotes. Alternatively use certificate authentication.

```
set AnaplanUser=firstname.lastname@company.com:"mysecretpassword"
```

Windows

Optionally, you can remove the pause command at the end of the batch file.

The pause command leaves the messages on the screen that record what the batch file has done.

Scheduler for Windows XP

This example shows the steps involved on a Windows XP operating system to schedule an import on a specific day and time, once a month:

- Start > All Programs > Accessories > System Tools > Scheduled Tasks > Add Scheduled Tasks > Next > Browse

```
C:\anaplan-connect-1-4\nameofscript.bat
```

- Monthly > Next > The First Monday at 05.00 > Enter name & password for the PC > **Finish**

Scheduler for Windows 7

The Scheduler in Windows 7 is almost the same as XP:

- Start > All Programs > Accessories > System Tools > Task Scheduler > Create Basic Task > *Name the task* > Next > *Set when to trigger the task* > Next > Select Start a program and browse to *C:\anaplan-connec\nameofscript.batt* > Next > **Finish**

Linux or Mac OS

Consider using a job scheduling utility for UNIX-like operating systems, such as cron.

Log Files in Anaplan Connect

Logging files generated by Anaplan Connect in v1.4 and later use logback to provide file output.

The log line format is:

YYYY-MM-DD hh:mm:ss <Class Path> <PID> |-- <Logging statement description>

Where:

Element	Description
YYYY	Year in four digit format.
MM	Month in two digit format
DD	Day of the month in two digit format
hh	Hour in 24-hour format
mm	Minutes
ss	Seconds
<Class Path>	The shortened class path. For example: DEBUG [c.a.client.Program :1594] or INFO [a.BasicAuthenticator:26]
<PID>	The UNIX process ID. **Note**: On Windows systems, this entry may be blank.
|--	A delimiter between the basic logging information and the logging statement description.
<Logging statement description>	A descriptive logging statement.

Example:

```
2018-05-06 20:30:18 DEBUG [c.a.client.Program  :1594 ] 96461 |--
=====================================================================
2018-05-06 20:30:18 DEBUG [c.a.client.Program  :1595 ] 96461 |--   Anaplan
Connect 2.0.0-release
2018-05-06 20:30:18 DEBUG [c.a.client.Program  :1596 ] 96461 |--   Java
```

```
HotSpot(TM) 64-Bit Server VM (Oracle Corporation)/ (25.60-b23)/
2018-05-06 20:30:18 DEBUG [c.a.client.Program  :1598 ] 96461 |--   (Mac OS
Xx86_64)/10.13.4
2018-05-06 20:30:18 DEBUG [c.a.client.Program  :1599 ] 96461 |--
================================================================
2018-05-06 20:30:18 INFO  [c.a.client.Service  :88   ] 96461 |--
Initializing Service...
2018-05-06 20:30:18 INFO  [a.BasicAuthenticator:26   ] 96461 |--
Authenticating via Basic...
2018-05-06 20:30:21 INFO  [c.a.c.ServerFile    :259  ] 96461 |--
Uploading file: /Users/USERX/Downloads/gross_sales_actuals.csv
2018-05-06 20:30:33 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 0 (size=30MB)
2018-05-06 20:30:39 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 1 (size=30MB)
2018-05-06 20:30:46 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 2 (size=30MB)
2018-05-06 20:30:52 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 3 (size=30MB)
2018-05-06 20:30:57 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 4 (size=30MB)
2018-05-06 20:31:03 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 5 (size=30MB)
2018-05-06 20:31:09 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 6 (size=30MB)
2018-05-06 20:31:15 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 7 (size=30MB)
2018-05-06 20:31:21 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 8 (size=30MB)
2018-05-06 20:31:26 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
```

```
Uploaded chunk: 9 (size=30MB)
2018-05-06 20:31:32 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 10 (size=30MB)
2018-05-06 20:31:38 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 11 (size=30MB)
2018-05-06 20:31:43 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 12 (size=30MB)
2018-05-06 20:31:49 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 13 (size=30MB)
2018-05-06 20:31:55 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 14 (size=30MB)
2018-05-06 20:31:59 DEBUG [c.a.c.ServerFile    :306  ] 96461 |--
Uploaded chunk: 15 (size=30MB)
2018-05-06 20:31:59 INFO  [c.a.client.Program  :464  ] 96461 |--   The
file "/Users/USERX/Downloads/gross_sales_actuals.csv"
    has been uploaded as gross_sales_actuals_DEC15.csv.
2018-05-06 20:32:00 INFO  [c.a.c.TaskFactory   :116  ] 96461 |--
Creating Import task: 112000000030
2018-05-06 20:32:02 INFO  [c.a.client.Task     :207  ] 96461 |--   Run
status: Retrieving data from gross_sales_actuals_DEC15.csv
2018-05-06 20:32:03 INFO  [c.a.client.Task     :207  ] 96461 |--   Run
status: Retrieving data from gross_sales_actuals_DEC15.csv
2018-05-06 20:32:04 INFO  [c.a.client.Task     :207  ] 96461 |--   Run
status: Retrieving data from gross_sales_actuals_DEC15.csv
2018-05-06 20:32:05 INFO  [c.a.client.Task     :207  ] 96461 |--   Run
status: Retrieving data from gross_sales_actuals_DEC15.csv
2018-05-06 20:32:07 INFO  [c.a.client.Task     :207  ] 96461 |--   Run
status: Retrieving data from gross_sales_actuals_DEC15.csv
2018-05-06 20:32:08 INFO  [c.a.client.Task     :207  ] 96461 |--   Run
status: Retrieving data from gross_sales_actuals_DEC15.csv
```

```
2018-05-06 20:32:09 INFO  [c.a.client.Task     :207  ] 96461 |--   Run
status: Retrieving data from gross_sales_actuals_DEC15.csv
2018-05-06 20:32:10 INFO  [c.a.client.Task     :207  ] 96461 |--   Run
status: Retrieving data from gross_sales_actuals_DEC15.csv
2018-05-06 20:32:12 INFO  [c.a.client.Task     :207  ] 96461 |--   Run
status: Retrieving data from gross_sales_actuals_DEC15.csv
2018-05-06 20:32:13 INFO  [c.a.client.Task     :207  ] 96461 |--   Run
status: Retrieving data from gross_sales_actuals_DEC15.csv
2018-05-06 20:32:14 INFO  [c.a.client.Task     :207  ] 96461 |--   Run
status: Retrieving data from gross_sales_actuals_DEC15.csv
2018-05-06 20:32:25 INFO  [c.a.client.Task     :207  ] 96461 |--   Run
status: Retrieving data from gross_sales_actuals_DEC15.csv
2018-05-06 20:32:36 INFO  [c.a.client.Task     :207  ] 96461 |--   Run
status: Generating failure dump (100.0%)
2018-05-06 20:32:46 INFO  [c.a.client.Task     :207  ] 96461 |--   Run
status: Complete. (100.0%)
2018-05-06 20:32:46 INFO  [c.a.client.Task     :228  ] 96461 |--   - - - -
- - - - - - - - - - - - - - - - - - - - - - - - - -
2018-05-06 20:32:46 INFO  [c.a.client.Task     :230  ] 96461 |--   The
operation was successful.
2018-05-06 20:32:46 INFO  [c.a.client.Task     :3880 ] 96461 |--   large_
list_USERX: 2 (0/2) rows successful, 1179070 ignored
2018-05-06 20:32:46 INFO  [c.a.client.Task     :3880 ] 96461 |--   Another
row has already been processed with this key: 1179070
2018-05-06 20:32:46 INFO  [c.a.client.Task     :235  ] 96461 |--   - - - -
- - - - - - - - - - - - - - - - - - - - - - - - - -
2018-05-06 20:32:46 INFO  [c.a.c.ServerFile    :109  ] 96461 |--
Downloading file /Users/USERX/projects/anaplan-connect/errors.txt
2018-05-06 20:32:46 DEBUG [c.a.c.TaskResult    :127  ] 96461 |--
Fetching Import action's dump file chunks for
```

```
   task=A96ADE416B594080BD8E7F7FB25BA51B
2018-05-06 20:32:47 DEBUG [c.a.c.TaskResult    :138   ] 96461 |--
Downloading dump data-chunk 0
2018-05-06 20:32:51 DEBUG [c.a.c.TaskResult    :138   ] 96461 |--
Downloading dump data-chunk 1
2018-05-06 20:32:55 DEBUG [c.a.c.TaskResult    :138   ] 96461 |--
Downloading dump data-chunk 2
2018-05-06 20:32:57 INFO  [c.a.client.Program  :714   ] 96461 |--   Dump
file written to "errors.txt"
```

Troubleshooting Tips

Getting Debug Information

To get verbose command-line output that might be useful for debugging, include the **-debug** argument at the beginning of the Operation statement. If your script fails to run, enable debug logging and save the output to a text file. If you raise a Support case for assistance, providing the output file of your debug logging will aid the Support team in helping to resolve the problem.

```
set Operation="-debug -file 'file-to-import.csv' -put '/path/to/anaplan-
connect/file-to-import.csv'
-import 'Organization from Salesforce' -execute -output
'MyImportErrors.txt'"
```

Symptoms and Remedies

Symptom	Remedy
Error message: *“.\AnaplanClient.bat is not recognized as an internal or external command, operable program or batch file.”*	Move the .bat file to the root of your Anaplan Connect installation. This location guarantees that Anaplan Connect can use any batch file you create.

Symptom	Remedy
The batch file appears to run, but numbers do not appear to come through into Anaplan	- Try changing page, refreshing the browser, or closing and re-opening the target module. The target module might have already been open when you ran the batch file, in which case it would need some event to trigger recontacting the Anaplan server, which refreshes the view with the latest data. - Check syntax details. For example: - variables such as the file name are enclosed in straight double quotes " " (as typed in Notepad), not curly double quotes “ ” (as typed in MS Word). - A percent sign (%) anywhere in a batch file (such as in a file or import name) probably needs an escape character preceding it. - A leading dollar sign ($) in a shell script requires a preceding escape character unless it is inside single quotes. - These characters require an preceding escape character: - a single quote (') - A single quote around a word containing an apostrophe ('Jane's') - double-quote ("'Jane'") - backtick \\ - backquote \\ - accent grave (`). - A common problem for .sh files is not having execute permissions on the file you want to copy. You can grant

Symptom	Remedy
	execution permission with the command chmod +x filename.sh
Model-to-model import not running or not recognized.	Make sure there is a space before and after the / in the syntax - import "*Target Module from Source Model / Source Module*"
If you're unable to perform a Delete action on items from a list	Check that the list does not contain summary items or subtotals. Such lists cannot use the Delete action (due to the difficulty of dealing with *orphaned* subtotals that lack children). Lists that have parent hierarchies or top-level items can use the bulk delete action, provided that the list that you are editing does not have subtotals.
If you're performing a SQL query from a Windows machine with the -jdbcquery option, and using the percent character (%) as the wildcard character in a pattern for the like operator	The Windows command processor might perform variable substitution on an expression like '%a%', even though no variable a has been defined, resulting in an empty pattern. If so, escape the % sign with %%. For example, %a% is escaped by %%a%%

Appendix A: Network Drive as Location for Anaplan Connect

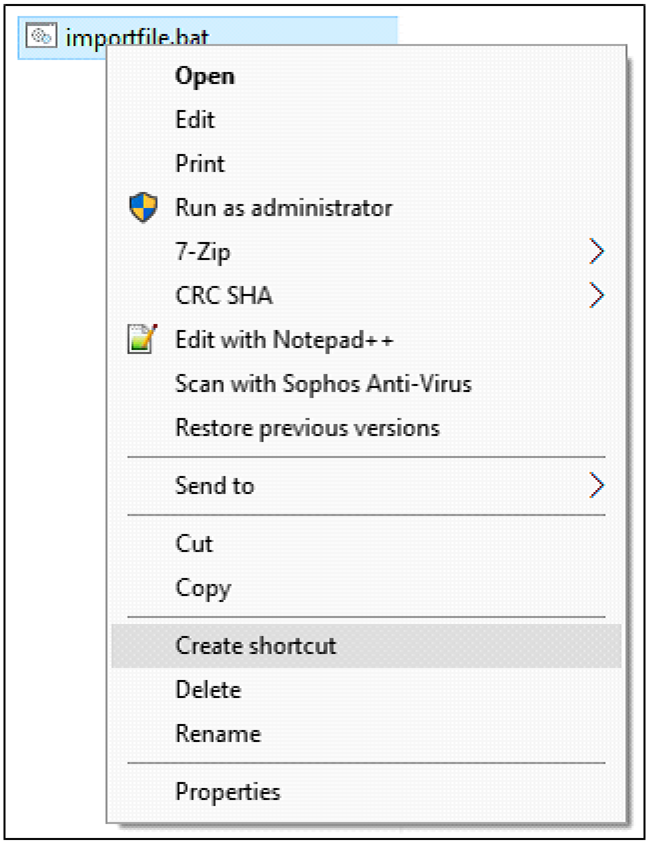
This example is for Windows.

1. Put a copy of example.bat (in the Anaplan Connect examples folder) into the main Anaplan Connect folder.
2. Replace line 13:

   cd %~dp0

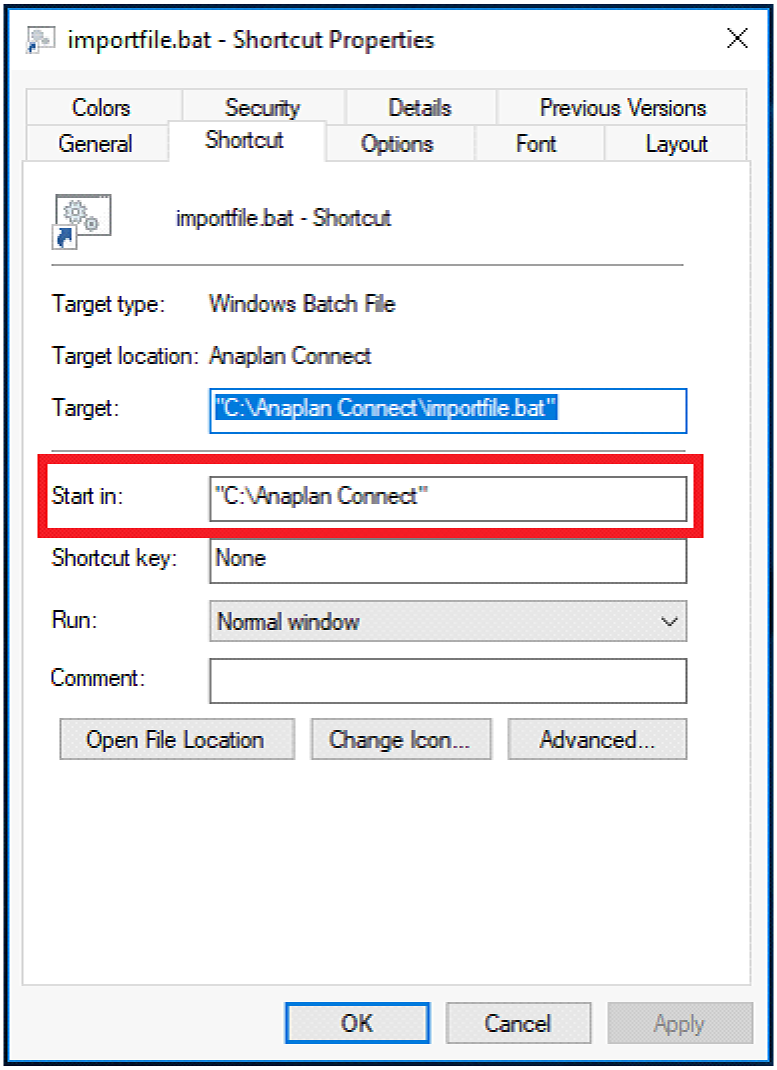
   with:

   pushd %~dp0
3. Before the pause line, insert **popd**.
4. Create a shortcut to the batch file in the same directory. Right-click and select **Create shortcut**.



5. Once the shortcut is created, right-click on the shortcut and select **Properties**.

6. On the **Shortcut** tab, in the **Start** in box type the local directory that runs Anaplan Connect. For example, *%USERPROFILE%* can substitute the user's profile folder.



Target type: Windows Batch File

Target location: Anaplan Connect

Target: "C:\Anaplan Connect\importfile.bat"

Start in: "C:\Anaplan Connect"

Shortcut key: None

Run: Normal window

Comment:

When given a UNC path (\\*computer**share*\...), the **pushd** maps the share to a drive, typically Z: or the last unmapped drive letter. The **popd**unmaps the drive and returns to the original location. If the command window is closed before the program completes, the drive remains mapped.

Appendix B: Java Compatibility

Anaplan Connect supports Java 7 and Java 8. Note that Java 6 is not supported.

We strongly recommend you upgrade to Java 8 to benefit from the security offered by TLS 1.2 if your organization uses Java 6.x or 7.x with Anaplan Connect.

ODBC is officially deprecated in Java 8. Make sure you update any ODBC connections to use JDBC. See Appendix D: JDBC for Oracle, Access, MySQL, Excel and Appendix E: Import through a JDBC Connection for a Microsoft SQL Server database.

Use one of the options below to upgrade to Java 8.

Create a shell script to set the JAVA_HOME environment variable

Write a shell script that sets the **JAVA_HOME** environment variable to the location of the Java 8 Runtime Environment you want to use. You only need to carry out this change to run the Anaplan Connect script.

Create a replacement script

If you've already installed Anaplan Connect you can create your own replacement script. The instructions in this section do not change the version of Java you use for your other applications.

To create the replacement script:

1. Navigate to the Anaplan Connect directory. For example, on Windows, the directory might be *C:\anaplan-connect-1-4*.
2. Make a backup copy of the script that calls Anaplan Connect:
   - **Windows**: Make a copy of *AnaplanClient.bat* and name it *AnaplanClient.bat-OLD*.
   - **Linux/MacOS**: Make a copy of *AnaplanClient.sh* and name it *AnaplanClient.sh-OLD*.
3. Edit the script that calls Anaplan Connect:
   - **Windows**: In **AnaplanClient.bat** replace *%JAVA%* with the version directory of Java 8 to use for Anaplan Connect.
   - **Linux/MacOS**: In **AnaplanClient.sh** replace *${java}* with the version directory of Java 8 to use for Anaplan Connect.

	Windows.bat file	Linux/MacOS.sh file
Original	rem Start the Java virtual machine "%JAVA%" %JAVA_OPTS% -classpath "%CP%" com.anaplan.client.Program %*	# Start the Java virtual machine exec ${JAVA_OPTS} -classpath "${classpath}" "${java}" com.anaplan.client.Program "$@"
Change to Java 8	rem Start the Java virtual machine "C:\Program Files\Java\jre1.8.0_66\bin\java" %JAVA_ OPTS% -classpath "%CP%" com.anaplan.client.Program %*	# Start the Java virtual machine for MacOS involves /Library exec "/Library/Java/JavaVirtualMachines/jdk1.8.0_60.jdk ${JAVA_OPTS} -classpath/Contents/Home/bin/java" "${classpath}" com.anaplan.client.Program "$@"

The directory name on your computer might differ from these examples. For more information, visit the Data Integration Knowledge Base on Anaplan Community.

Appendix C: List of all Operation Commands

Navigate to the installation directory and type the following to get a list of the operation commands:

Windows	C:\Windows\AnaplanConnect\AnaplanClient.bat -version Java AnaplanClient -version
Java	AnaplanClient -version
Linux/Mac OS	./AnaplanClient.sh ./AnaplanClient.sh -version

The following table shows the commands for the operation line of the batch file. The abbreviated syntax can be used to reduce typing, for example -x instead of -execute. Some operations are followed by a variable, such as a path to a file. For example, -put "*C:\testdata\Europe P&L.txt" or -p "C:\testdata\Europe P&L.txt*"

Syntax	Abbreviated Syntax	Followed by a variable, if applicable	What it does
-help	-h		Display this help
-chunksize	-ch		Determines the size of chunks for import in bytes. Select a whole non decimal number between 1 - 50MB (defaults to 1MB). Note: The -chunksize parameter must be specified before the -put parameter. Example: In the following example we have set the chunksize to 5MB. Operation="-file 'My Source.txt' -chunksize 5 -put 'data/My Source.txt' -import 'My Module from My Source.txt' -execute -output 'data/My Errors.txt'"
-debug	-d		Show detailed (verbose) output, which can help you debug any problems. See Getting Debug Information.
-quiet	-q		Show less detailed output
-service	-s		API service endpoint

Syntax	Abbreviated Syntax	Followed by a variable, if applicable	What it does
-user	-u	username:"password"	Anaplan user name and optional password in the format username:"password". If the batch file (or shell script) does not set a value for the AnaplanUser, the program prompts the user to supply the username, then the password, enclosed in double-quotes.
-certificate	-c	Pathname on local machine	Path to user certificate used for authentication (an alternative to using a key store)
-keystore	-k	Pathname on local machine	Path to local key store containing user certificate(s) for authentication
-keystorepass	-kp	Password	Password for the key store. If this option is not provided, and the file ~/.anaplan/api-client/keystore-access.txt exists (where ~ is the user's home directory), the password is read and decoded from the contents of this file. Otherwise, the user is prompted for a password. Note that obfuscation is the URL-encoded form of the result of: taking the exclusive -or of each of the characters in the password and the value 129
-keystorealias	-ka	Alias	Alias of the public certificate in the specified key store
-via	-v	Proxy URL	Use specified proxy
-viauser	-vu	username:password	Pass credentials to authenticating proxy. Consider NTLM Authentication using JCIFS

Syntax	Abbreviated Syntax	Followed by a variable, if applicable	What it does
-workspaces	-W		**Note**: This operator is deprecated and will be removed in the next release of Anaplan Connect. To get workspace values from a batch file (or shell script), use the -W flag. # This example lists the workspaces AnaplanUser=firstname.lastname @company .com WorkspaceId= "8a819488472c2c950147411f574a085b" ModelId= "B378C20804234C1BBFBAB0D7EF1CD1D3" Operation= "-W" The output lists Workspace ID followed by Workspace Name: 8a8194824317414b0143540f2d5e357d Finance 8a819488472c2c250147411f574a085b Sales
-workspace	-w	Workspace name or ID	Select a workspace by ID or name
-models	-M		**Note**: This operator is deprecated and will be removed in the next release of Anaplan Connect. List available models in selected workspace To get model values from a batch file (or shell script), use the -M flag. # This example lists the models AnaplanUser=firstname.lastname @company .com WorkspaceId= "8a819488472c2c950147411f574a085b" ModelId= "B378C20804234C1BBFBAB0D7EF1CD1D3" Operation= "-M" The output lists Model ID followed by Model Name: 7C5B30DE5C374163A23ADD6D2B0622F8 Financial Consolidation 569418FFCDF041678273620FFCDE2330 Sales Rep Territory
-model	-m	Model name or ID	Select a model by ID or name
-modules	-MO		List available modules in selected model
-module	-mo	Module name or ID	Select a module by ID or name
-views	-VI		List available views in selected module

Syntax	Abbreviated Syntax	Followed by a variable, if applicable	What it does
-view	-vi	View name or ID	Select a view by ID or name
-files	-F		List available files on the Anaplan server in selected model
-file	-f	File name on Anaplan server	File name on Anaplan server Select a server file by ID or name
-get	-g	Pathname on local machine	Download the specified file
-gets			Write specified server file to standard output
-getc			Write tab-separated server file to standard output
-put	-p	Pathname on local machine	Upload the specified file
-puts			Upload to specified server file from standard input
-putc			Upload to specified server file from tab-separated standard input
-imports	-I		List available imports in selected model The output lists all the import definitions that are available in a given model. The list of imports and exports is also available in Anaplan. To view, on the Settings tab, click Actions.
-import	-i	Import name or ID	Select an import by ID or name
-exports	-E		List available exports in selected model
-export	-e	Export name or ID	Select an export by ID or name
-actions	-A		List available actions in selected model, such as delete actions. This list corresponds to the Other Actions (Settings tab, Actions) list, and does not include Processes, Imports, or Exports.
-action	-a	Action name or ID	Select a saved action. For example, Delete items from a list.

Syntax	Abbreviated Syntax	Followed by a variable, if applicable	What it does
-processes	-P		List available processes in selected model
-process	-pr	Process name or ID	Select a process by ID or name
-locale	-xl	ISO language & country code separated by underscore. For example, 'en_US'	Specify the locale to use when performing the server operation, which affects the available date formats when parsing date values in imports, and the month names when using a specified timescale format. For details, see http://docs.oracle.com/javase/8/docs/api/java/util/Locale.html.
-connectorproperty	-xc	Property identifier and value separated by colon. If value is ?, prompt user	Specify import data source connection property. For example, Salesforce credentials.
-mappingproperty	-xm	Dimension and value separated by colon. If value is ?, prompt user	Specify prompt-at-runtime import mapping value
-execute	-x		Run the preceding -import, -export, -process or -action
-output	-o	Pathname on local machine	Retrieve dump file for completed import
-emd			Get metadata for an export
-maxretrycount	-mrc	Retry count	Maximum retry count for API calls. If not specified, the default is 4 retries.
-retrytimeout	-rt	Timeout in seconds	Retry timeout for HTTP client calls.
-httptimeout	-ct	Timeout in seconds	HTTP client connection timeout. If not specified, the default is 10 seconds.

Appendix D: JDBC for Oracle, Access, and Excel

Provided you have the appropriate JDBC driver, you can link directly into Anaplan from databases such as Oracle, Access, MySQL, or from Excel. Both lists and data can be imported into Anaplan in this way, and when combined with a scheduler, can be updated on a regular basis automatically.

The command contains a single argument *-jdbproperties*, which provides the path to a properties file.

```
set Operation=-file "Anaplan_Demo_Sql" -jdbcproperties
</path/to/jdbc.properties>
```

The *jdbc.properties* file contains the connection details including the path to the database, username, password, and the query string.

```
# JDBC Connection string (Oracle, Mysql, H2, etc.)
jdbc.connect.url= "jdbc:mysql://localhost:3306/apcustomer”

# JDBC login username
jdbc.username=user1

#JDBC login password
jdbc.password=dbpasswordforuser1

# (Optional) JDBC Fetch size, might not be applicable for all JDBC
flavors.
jdbc.fetch.size=10

# Boolean flag, needs to be true if provided {jdbc.query} is a
# stored-procedure/function call.
# Defaults to false if no value provided, or invalid.
jdbc.isStoredProcedure=false

# SQL query could be Select or a Stored-Procedure/Function call. In case
of latter,
# require {jdbc.isStoredProcedure} above set to "true".
# NOTE: SQL query needs to be parametrized with ? placeholders and values
```

```
for
# placeholders needs to be provided below in {jdbc.params}.
jdbc.query=” SELECT * FROM Anaplan where col1 = ? and col2 = ?”

# JDBC parameters for parametrized SQL query in {jdbc.query} above.
# NOTE: Number of parameters must equal to the number of parameters
provided in {jdbc.query} above.
jdbc.params= 1234,ABC
```

Where:

-jdbc.connect.url	-jdbcurl is the location of your database instance. This location does not have to be on the local host if the machine connecting to the database has access to the network location. To connect to an ODBC source (no longer supported in JRE 8+), you must configure your URL as follows: "*jdbc:odbc:location_of_instance*"
- jdbc.username	The database username. If no user name or password is required by the source, omit the *-jdbcuser* parameter.
- jdbc.password	The database password. If no user name or password is required by the source, omit the *-jdbcpassword* parameter.
- jdbc.fetch.size	Limits the number of rows that will be returned by the query.
- jdbc.isStoredProcedure	Boolean value that indicates whether the following statement (jdbc.query) is a stored procedure or function call.
-jdbcquery	-Specifies the query type -jdbcquery = "SELECT * FROM Anaplan where col1 = ? and col2 = ?" is the query to run.
-jdbc.params	JDBC parameters for parametrized SQL query in {jdbc.query} above. -jdbc.params = 1234,ABC

Appendix E: Import a JDBC Connection for a Microsoft SQL Server Database

An alternative to importing data into Anaplan from a file on the local host is to import data from a relational database using a Java Database Connectivity (JDBC) connection.

- The database can be any database that directly supports JDBC.
- Although you write Windows batch files (or Linux/Mac OS shell script files) for Anaplan Connect to run, Anaplan Connect itself is written in Java, and thus is well-suited for JDBC.
- Anaplan Connect supports using JDBC for Import into Anaplan, not for exporting from Anaplan into an external database. In effect, you run a query against the database, but you can't update the database.

Preparation

If you want to create a connection to a Microsoft SQL Server database:

1. Copy a .jar file from the Microsoft SQL Server database server or client tools directory. If using Microsoft SQL Server 2008, the file name is *sqljdbc41.jar*.
2. Paste the .jar file into the **lib** subfolder of the Anaplan Connect installation.
3. Make sure you have the following information:
   - UNC path to the Sql Server instance
   - Valid user name and password for the database login
   - A valid query to select the data you want brought into Anaplan.
     For example, SELECT * FROM MYTABLE

An example batch file for an Import action through JDBC:

```
@echo off
rem This example loads a source text file and runs an Anaplan import into a module.
rem For details of how to configure this script see doc\Anaplan Connect User Guide.doc
set AnaplanUser="Anaplan.User@anaplan.com:Password"
set WorkspaceId="8a8194884b27c72b014bf06a2b227f90"
set ModelId="CD9662D60CA84E9A871C1C5D061C7426"
set Operation=-file "Anaplan_Demo_Sql" -jdbcproperties "C:\My Source.txt"
>
rem *** End of settings - Do not edit below this line ***
setlocal enableextensions enabledelayedexpansion || exit /b 1
cd %~dp0
if not %AnaplanUser% == "" set Credentials=-user %AnaplanUser%
set Command=.\AnaplanClient.bat %Credentials% -workspace %WorkspaceId% -
model %ModelId% %Operation%
@echo %Command%
cmd /c %Command%
pause
set Operation=-file "Anaplan_Demo_Sql" -jdbcproperties
</path/to/jdbc.properties>
```

The jdbc.properties file contains the connection details including the path to the database, username, password, and the query string.

```
# JDBC Connection string (Oracle, Mysql, H2, etc.)
jdbc.connect.url= " jdbc:sqlserver://localhost”

# JDBC login username
jdbc.username=user1

#JDBC login password
jdbc.password=dbpasswordforuser1

# (Optional) JDBC Fetch size, might not be applicable for all JDBC
flavors.
jdbc.fetch.size=10

# Boolean flag, needs to be true if provided {jdbc.query} is a
# stored-procedure/function call.
# Defaults to false if no value provided, or invalid.
jdbc.isStoredProcedure=false

# SQL query could be Select or a Stored-Procedure/Function call. In case
of latter,
# require {jdbc.isStoredProcedure} above set to "true".
# NOTE: SQL query needs to be parametrized with ? placeholders and values
for
# placeholders needs to be provided below in {jdbc.params}.
jdbc.query=” SELECT * FROM Anaplan where col1 = ? and col2 = ?”
# JDBC parameters for parametrized SQL query in {jdbc.query} above.
# NOTE: Number of parameters must equal to the number of parameters
provided in {jdbc.query} above.
jdbc.params= 1234,ABC
```

where:

-file "Anaplan_Demo_Sql"	The action involves a file with the specified name. (A model-to-model import would not involve a file.) "Anaplan_Demo_Sql" represents the name of the new import file that will show up in Anaplan after the batch file has been run successfully.
-jdbcurl "jdbc:sqlserver://localhost"	Get data from a specified URL using a JDBC driver. • jdbc:sqlserver means JDBC is the type of connection, sqlserver means Microsoft SQL Server. • localhost means the local computer is acting as the test server. A production server would have a different name and path.
-jdbcuser "userName:Password"	• -jdbcuser specifies that the user is connecting through JDBC. • “userName:Password” represent the user name and password.
-jdbcquery "SELECT * FROM Anaplan_ Demo.dbo.Roster"	-jdbcquery specifies the query type. "SELECT * FROM Anaplan_Demo.Table" is the query to run.

Appendix F: Import through an ODBC Connection for an Oracle Database

Note: This functionality is only available to users on Java 7 or below. ODBC has been deprecated in Java 8.

Open Database Connectivity (OBDC) is similar to JDBC, but you can use ODBC for applications that are written in a language other than Java. To enable Anaplan Connect, a Java application, to connect to an Oracle database, the core of which is a C programming language application, you must reference an ODBC-JDBC bridge.

- We do not reference an ODBC-JDBC bridge for Microsoft SQL Server because SQL Server provides direct JDBC support.
- If you want to use Anaplan Connect to import from an Oracle database, note that Java 8 does not support the JDBC-ODBC Bridge (see https://blogs.oracle.com/Lance/entry/removal_of_the_jdbc_odbc).

Set the ODBC operation

```
set Operation=-file "Anaplan_Demo_ORA" -jdbcurl "jdbc:o
dbc:Anaplan_Demo" -jdbcuser "User:Password" -jdbcquery "Select * from
Anaplan"
@echo off
rem This example loads a source text file and runs an Anaplan import into
a module.
rem For details of how to configure this script, see doc\Anaplan Connect
User Guide.doc
set AnaplanUser="anaplan.user@anaplan.com:Password"
set WorkspaceId="8a8194884b27c72b014bf06a2b227f90"
set ModelId="CD9662D60CA84E9A871C1C5D061C7426"
set Operation=-file "Anaplan_Demo_ORA" -jdbcurl "jdbc:odbc:Anaplan_Demo" -
jdbcuser "User:Password" -jdbcquery "Select * from Anaplan"
rem *** End of settings - Do not edit below this line ***
setlocal enableextensions enabledelayedexpansion || exit /b 1
cd %~dp0
if not %AnaplanUser% == "" set Credentials=-user %AnaplanUser%
set Command=.\AnaplanClient.bat %Credentials% -workspace %WorkspaceId% -
model %ModelId% %Operation%
@echo %Command%
cmd /c %Command%
pause
```

where:

-file "Anaplan_Demo_ORA"	The action involves a file with the specified name. (A model-to-model import would not involve a file.) "Anaplan_Demo_ORA" represents the name of the new import file that will show up in Anaplan after the batch file has been run successfully.

-jdbcurl "jdbc:odbc:Anaplan_Demo"	Get data from a specified URL using a JDBC driver. • jdbc:obdc means the type of connection is an ODBC-JDBC bridge • Anaplan_Demo refers to the ODBC connection
-jdbcuser "User:Password"	-jdbcuser specifies that the user is connecting through JDBC, which is now possible through the ODBC-JDBC bridge. "User:Password" are credentials for the ODBC connection, which might be different from the credentials for the database or schema.
-jdbcquery "SELECT * FROM Anaplan	-jdbcquery specifies the query type "SELECT * FROM Anaplan" is the query to run