

# Building an FDM application for an HFM target

*Building an FDM application for the HFM adapter doesn't need to be a complicated process.*

## Background:

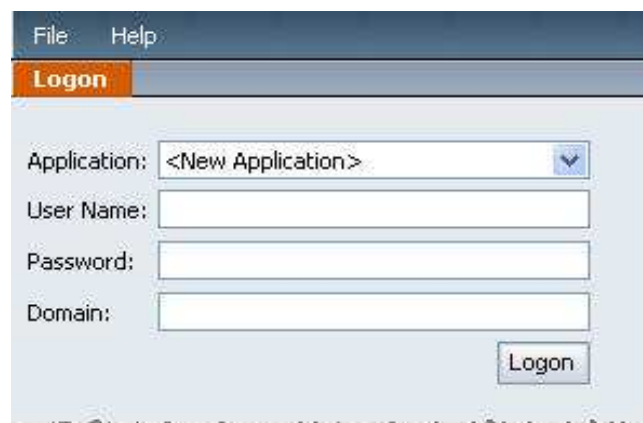
Using FDM to load G/L source data to HFM is becoming the leading tool for this process. Building the FDM application, however, can sometimes seem like a complicated process, though it doesn't need to be. Below is a checklist/workflow that can be used to ease the process of building an FDM application, connecting to an HFM adapter, and loading mapping templates.

## Building an FDM application:

This section shows the steps to create a new FDM application, connect to the HFM adapter and test the connection to the application.

- 1) Create an application - To create an application, log into the FDM Workbench using New Application. You will be prompted to enter a User name and Password.

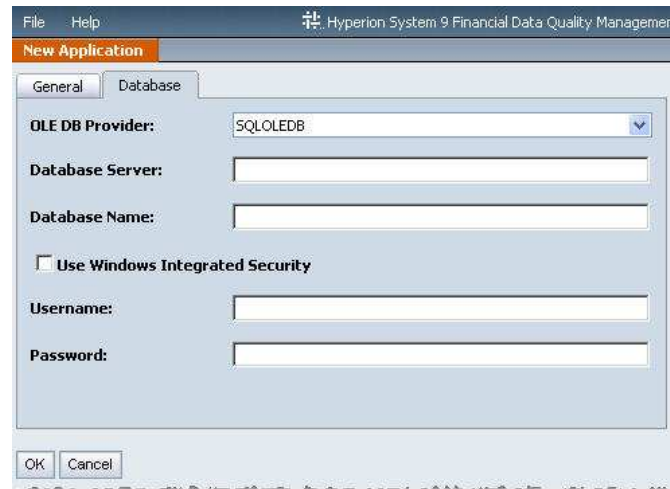
**Note: If you are using FDM 9.3.0 or later, you do not require a Domain name to be entered.**

A screenshot of a software dialog box titled "Logon". The dialog has a menu bar with "File" and "Help". Below the title bar, there is a section with a blue header labeled "Logon". The main area contains four input fields: "Application:" with a dropdown menu showing "<New Application>", "User Name:", "Password:", and "Domain:". A "Logon" button is located at the bottom right of the dialog.

- a. Click the General tab and enter the following:
  - Name: Name of FDM application
  - Description: Description for the application
  - Path: Location where the Inbox, Outbox, Template folders are created\*

\*On occasion, using a mapped drive location such as **f:\FDM** may cause data load errors as the SQL/HFM service ID cannot access a mapped drive. Therefore, using the UNC **Universal Naming Convention** name of a location such as: **\\clientfdmhfm05\FDM** is recommended.

- b. On the Database tab, enter the following:
  - OLE DB Provider: Name of DB provider to be used
  - Database Server : Name of DB server where the FDM database will be created
  - Database Name: The name of the FDM database to be created. This will also be the name of the Application folder to be created at the Location path specified in the General tab.
- c. Enter the database Username and Password to validate security for the DB. The ID must have administrative access to the DB server. This allows for application creation, in addition to writing to the FDM database tables.



- 2) Install the Adaptor:\*
  - a. File/Import Target System Adaptor
  - b. File/Register the Adaptor
  - c. Create Machine Profile – To create the Machine Profile:
    - i. Select the adapter in the Workbench
    - ii. Select Machine Profile
    - iii. Right-click to Add Machine Profile.  
For options enter the following:

- Source Machine is FDM Server
  - Target Machine is HFM Server or cluster
  - Leave user name and pw blank
- iv. Click OK to connect to FM application

\* There are two files when installing the adaptor.

- .xml – This contains all of the code.
- .dll – This needs to be registered on the machine that it's installed on.

**Note: When the Global Login option is selected, Upstream will always use this username and password to logon to the target application server / target application.**

3) Configure the Application:

a. Administration/Application Settings to confirm application settings

- i. Select Options: System Code
- ii. Ensure correct HFM Adapter is listed

b. Administration/ Integration Settings to Update Integration Settings

- i. Select Options: Application Name
- ii. Enter Target HFM Application Name

4) Test Target Application connection

a. Use drop-down box in Target Category in the C

- 9) Create Validation Entities
- 10) Assign validation Group and Import formats to locations.

### Loading Reports:

There are 80+ audit and system reports that are a part of the FDM application but they are not loaded by default. This next section outlines the steps to load these reports into your FDM application.

**Note: The reports may be in a compressed file (.zip). Zipped files must be extracted before importing into Hyperion FDM. Unzip the file into the <application name>\Templates folder.**

- 1) From the Hyperion FDM Workbench desktop activate the Reports screen by clicking the Reports tab.
- 2) Select File>Import
- 3) Locate and select the standard reports XML file applicable to the database that you are using (SQL or Oracle).
- 4) Click and the Import dialog will be displayed. The Import box contains two tabs -- General and Options
  - a. General - select the reports folder in the Import dialog
  - b. Options - contains two checkboxes.
    - i. The Save Existing Machine Profiles box enables/disables the machine profile override.
    - ii. The Remove All Report Groups checkbox enables/disables the Report Groups override. Checking this box will delete existing report groups before updating the application with new reports. Leave this box unchecked.
- 5) Click OK when done and the reports directory will appear on the Reports tab

### Importing Standard Templates:

Templates for loading mapping tables are provided with the FDM installation but not loaded into the application by default\*. Below are the steps required to make available the FDM templates to your application.

**Note: The standard template folder may be in a compressed file (.zip) named templates.zip. Extract this file into the <application name>\Outbox\Templates folder.**

\*Templates are not loaded into the application as the Reports are. Templates are copied into the templates folder and used at various intervals of the location and mapping setup process.

**Enable Dimensions and Change Dimension Labels:**

- 1) Select Admin Tasks>Dimension Setup>Dimensions
  - a. Edit the Custom dimensions you wish to use and place a check in the Enabled box.
  - b. Save Changes.
  
- 2) Select Admin Tasks>Dimension Setup>Dimension Labels
  - a. Chose the Dimensions you wish to change the labels for.

Examples: -Edit the Account Dimension and change the label to Source Account.

-Edit the UD1 dimension and change the label to Source Department.

-Save Changes.

**Populating the Excel Mapping Template with HFM dimensions:**

FDM provides maploader templates to easily create mappings in an excel spreadsheet. The mapping template also includes a macro script that pulls the HFM dimensions directly from the target HFM application you are connected to into the spreadsheet for easy selection. Below are the steps to populate the maploader template with HFM metadata.

- 1) Find the **Maploader.xml** template and place it in the ...Outbox\Templates folder
- 2) From the Workbench, create the new script ExportFMDimensions if it doesn't already exist. (Copy syntax below):

```
Sub ExportFMDimensions()  
'-----  
'UpStream WebLink Custom Script:  
,  
'Created By:    WLAdmin  
'Date Created: 10/22/2002 7:54:03 PM  
,  
'Purpose:      Export Hyperion Financial Management Dimensions  
,                                     to CSV File Format. These files will then be  
,                                     used to feed the standard Excel mapping template.  
,  
,  
,  
'-----  
'Declare local variables  
Dim objFileSys  
Dim objFile  
Dim objBP  
Dim strFile  
Dim lngCount
```

```

Dim varList

'Open the file system object
Set objFileSys = CreateObject("Scripting.FileSystemObject")

'Intiialize the block processor (Pass Active API & Script object References)
Set objBP = CreateObject("upsWBlockProcessorDM.clsBlockProcessor")
objBP.Initialize API, SCRIPTENG

'Call the Connect Action for the active integration block
Set API.IntBlockMgr.IntegrationMgr.PobjIntegrate = objBP.ActConnect("ExportDims")

'Check for runtime errors
If Err.Number <> 0 Then
    objR.intResult = False
    objR.strErrDesc = Err.Description
    objR.lngErrCode = Err.Number
    objR.blnIsErr = True
        'Destroy the object
        Set API.HyperionMgr.HPIIntegrationMgr.PobjIntegrate = Nothing
        Set objBP = Nothing
    Exit Sub
End If

'Check the connection and start checking intersections
If API.IntBlockMgr.IntegrationMgr.PobjIntegrate.intResult Then
    'Set the output file
    strFile = API.DataWindow.Connection.PstrDirOutbox & "\Templates\DimAccount.csv"
    Set objFile = objFileSys.CreateTextFile(strFile, True)
    'Execute method and assign Return value
    varList =
API.IntBlockMgr.IntegrationMgr.PobjIntegrate.varCon.DataAccess.fListAccounts(False)

        For lngCount = LBound(varList,2) To UBound(varList,2)
            objFile.WriteLine("***** & varList(0,lngCount) & """, "" & varList(1,lngCount) &
*****)
        Next
    objFile.Close

    'Set the output file
    strFile = API.DataWindow.Connection.PstrDirOutbox & "\Templates\DimEntity.csv"
    Set objFile = objFileSys.CreateTextFile(strFile, True)
    'Execute method and assign Return value
    varList =
API.IntBlockMgr.IntegrationMgr.PobjIntegrate.varCon.DataAccess.fListEntities(False)

        For lngCount = LBound(varList,2) To UBound(varList,2)
            objFile.WriteLine("***** & varList(0,lngCount) & """, "" & varList(1,lngCount) &
*****)
        Next
    objFile.Close

```

```

'Set the output file
strFile = API.DataWindow.Connection.PstrDirOutbox & "\\Templates\DimIcp.csv"
Set objFile = objFileSys.CreateTextFile(strFile, True)
'Execute method and assign Return value
varList =
API.IntBlockMgr.IntegrationMgr.PobjIntegrate.varCon.DataAccess.fListICPs()
  For lngCount = LBound(varList,2) To UBound(varList,2)
    objFile.WriteLine("***** & varList(0,lngCount) & ",***** & varList(1,lngCount) &
*****)
  Next
  objFile.Close

'Set the output file
strFile = API.DataWindow.Connection.PstrDirOutbox & "\\Templates\DimC1.csv"
Set objFile = objFileSys.CreateTextFile(strFile, True)
'Execute method and assign Return value
varList =
API.IntBlockMgr.IntegrationMgr.PobjIntegrate.varCon.DataAccess.fListCustom1()

  For lngCount = LBound(varList,2) To UBound(varList,2)
    objFile.WriteLine("***** & varList(0,lngCount) & ",***** & varList(1,lngCount) &
*****)
  Next
  objFile.Close

'Set the output file
strFile = API.DataWindow.Connection.PstrDirOutbox & "\\Templates\DimC2.csv"
Set objFile = objFileSys.CreateTextFile(strFile, True)
'Execute method and assign Return value
varList =
API.IntBlockMgr.IntegrationMgr.PobjIntegrate.varCon.DataAccess.fListCustom2()

  For lngCount = LBound(varList,2) To UBound(varList,2)
    objFile.WriteLine("***** & varList(0,lngCount) & ",***** & varList(1,lngCount) &
*****)
  Next
  objFile.Close

'Set the output file
strFile = API.DataWindow.Connection.PstrDirOutbox & "\\Templates\DimC3.csv"
Set objFile = objFileSys.CreateTextFile(strFile, True)
'Execute method and assign Return value
varList =
API.IntBlockMgr.IntegrationMgr.PobjIntegrate.varCon.DataAccess.fListCustom3()

  For lngCount = LBound(varList,2) To UBound(varList,2)
    objFile.WriteLine("***** & varList(0,lngCount) & ",***** & varList(1,lngCount) &
*****)
  Next
  objFile.Close

'Set the output file

```

```

strFile = API.DataWindow.Connection.PstrDirOutbox & "\Templates\DimC4.csv"
Set objFile = objFileSys.CreateTextFile(strFile, True)
'Execute method and assign Return value
varList =
API.IntBlockMgr.IntegrationMgr.PobjIntegrate.varCon.DataAccess.fListCustom4()

    For lngCount = LBound(varList,2) To UBound(varList,2)
        objFile.WriteLine("***** & varList(0,lngCount) & " & varList(1,lngCount) &
*****")
    Next
objFile.Close

'Destroy Objects
API.IntBlockMgr.IntegrationMgr.PobjIntegrate.varCon.Connection.fDisconnect
Set API.IntBlockMgr.IntegrationMgr.PobjIntegrate.varCon = Nothing

'Tell user we are done
If LCase(API.Datawindow.Connection.PstrClientType) = "WB.exe" Then
    MsgBox "Dimension Extract Complete!",vbexclamation,"WebLink"
Else
    RES.PlngActionType = 2
    RES.PstrActionValue = "Dimension Extract Complete!"
End If
Else
    If LCase(API.Datawindow.Connection.PstrClientType) = "WB.exe" Then
        MsgBox "Connection Error",vbexclamation,"WebLink"
    Else
        RES.PlngActionType = 2
        RES.PstrActionValue = "Connection Error"
    End If
End If

'Destroy objects
Set API.IntBlockMgr.IntegrationMgr.PobjIntegrate = Nothing
Set objBP = Nothing
Set objFile = Nothing
Set objFileSys = Nothing

End Sub

```

- 3) Select the script and run File/Run Script.
- 4) Check that a .csv file was created for each dimension in the Outbox\Templates folder
- 5) Open the **Maploader.xml** template (macro security must be set to Medium)
- 6) Select the Dimension lists tab, review.
- 7) Click the Setup tab and run Import Dimension CSV files.

- 8) Select the dimension tab again – dimension lists should be populated.
- 9) Select the main tab – ensure that selecting each dimension in the drop-down box in the header brings up a list of **base-level** members for that dimension in the target column.

#### **Mapping dimensions using load templates:**

- 1) Save the **maploader.xml** file several times – once for each dimension, i.e., **maploaderentity.xml**, **maploaderaccount.xml**, etc. For each file enter the Location ID and select the Dimension in the Header section.
- 2) For each file, select the appropriate dimension and begin mapping by selecting source and target members.

#### **Load Mapping templates:**

Once the mapping templates are complete, they are ready to be loaded into the FDM application.

- 1) Select Maps in the Web client
- 2) Browse Select File, navigate to the ...Outlook/Templates folder and select one of the **maploaderdimension.xml** files.
- 3) Select the appropriate dimension in the dimension drop-down list.
- 4) Select Import.
- 5) All mappings from the load file should have loaded.
- 6) Continue with all of the dimension files.

#### **Copy Mapping templates:**

Copying mapping templates from one Location to another is useful when G/L source systems have the same COA structure.

- 1) Set the POV to the Location to be copied.
- 2) Click the Copy button
- 3) Select the Target Location maps are to be copied to.

**Note: Only the active dimension conversion table is copied.**

### **Assigning Parent Maps:**

A location can be assigned a parent location to leverage the same mapping table as the parent. Multiple locations may have the same parent location.

- 1) Select the Location
- 2) Select the General tab
- 3) Select the location under Parent Location dropdown.

You have successfully created an FDM application with a target HFM application!



## Contact Information

For more information please contact:

Thomas Antunez, Principal  
Percentix, Inc.  
Irvine, CA  
Tel: +1.949.230.6400  
Email: [tantunez@percentix.com](mailto:tantunez@percentix.com)

Chris DeBeer, Managing Director  
Percentix, Inc.  
Englewood, CO  
Tel: +1.303.807.2238  
Email: [cdebeer@percentix.com](mailto:cdebeer@percentix.com)

Marisa Gregorio, Manager  
Percentix, Inc.  
Englewood, CO  
Tel: +1.303.957.7499  
Email: [mgregorio@percentix.com](mailto:mgregorio@percentix.com)

This publication contains general information only and Percentix, Inc. is not, by means of this publication, rendering business, financial, investment, or other professional advice or services. This publication is not a substitute for such professional advice or services, nor should it be used as a basis for any decision or action that may affect your business. Before making any decision or taking any action that may affect your business you should consult a qualified professional advisor. Percentix, Inc., its affiliates, and related entities shall not be responsible for any loss sustained by any person or entity who relies on this publication.

For more information about Percentix, Inc. please visit our website at: [www.percentix.com](http://www.percentix.com)