



How To Test Data Abstraction Best Practices

An Open Source Asset for use with TIBCO® Data Virtualization

TIBCO Software empowers executives, developers, and business users with Fast Data solutions that make the right data available in real time for faster answers, better decisions, and smarter action. Over the past 15 years, thousands of businesses across the globe have relied on TIBCO technology to integrate their applications and ecosystems, analyze their data, and create real-time solutions. Learn how TIBCO turns data—big or small—into differentiation at www.tibco.com.

Project Name	AS Assets Data Abstraction Best Practices
Document Location	This document is only valid on the day it was printed. The source of the document will be found in the ASAssets_DataAbstractionBestPractices folder (https://github.com/TIBCOSoftware)
Purpose	Self-paced instructional



www.tibco.com

Global Headquarters
3303 Hillview Avenue
Palo Alto, CA 94304

Tel: +1 650-846-1000
+1 800-420-8450
Fax: +1 650-846-1005

Revision History

Version	Date	Author	Comments
1.0	08/28/2013	Mike Tinius	Initial revision
8.0	12/05/2013	Mike Tinius	Updated for Best Practices 8.0
8.1	05/05/2014	Mike Tinius	Updated for Best Practices 8.1 and ASAssets change.
8.1.3	08/08/2014	Mike Tinius	Updated docs for Best Practices 8.1.3 release.
8.1.4	08/25/2014	Mike Tinius	Updated for Best Practices 8.1.4 release.
8.1.5	11/26/2014	Mike Tinius	Updated for Best Practices 8.1.5 release
8.1.6	05/20/2014	Mike Tinius	Updated for Best Practices 8.1.6 release
8.1.7	11/26/2014	Mike Tinius	Updated for Best Practices v8.1.7 – added generateViews=2 to allow generating views with a SELECT * projection.
8.1.8	05/24/2017	Mike Tinius	Updated for Best Practices v8.1.8 – added Privilege scripts.
8.1.9	12/06/2017	Mike Tinius	Transitioned to Tibco for release 8.1.9

Related Documents

Name	Version
How To Use Utilities.pdf	2017Q4
How To Use Data Abstraction Best Practices.pdf	8.1.9
How To Learn Data Abstraction Best Practices.pdf	8.1.9
How To Use Data Abstraction Best Practices Manage Annotations.pdf	8.1.9
How To Use Data Abstraction Best Practices Privilege Scripts.pdf	8.1.9

Supported Versions

Name	Version
TIBCO® Data Virtualization	7.0 or later
AS Assets Utilities open source	2017Q4 or later



Table of Contents

- 1 Overview 4
- 2 Project Maintenance Functional Tests 5
- 3 Upgrade Project Functional Tests 7
- 4 Generate Views Functional Test 12
- 5 Perform Round Trip Test 21
 - Test Round Trip for “lab00” 21
 - Test Round Trip for “BestPracticesTestSmall” 24

1 Overview

This document provides a test plan for the Data Abstraction Best Practices. The test plan should be followed whenever the Utilities and Best Practice code is changed. In the text below, any reference to “X_X_X” should be replaced with the current version of the Best Practices. The test project that will be used for testing will be called “lab00”. There are also references to the major and minor version XX. In this case the point release is dropped from the folder name. These releases should be considered minor point release upgrades to the existing Data Abstraction Best Practices folder. For example:

- /shared/BestPractices_v81
 - The same folder is used for 8.1.1, 8.1.2, 8.1.3, 8.1.4, 8.1.5, 8.1.6, and 8.1.7
 - Each point release will overwrite the previous release.
- /shared/BestPractices_v82 - this would represent a minor release change where the functionality was great enough to determine elicit a new folder while preserving the original folder.
 - This could be a result of structural changes to the code base or differences in the underlying Utilities parameters.

2 Project Maintenance Functional Tests

The project maintenance methods are found in the folder:

/shared/BestPractices_vXX/_ProjectMaintenance

Assumptions:

- The Data Abstraction Sample was imported during installation.
 - BestPractices_vX_X_X_AS_DataAbstractionSample81.car

The following are a list of functional tests that should be performed:

1. Get the Best Practices Version
 - **getBestPracticesVersion()**
 - **Result:** returns the current version. Example: 8.0
2. Update Impacted Resources
 - **updateImpactedBestPractices()**
 - **Result:** returns 1 for success
3. Generate Project
 - **generateProject**("/shared/lab00", 1, 1)
 - **Result:** Refresh Studio...generates the project specified on input.
4. Create data source metadata
 - Copy the entire folder of Metadata found at:
/shared/DataAbstractionSampleXX/Physical/ Metadata to
/shared/lab00/Physical
 - Copy the entire folder of Transformations found at:
/shared/DataAbstractionSampleXX/Physical/ Formatting/Transformations to
/shared/lab00/Physical/Formatting
 - Rebind the XSLT in Formatting/Transformations to test/Metadata...
5. Generate Configure Starting Folders
 - **generateConfigureStartingFolders**("/shared/lab00")
 - **Result:** Refresh Studio...generates a new ConfigureStartingFolders at
/shared/lab00/_scripts/Configure/ConfigureStartingFolders
6. Rename Project
 - **renameProject**("/shared/lab00", "lab00_new", 1, 1, 1)
 - **Result:** Refresh Studio...project has a new name of test_new

- **Verify /Constants/defaultValues** – basePath should be rebound to new name.
- **Verify /Documentation/documentationTrigger*** - parameters should be rebound to new name.
- **Verify /Generate/generate*** - all should be rebound to new name.

7. Move Project

- **moveProject**("/shared/lab00_new", "/shared/labs/test", 1, 1, 1)
- **Result:** Refresh Studio...project has been moved to a new folder
 - **Verify /Constants/defaultValues** – basePath should be rebound to new name.
 - **Verify /Documentation/documentationTrigger*** - parameters should be rebound to new name.

Verify /Generate/generate* - all should be rebound to new name.

3 Upgrade Project Functional Tests

The upgrade project maintenance method is found in the folder: **/shared/BestPractices_vXX/_ProjectMaintenance**. Import sample projects are previous versions of the Best Practices from the RegressionTest folder. The following are a list of functional tests that should be performed:

1. Upgrade version 1.0
 - a. Import (overwrite) sample project DataAbstractionSample10.car
 - b. Import (overwrite) B.P. 3.0 – BestPractices_generic_v3-2010-09-30.car
 - c. **upgradeProject**(X.X, "/shared/DataAbstractionSample10", 1)
 - d. **Result:** creates folder: /shared/DataAbstractionSample10_vXX
2. Upgrade version 4.0
 - a. Import (overwrite) sample project DataAbstractionSample40.car
 - b. Import (overwrite) B.P. 4.0 – BestPractices_v4-2011-02-08.car
 - c. **upgradeProject**(X.X, "/shared/DataAbstractionSample40", 1)
 - d. **Result:** creates folder: /shared/DataAbstractionSample40_vXX
 - e. **Known Issues:**
 - i. **/CRUD/isEmpty/isEmpty_SupportingDocuments**
 1. **isEmptyString**(inVector[1].FileContentBlob, explicit, result);
 - a. change to isEmptyBlob
 2. **isEmptyString**(inVector[1].FileContentClob, explicit, result);
 - a. change to isEmptyClob
 - ii. **/CRUD/RetrievePK/retrievePK_SupportingDocuments**
 1. **formatString**(inVector[i].FileContentBlob, 'FileContentBlob'...
 - a. comment out or remove this line
 2. **formatString**(inVector[i].FileContentClob, 'FileContentClob'...
 - a. change to formatClob
3. Upgrade version 5.0
 - a. Import (overwrite) sample project DataAbstractionSample50.car

- b. Import (overwrite) B.P. 5.0 – BestPractices_v5_1-2012_02_15.car
- c. **upgradeProject**(X.X, “/shared/DataAbstractionSample50”, 1)
- d. **Result:** creates folder: /shared/DataAbstractionSample50_vXX

4. Upgrade version 6.0

- a. Import (overwrite) sample project DataAbstractionSample40.car
- b. Import (overwrite) B.P. 6.0 – BestPractices_v6_0_repaired_ASAssets.car
- c. **upgradeProject**(X.X, “/shared/DataAbstractionSample60”, 1)
- d. **Result:** creates folder: /shared/DataAbstractionSample60_vXX
- e. **Known Issues:**
 - i. **/CRUD/isEmpty/isEmpty_SupportingDocuments**
 - 1. **isEmptyString**(inVector[1].FileContentBlob, explicit, result);
 - a. change to isEmptyBlob
 - 2. **isEmptyString**(inVector[1].FileContentClob, explicit, result);
 - a. change to isEmptyClob
 - ii. **/CRUD/RetrievePK/retrievePK_SupportingDocuments**
 - 1. **formatString**(inVector[i].FileContentBlob, 'FileContentBlob'...
 - a. comment out or remove this line
 - 2. **formatString**(inVector[i].FileContentClob, 'FileContentClob'...
 - a. change to formatClob

5. Upgrade version 6.6

- a. Import (overwrite) sample project DataAbstractionSample66.car
- b. Import (overwrite) B.P. 6.6 – BestPractices_v6_6_repaired_ASAssets.car
- c. **upgradeProject**(X.X, “/shared/DataAbstractionSample66”, 1)
- d. **Result:** creates folder: /shared/DataAbstractionSample66_vXX
- e. **Known Issues:**
 - i. **/CRUD/isEmpty/isEmpty_SupportingDocuments**
 - 1. **isEmptyString**(inVector[1].FileContentBlob, explicit, result);

- a. change to isEmptyBlob
 - 2. **isEmptyString**(inVector[1].FileContentClob, explicit, result);
 - a. change to isEmptyClob
 - ii. /CRUD/RetrievePK/retrievePK_SupportingDocuments
 - 1. **formatString**(inVector[i].FileContentBlob, 'FileContentBlob'...
 - a. comment out or remove this line
 - 2. **formatString**(inVector[i].FileContentClob, 'FileContentClob'...
 - a. change to formatClob
6. Upgrade version 7.0
 - a. Import (overwrite) sample project DataAbstractionSample70.car
 - b. Import (overwrite) B.P. 7.0 – BestPractices_v7_0_repaired_ASAssets.car
 - c. **upgradeProject**(X.X, "/shared/DataAbstractionSample70", 1)
 - d. **Result:** upgrades project in place: /shared/DataAbstractionSample70
 - e. **Known Issues:**
 - i. /CRUD/isEmpty/isEmpty_SupportingDocuments
 - 1. **isEmptyString**(inVector[1].FileContentBlob, explicit, result);
 - a. change to isEmptyBlob
 - 2. **isEmptyString**(inVector[1].FileContentClob, explicit, result);
 - a. change to isEmptyClob
 - ii. /CRUD/RetrievePK/retrievePK_SupportingDocuments
 - 1. **formatString**(inVector[i].FileContentBlob, 'FileContentBlob'...
 - a. comment out or remove this line
 - 2. **formatString**(inVector[i].FileContentClob, 'FileContentClob'...
 - a. change to formatClob
7. Upgrade version 7.1
 - a. Import (overwrite) sample project DataAbstractionSample71.car

- b. Import (overwrite) B.P. 7.1 – BestPractices_v7_1_repaired_ASAssets.car
- c. **upgradeProject**(X.X, “/shared/DataAbstractionSample71”, 1)
- d. **Result:** upgrades project in place: /shared/DataAbstractionSample71
- e. **Known Issues:**

- i. **/CRUD/isEmpty/isEmpty_SupportingDocuments**

- 1. **isEmptyString**(inVector[1].FileContentBlob, explicit, result);
 - a. change to isEmptyBlob
- 2. **isEmptyString**(inVector[1].FileContentClob, explicit, result);
 - a. change to isEmptyClob

- ii. **/CRUD/RetrievePK/retrievePK_SupportingDocuments**

- 1. **formatString**(inVector[i].FileContentBlob, 'FileContentBlob'...
 - a. comment out or remove this line
- 2. **formatString**(inVector[i].FileContentClob, 'FileContentClob'...
 - a. change to formatClob

8. Upgrade version 7.2

- a. Import (overwrite) sample project DataAbstractionSample72.car
- b. Import (overwrite) B.P. 7.2 – BestPractices_v7_2_repaired_ASAssets.car
- c. **upgradeProject**(X.X, “/shared/DataAbstractionSample72”, 1)
- d. **Result:** upgrades project in place: /shared/DataAbstractionSample72

9. Upgrade version 7.3

- a. Import (overwrite) sample project DataAbstractionSample73.car
- b. Import (overwrite) B.P. 7.3 – BestPractices_v7_3_repaired_ASAssets.car
- c. **upgradeProject**(X.X, “/shared/DataAbstractionSample73”, 1)
- d. **Result:** upgrades project in place: /shared/DataAbstractionSample73

10. Upgrade version 8.0

- a. Import (overwrite) sample project DataAbstractionSample80.car

- b. Import (overwrite) B.P. 8.0 –
BestPractices_v8_0_repaired_ASAssets.car
- c. **upgradeProject**(X.X, “/shared/DataAbstractionSample80”, 1)
 - i. **Result:** upgrades project in place:
/shared/DataAbstractionSample80

4 Generate Views Functional Test

The following are a list of functional tests that should be performed. Refer to the project created in the first section. Open the scripts generate folder:

/shared/labs/lab00/_scripts/Generate.

1. Copy Common_Model_v3_file3_lab.xlsxs to
C:\CompositeSoftware\BestPractices\BestPractices_vXX\Common_Model_v3_file3.xlsxs
2. Cache common_model view before starting. Be sure that Common_Model_v3_file3.xlsxs contains "lab00" project entries.
3. Generate Physical Views
 - **generatePhysicalViews**(0, 2, 1, 1, 1, null, null, null, null, ds_orders1,cisOraDemoDS.CISORADEMO)

Parameter	Value	Null
generateViewsWrapper	0	<input type="checkbox"/>
overwrite	2	<input type="checkbox"/>
copyAnnotation	1	<input type="checkbox"/>
copyPrivilegeMode	1	<input type="checkbox"/>
exactMatch	1	<input type="checkbox"/>
derivedFilterPath		<input type="checkbox"/>
excludeDsPathsList		<input checked="" type="checkbox"/>
sourceResource		<input checked="" type="checkbox"/>
generateToFolder		<input checked="" type="checkbox"/>
groupIds	ds_orders1,cisOraDemoDS.CISORADEMO	<input type="checkbox"/>

- **Result:** rows=152. Folders created:
 - /Physical/Physical/CISORADEMO
 - /Physical/Physical/ds_orders1
- 4. Generate Formatting Views
 - **generateFormattingViews**(0, 2, 1, 1, 1, null, null, null, null, null)

Parameter	Value	Null
generateViewsWrapper	0	<input type="checkbox"/>
overwrite	2	<input type="checkbox"/>
copyAnnotation	1	<input type="checkbox"/>
copyPrivilegeMode	1	<input type="checkbox"/>
exactMatch	1	<input type="checkbox"/>
derivedFilterPath		<input checked="" type="checkbox"/>
excludeDsPathsList		<input checked="" type="checkbox"/>
sourceResource		<input checked="" type="checkbox"/>
generateToFolder		<input checked="" type="checkbox"/>
groupIds		<input checked="" type="checkbox"/>

-
- **Result:** rows=455. All folders that match the folders in Metadata are created.

5. Generate Logical Views

- **generateLogicalViews**(0, 2, 1, 1, 1, null, null, null, null, ds_orders1,cisOraDemoDS.CISORADEMO)

Parameter	Value	Null
generateViewsWrapper	0	<input type="checkbox"/>
overwrite	2	<input type="checkbox"/>
copyAnnotation	1	<input type="checkbox"/>
copyPrivilegeMode	1	<input type="checkbox"/>
exactMatch	1	<input type="checkbox"/>
derivedFilterPath		<input checked="" type="checkbox"/>
excludeDsPathsList		<input checked="" type="checkbox"/>
sourceResource		<input checked="" type="checkbox"/>
generateToFolder		<input checked="" type="checkbox"/>
groupIds	ds_orders1,cisOraDemoDS.CISORADEMO	<input type="checkbox"/>

-
- **Result:** rows=153. Folders created:
 - /Business/Business/CISORADEMO
 - /Business/Business/ds_orders1

6. Generate Business Views

- **generateBusinessViews**(1, 2, 1, 1, 1, null, null, null, null, ds_orders1,cisOraDemoDS.CISORADEMO)

Parameter	Value	Null
generateViewsWrapper	1	<input type="checkbox"/>
overwrite	2	<input type="checkbox"/>
copyAnnotation	1	<input type="checkbox"/>
copyPrivilegeMode	1	<input type="checkbox"/>
exactMatch	1	<input type="checkbox"/>
derivedFilterPath		<input checked="" type="checkbox"/>
excludeDsPathsList		<input checked="" type="checkbox"/>
sourceResource		<input checked="" type="checkbox"/>
generateToFolder		<input checked="" type="checkbox"/>
groupIds	ds_orders1,cisOraDemoDS.CISORADEMO	<input type="checkbox"/>

-
- **Result:** rows=153. Folders created:
 - /Business/Business/CISORADEMO
 - /Business/Business/ds_orders1

7. Generate Client Views

- **generateClientViews**(1, 2, 1, 1, 1, null, null, null, null, ds_orders1,cisOraDemoDS.CISORADEMO)

Parameter	Value	Null
generateViewsWrapper	1	<input type="checkbox"/>
overwrite	2	<input type="checkbox"/>
copyAnnotation	1	<input type="checkbox"/>
copyPrivilegeMode	1	<input type="checkbox"/>
exactMatch	1	<input type="checkbox"/>
derivedFilterPath		<input checked="" type="checkbox"/>
excludeDsPathsList		<input checked="" type="checkbox"/>
sourceResource		<input checked="" type="checkbox"/>
generateToFolder		<input checked="" type="checkbox"/>
groupIds	ds_orders1,cisOraDemoDS.CISORADEMO	<input type="checkbox"/>

-
- **Result:** rows=153. Folders created:
 - /Application/Views/CISORADEMO
 - /Application/Views/ds_orders1

8. Generate Client Published Views

- **generateClientPublished**(0, 2, 1, 1, 1, null, null, null, null, ds_orders1,cisOraDemoDS.CISORADEMO)

Parameter	Value	Null
generateViewsWrapper	0	<input type="checkbox"/>
overwrite	2	<input type="checkbox"/>
copyAnnotation	1	<input type="checkbox"/>
copyPrivilegeMode	1	<input type="checkbox"/>
exactMatch	1	<input type="checkbox"/>
derivedFilterPath		<input checked="" type="checkbox"/>
excludeDsPathsList		<input checked="" type="checkbox"/>
sourceResource		<input checked="" type="checkbox"/>
generateToFolder		<input checked="" type="checkbox"/>
groupIds	ds_orders1,cisOraDemoDS,CISORADEMO	<input type="checkbox"/>

-
- **Result:** rows=153. Folders created:
 - /Application/Published/CISORADEMO
 - /Application/Published/ds_orders1

9. Generate Cast Views

- Delete the folder /Application/Published/ds_orders1
- **generateCastViews**(0, 2, 1, 1, 1, null,
 /shared/labs/lab00/Application/Views/ds_orders1
 /shared/labs/lab00/Application/Published/ds_orders1
)

Parameter	Value	Null
generateViewsWrapper	0	<input type="checkbox"/>
overwrite	2	<input type="checkbox"/>
copyAnnotation	1	<input type="checkbox"/>
copyPrivilegeMode	1	<input type="checkbox"/>
exactMatch		<input checked="" type="checkbox"/>
excludeDsPathsList		<input checked="" type="checkbox"/>
sourceResource	/shared/labs/lab00/Application/Views/ds_orders1	<input type="checkbox"/>
targetResource	/shared/labs/lab00/Application/Published/ds_orders1	<input type="checkbox"/>

- **Result:** rows=41. Folders created:
 - /Application/Published/ds_orders1

- Insure the views were created with cast statements.

10. Generate Published Database Resource

- Create a published database called “**ds_orders1**”
- **generatePublishedResource**(0, 2, 1, 1, 1, null, null, null, null, ds_orders1)

Parameter	Value	Null
generateViewsWrapper	0	<input type="checkbox"/>
overwrite	2	<input type="checkbox"/>
copyAnnotation	1	<input type="checkbox"/>
copyPrivilegeMode	1	<input type="checkbox"/>
exactMatch	1	<input type="checkbox"/>
derivedFilterPath		<input checked="" type="checkbox"/>
excludeDsPathsList		<input checked="" type="checkbox"/>
sourceResource		<input checked="" type="checkbox"/>
generateToFolder		<input checked="" type="checkbox"/>
groupIds	ds_orders1	<input type="checkbox"/>

OK Cancel

- **Result:** rows=4. Published resources created:
 - /services/databases/ds_orders1
 - /customers
 - /orderdetails
 - /orders
 - /shippingmethods

11. Generate Type Definitions

- **generateTypeDefinitions**(0, 2, 1, 1, null, myTypeDefs, null, null, CR, cisOraDemoDS.CISORADEMO)

Parameter	Value	Null
generateViewsWrapper	0	<input type="checkbox"/>
overwrite	2	<input type="checkbox"/>
copyPrivilegeMode	1	<input type="checkbox"/>
exactMatch	1	<input type="checkbox"/>
derivedFilterPath	1	<input type="checkbox"/>
typeDefProcName	myTypeDefs	<input type="checkbox"/>
sourceResource		<input checked="" type="checkbox"/>
generateToFolder		<input checked="" type="checkbox"/>
layerType	CR	<input type="checkbox"/>
groupIds	cisOraDemoDS.CISORADEMO	<input type="checkbox"/>

-
- **Result:** rows=1. Procedure created:
 - /Application/Services/CRUD/Definitions/
myTypeDefs_cisOraDemoDS.CISORADEMO

12. Generate CRUD Operations

- Pre-requisite: generatePhysicalViews must have been executed with the “generateIndexes=1” variable set.
- **generateCRUDOperations**(0, 2, 1, 1, null, null, null, null, null, CR, cisOraDemoDS.CISORADEMO)

Parameter	Value	Null
generateViewsWrapper	0	<input type="checkbox"/>
overwrite	2	<input type="checkbox"/>
copyPrivilegeMode	1	<input type="checkbox"/>
exactMatch	1	<input type="checkbox"/>
derivedFilterPath		<input checked="" type="checkbox"/>
typeDefProcName		<input checked="" type="checkbox"/>
sourceResource		<input checked="" type="checkbox"/>
generateToFolder		<input checked="" type="checkbox"/>
layerType	CR	<input type="checkbox"/>
groupIds	cisOraDemoDS.CISORADEMO	<input type="checkbox"/>

-
- **Result:** rows=103. Folders created:
 - /Application/Services/CRUD/...multiple folders

13. Generate Data Source List CSV File

- **generateDatasourceListCSV**(

c:\temp\file1.csv, 100, 1, 1, R, null, null, null, null, null, null, FV,
ds_orders1,cisOraDemoDS.CISORADEMO)

Parameter	Value	Null
csvFullPath	c:\temp\file1.csv	<input type="checkbox"/>
bufferSize	100	<input type="checkbox"/>
generateHeader	1	<input type="checkbox"/>
generateLogicalNames	1	<input type="checkbox"/>
generateMode	R	<input type="checkbox"/>
caseRule		<input checked="" type="checkbox"/>
useAliasRule		<input checked="" type="checkbox"/>
generateUnsupportedColumnType		<input checked="" type="checkbox"/>
exactMatch		<input checked="" type="checkbox"/>
derivedFilterPath		<input checked="" type="checkbox"/>
excludeDsPathsList		<input checked="" type="checkbox"/>
targetResource		<input checked="" type="checkbox"/>
layerType	FV	<input type="checkbox"/>
groupIds	ds_orders1,cisOraDemoDS.CISORADEMO	<input type="checkbox"/>

-
- **Result:** error=0. File is created:

- A Common Model Best Practices CSV file gets created in the file system at C:\Temp\file1.csv

14. Generate Data Source List

- **generateDatasourceList**(1, R, null, null, null, null, null, null, null, FV,
ds_orders1,cisOraDemoDS.CISORADEMO)

Parameter	Value	Null
generateLogicalNames	1	<input type="checkbox"/>
generateMode	R	<input type="checkbox"/>
caseRule		<input checked="" type="checkbox"/>
useAliasRule		<input checked="" type="checkbox"/>
generateUnsupportedColumnType		<input checked="" type="checkbox"/>
exactMatch		<input checked="" type="checkbox"/>
derivedFilterPath		<input checked="" type="checkbox"/>
excludeDsPathsList		<input checked="" type="checkbox"/>
targetResource		<input checked="" type="checkbox"/>
layerType	FV	<input type="checkbox"/>
groupIds	ds_orders 1,cisOraDemoDS.CISORADEMO	<input type="checkbox"/>

OK Cancel

○

○ **Result:** rows=153.

15. Generate “Generic” Views

- Create a custom copy of generateFormattingViews and place in the _Custom folder
- Modify the following parameter
 - DECLARE generateViews SMALLINT DEFAULT 2;
 - -- 0=Do not generate - (browse only) print out what will happen but don't perform the generation
 - -- 1=Do generate [DEFAULT] - Perform the VIEW Generation with a column projection.
 - -- 2=Do generate - Perform the VIEW Generation with a select * projection.
- **generateViews**(1, 2, 1, 1, 1, null, null, null, <S>, <T>, null)

S /shared/lab00/Physical/Formatting/Transformations/ds_XML/productCatalog_Transformation

T /shared/lab00/Physical/Formatting_All/Transformations/ds_XML/productCatalog_Transformation

S /shared/lab00/Physical/Metadata/Excel/Common_Model_v3_file2/Common_Model_v3_file2.xlsx

T

/shared/lab00/Physical/Formatting_All/Excel/Common_Model_v3_file2/Common_Model_v3_file2.xlsx

S /shared/lab00/Physical/Metadata/OracleSource/cisOraDemoDS/CISORADEMO

T /shared/lab00/Physical/Formatting_All/OracleSource/cisOraDemoDS/CISORADEMO

S /shared/BestPracticesTestSmall/Physical/Metadata/OracleSource/cisOraDemoDS/CISORADEMO

T /shared/BestPracticesTestSmall/Physical/Formatting_All/OracleSource/cisOraDemoDS/CISORADEM
 O
 S /shared/lab00/Physical/Metadata/File/testfile
 T /shared/lab00/Physical/Formatting_All/File/testfile

Parameter	Value	Null
generateViewsWrapper	0	<input type="checkbox"/>
overwrite	2	<input type="checkbox"/>
copyAnnotation	1	<input type="checkbox"/>
copyPrivilegeMode	1	<input type="checkbox"/>
exactMatch		<input checked="" type="checkbox"/>
derivedFilterPath		<input checked="" type="checkbox"/>
excludeDsPathsList		<input checked="" type="checkbox"/>
sourceResource	/shared/lab00/Physical/Metadata/File/testfile	<input type="checkbox"/>
generateToFolder	/shared/lab00/Physical/Formatting_All/File/testfile	<input type="checkbox"/>
groupIds		<input checked="" type="checkbox"/>

OK Cancel

- **Result:** Views generated with SELECT * projection in the /Formatting_All folder.

5 Perform Round Trip Test

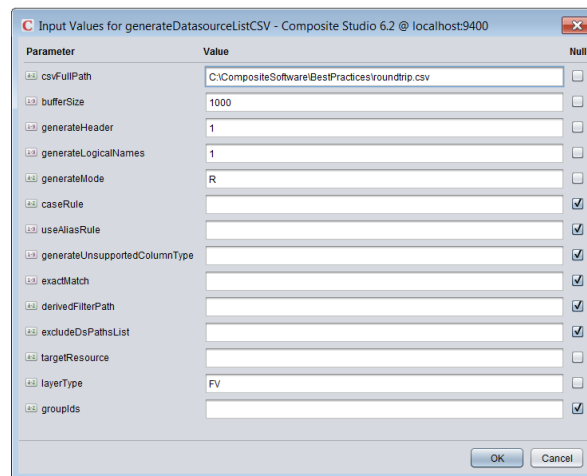
The following is a procedure to follow to execute a round trip test. This test will use the both the “lab00” folder previously created as well as the much larger test base from the “BestPracticesTestSmall” project. The objective of this test is to make sure the code is consistent between generating views and generating the CSV spreadsheet.

Test Round Trip for “lab00”

1. Assumption: The cache has been refreshed.
2. Remove /shared/labs/lab00/Physical/Formatting/<all-subfolders-except-Transformations>
3. Generate Formatting Views
 - **generateFormattingViews**(0, 2, 1, 1, 1, null, null, null, null, null)

Parameter	Value	Null
generateViewsWrapper	0	<input type="checkbox"/>
overwrite	2	<input type="checkbox"/>
copyAnnotation	1	<input type="checkbox"/>
copyPrivilegeMode	1	<input type="checkbox"/>
exactMatch		<input checked="" type="checkbox"/>
derivedFilterPath		<input checked="" type="checkbox"/>
excludeDsPathsList		<input checked="" type="checkbox"/>
sourceResource		<input checked="" type="checkbox"/>
generateToFolder		<input checked="" type="checkbox"/>
groupIds		<input checked="" type="checkbox"/>

4. Generate Data Source List CSV File
 - **generateDatasourceListCSV**(
C:\CompositeSoftware\BestPractices\roundtrip.csv, 1000, 1, 1, R, null, null,
null, null, null, null, null, FV, null)



- **Result:** error=0. File is created:
 - A Common Model Best Practices CSV file gets created in the file system at C:\CompositeSoftware\BestPractices\roundtrip.csv
- 5. Make a copy of the Formatting layer views
Target=/shared/labs/lab00/Physical/Formatting_Copy_1
- 6. Copy the contents *roundtrip.csv* to *Common_Model_v3_file4_sample_lab.xlsx*
 - Open *Common_Model_v3_file4_sample_lab.xlsx*
 - Open *roundtrip.csv* in Excel
 - Select cell A2 and use the keystrokes: Ctrl-Shift-End (this will select Columns A-L and all rows except the header row)
 - Do a Ctrl-C (copy)
 - Switch to *Common_Model_v3_file4_sample_lab.xlsx*
 - Locate the cells for "lab00"
 - Place your cursor in the first cell for lab00
 - Do a Ctrl-V (Paste)
 - Save the spreadsheet
 - Refresh the cache of the spreadsheets: *Common_Model_v3_file[1-4].xlsx*
 - Switch to Studio, Manager tab
 - Select "Cached Resources"
 - Select "common_model"
 - Click Refresh Cache
- 7. Remove the views in the Formatting layer except the "/Transformations" folder which is considered a source folder for transformation procedures.

8. Generate the Formatting layer views
generateFormattingViews(1, 2, 1, 0, null, null, null, null, null, null)
9. Validate the view generation process
 - validateGenerateViews(
/shared/labs/lab00/Physical/Formatting,
/shared/labs/lab00/Physical/Formatting_Copy_1, null, 1, N, C, N)
 - **Result: There should be no rows displayed as this indicates that all views are equal and the round-trip was successful.**

Test Round Trip for “BestPracticesTestSmall”

1. Import BestPracticesTestSmall.car.
2. Assumption: The cache has been refreshed.
3. Remove /shared/BestPracticesTestSmall/Physical/Formatting/<all-subfolders-except-Transformations>
4. Generate Formatting Views
 - **generateFormattingViews**(0, 2, 1, 1, 1, null, null, null, null, null)

Parameter	Value	Null
generateViewsWrapper	0	<input type="checkbox"/>
overwrite	2	<input type="checkbox"/>
copyAnnotation	1	<input type="checkbox"/>
copyPrivilegeMode	1	<input type="checkbox"/>
exactMatch		<input checked="" type="checkbox"/>
derivedFilterPath		<input checked="" type="checkbox"/>
excludeDsPathsList		<input checked="" type="checkbox"/>
sourceResource		<input checked="" type="checkbox"/>
generateToFolder		<input checked="" type="checkbox"/>
groupIds		<input checked="" type="checkbox"/>

5. Generate Data Source List CSV File

- **generateDatasourceListCSV**(
C:\CompositeSoftware\BestPractices\roundtrip.csv, 1000, 1, 1, R, null, null,
null, null, null, null, null, FV, null)

Parameter	Value	Null
csvFullPath	C:\CompositeSoftware\BestPractices\roundtrip.csv	<input type="checkbox"/>
bufferSize	1000	<input type="checkbox"/>
generateHeader	1	<input type="checkbox"/>
generateLogicalNames	1	<input type="checkbox"/>
generateMode	R	<input type="checkbox"/>
caseRule		<input checked="" type="checkbox"/>
useAliasRule		<input checked="" type="checkbox"/>
generateUnsupportedColumnType		<input checked="" type="checkbox"/>
exactMatch		<input checked="" type="checkbox"/>
derivedFilterPath		<input checked="" type="checkbox"/>
excludeDsPathsList		<input checked="" type="checkbox"/>
targetResource		<input type="checkbox"/>
layerType	FV	<input type="checkbox"/>
groupIds		<input checked="" type="checkbox"/>

- **Result:** error=0. File is created:
 - A Common Model Best Practices CSV file gets created in the file system at C:\CompositeSoftware\BestPractices\roundtrip.csv

6. Make a copy of the Formatting layer views

- Target=/shared/ BestPracticesTestSmall/Physical/Formatting_Copy_1
7. Copy the contents *roundtrip.csv* to *Common_Model_v3_file3.xlsx*
- Open *Common_Model_v3_file3.xlsx*
 - Open *roundtrip.csv* in Excel
 - Select cell A2 and use the keystrokes: Ctrl-Shift-End (this will select Columns A-L and all rows except the header row)
 - Do a Ctrl-C (copy)
 - Switch to *Common_Model_v3_file3.xlsx*
 - Remove all cells with data for columns A-L
 - Place cursor in cell A2
 - Do a Ctrl-V (Paste)
 - Save the spreadsheet
 - Refresh the cache of the spreadsheets: *Common_Model_v3_file[1-4].xlsx*
 - Switch to Studio, Manager tab
 - Select "Cached Resources"
 - Select "common_model"
 - Click Refresh Cache
8. Remove the views in the Formatting layer except the "/Transformations" folder which is considered a source folder for transformation procedures.
9. Generate the Formatting layer views
`generateFormattingViews(1, 2, 1, 0, null, null, null, null, null, null)`
10. Validate the view generation process
`validateGenerateViews(/shared/BestPracticesTestSmall/Physical/Formatting_Copy_1, null, 1, N, C, N)`
- **Result: There should be no rows displayed as this indicates that all views are equal and the round-trip was successful.**