How To Use Data Abstraction Best Practices Manage Annotations

An Open Source Asset for use with TIBCO® Data Virtualization

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

Revision History

Version	Date	Author	Comments
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
2018Q1	03/20/2018	Mike Tinius	Release 2018Q1 – no changes.

Related Documents

Name	Version
How To Use Utilities.pdf	2018Q1
How To Use Data Abstraction Best Practices.pdf	2018Q1
How To Learn Data Abstraction Best Practices.pdf	2018Q1
How To Test Data Abstraction Best Practices.pdf	2018Q1
How To Use Data Abstraction Best Practices Privilege Scripts.pdf	2018Q1

Supported Versions

Name	Version
TIBCO® Data Virtualization	7.0 or later
AS Assets Utilities open source	2018Q1 or later

Table of Contents

1   Introduction

Purpose

The purpose of Best Practices Manage Annotation Scripts is to provide a framework for performing mass-updates on resource annotations based on rows maintained in a spreadsheet or a database.

The Manage Annotation Scripts help the developer to maintain and execute annotation updates during deployment for the various layers. The proper place for annotations to be set (especially column annotations) is at the immediate layer below the published layer. According to the best practices this may be either the Application Layer Published or Application Layer Views. The reason for this is that Data Virtualization (DV) Business Directory reads annotations from the layer of views just below the published. In fact, there is no way to set COLUMN annotations on published resources. The red highlighted boxed indicate where annotations are set depending on which layer is directly published to the DV Published Database Layer.
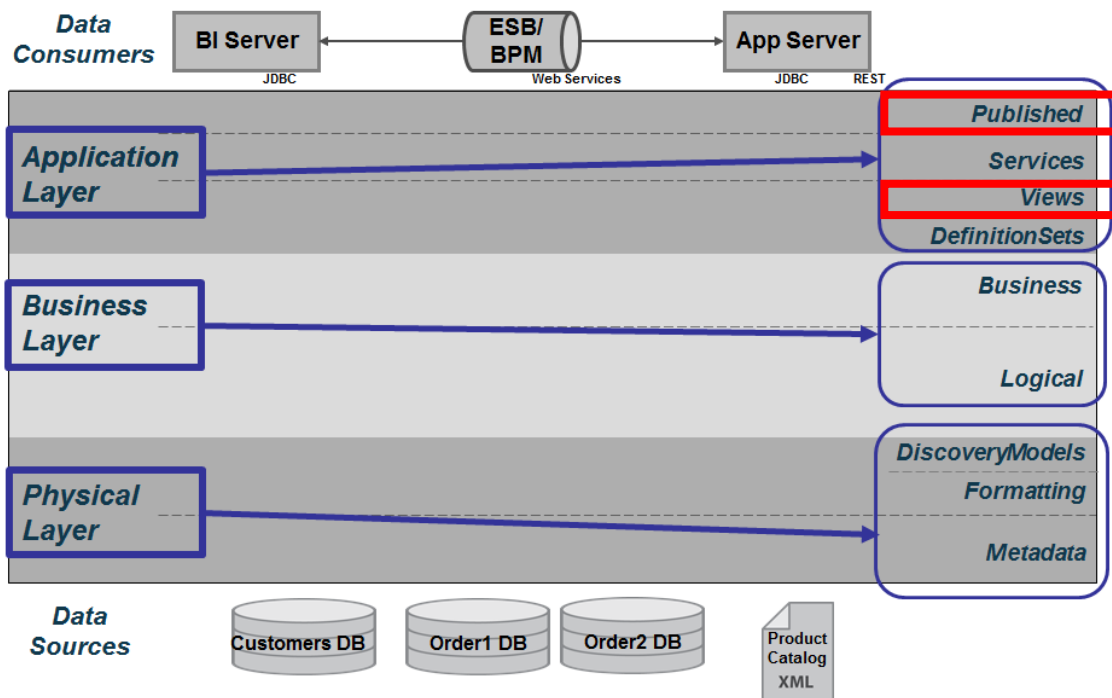
Figure one: Technical Data Abstraction Layers

Audience

This document is intended to provide guidance for the following users:

- Data Virtualization Administrators – provides a guide for installation.

- Architects – provides the data abstraction architecture.

- Data professionals – provides background on the published views and usage.

- Operations users – provides insight into triggers and procedures that are executed.

- Project Managers – provides general information on data abstraction best practices.

References

Product references are shown below. Any references to CIS or DV refer to the current TIBCO® Data Virtualization.

- TIBCO® Data Virtualization was formerly known as

  o Cisco Data Virtualization (DV)

  o Composite Information Server (CIS)

Pre-Requisites

Follow the steps below to create a new project.

1. LDAP groups have been brought into the target environment.
   /shared/ASAssets/Utilities have been installed and configured for 2018Q1.

2   Configuration

How to Configure

This section provides information on how to configure the Manage Annotation Scripts.

Manage Annotation Script Configuration Summary

1. Copy annotation spreadsheet to file system

2. Modify datasource connections

3. Reintrospect datasources

4. Test datasources

Best Practices Installation

1. Install Best Practices Spreadsheets

    1.1. Follow the Data Abstraction Best Practices installation directions found in this document:
    "**How To Use AS Data Abstraction Best Practices.pdf"**

2. Copy annotation spreadsheet to file system

    2.1. This should have been done during installation of the Best Practices. Verify this.

3. Modify the data source path or connection information:

    3.1. EXCEL: modify the root path if needed.

        3.1.1.DV Location: [EXCEL]
               /shared/ASAssets/BestPractices_v81/ManageAnnotations/Metadata/ManageAnnotations_EXCEL

        3.1.2.E.g. root path: C:/CIS7.0/BestPractices/Annotations

4. Reintrospect the "ManageAnnotations_EXCEL" data source

5. Test the data source

    5.1. Show Contents for the "ResourceAnnotations" worksheet to ensure that data is being
         retrieved.

6. Generate the Resource Annotation spreadsheet using the script
   /shared/ASAssets/BestPractices_v81/ManageAnnotations/generateResourceListToCSV

7. Modify the annotations as needed

    Place the values back into the original spreadsheet for use during "updateAnnotations".

3   Executing Resource Annotations

Introduction

This section provides guidance on how to manage the process for generating and updating DV resource annotations.

Instructions

1. The basic flow looks like this

    1.1. Generate the Resource Annotation spreadsheet using the script
        /shared/ASAssets/BestPractices_v81/ManageAnnotations/generateResourceListToCSV

        1.1.1.   Resources get generated onto the DV server. You will need a download user folder and the ability to FTP the file to your local machine.

        1.1.2.   Generate Resource Annotation List to CSV
            1.1.2.1.   Sheet: ResourceAnnotations
            1.1.2.2.   Script: **generateResourceListToCSV**
                1.1.2.2.1.   *startingPath*:   /shared/lab00/Application/Views/ds_orders1
                1.1.2.2.2.   *csvFilePath:*   C:\BestPractices\Annotations\note.csv

    1.2. Modify the annotations as needed

        1.2.1.   Convert the .csv to .xlsx as it makes it easier to edit

    1.3. Place the values back into the original spreadsheet for use during "updateAnnotations".

        1.3.1.   Either rename the file you downloaded or copy the values back to the original ResourceAnnotations.xlsx spreadsheet.

    1.4. Upload the file back to the DV server to replace the existing one.

    1.5. Test the data source

        1.5.1.   Show Contents for the "ResourceAnnotations" worksheet to ensure that data is being retrieved.

    1.6. Update Resource Annotations
        1.6.1.   Sheet: ResourceAnnotations
        1.6.2.   Script: **updateAnnotations**
            1.6.2.1.   *performUpdate*:   0 or 1.
            1.6.2.2.   The next parameter "*inSqlStatement*" provides the user with a way to inject there own datasoure rather than using the default
            1.6.2.3.   *inSqlStatement*:   select PhysicalName, PhysicalType, ResourcePath, ResourceAnnotation from

/shared/ASAssets/BestPractices_v81/ManageAnnotations/Formatting/ResourceAnnotati
ons

ResourceAnnotation Spreadsheet Definition

2. The basic flow looks like this

There are several columns that get generated that provide good informational value. However, there are only four columns that matter when performing the "updateAnnotations" and those are as follows:

PhysicalName, PhysicalType, ResourcePath, ResourceAnnotation

This chart describes the columns in the ResourceAnnotations spreadsheet:

Column Name	Description
DataSourceName	The data source name that the ResourcePath maps to if applicable.
DataSourceType	The data source type that the ResourcePath maps to if applicable.
DataSourceHostname	The data source hostname that the ResourcePath maps to if applicable.
PhysicalName	The name of the DV resource.
PhysicalType	The type of the DV resource. E.g. [TABLE, COLUMN]
ResourcePath	The full path of the DV resource. Paths are not double-quoted if they contain spaces or special characters.
ResourceAnnotation	The mult-line annotation of the DV resource.
ColumnDependencies	The column dependency lineage.

4 Manage Annotation Scripts Method Definitions

Detailed Definitions

Detailed documentation on the inputs and outputs can be found in the header and annotation section of each procedure.

Annotation Maintenance: Generate Annotations

1. **generateResourceListToCSV –** This procedure is used to generate the table and column annotations to a CSV file on the DV server. The user is responsible for downloading the file to their local workstation to work on the annotation list.

    1.1. Location:
        /shared/ASAssets/BestPractices_v81/ManageAnnotations/generateResourceListToCSV

Direction	Parameter Name	Parameter Type
IN	**startingPath** – path to folder that is one level below the published data service folder. Note: the startingPath may be a parent path as the procedure. This procedure recursively drills into child folder paths to find views. The reason for one level below the published is because this is where Business Directory gets the column annotations.	LONGVARCHAR
IN	**csvFilePath** – The output file path on the DV server. The user is responsible for downloading the file to their workstation for editing.	LONGVARCHAR

Annotation Maintenance: Update Annotations

2. **updateAnnotations –** This script is used to update resource annotations. The main objective is to update the resources and columns at the level directly below the published datasource layer in DV. This is the layer in which table columns may be updated. This is the layer in which Business Directory will read in the column metadata.

    2.1. RULES:

        2.1.1. If the annotation in the spreadsheet is blank with no spaces, it is considered a null indicating take no action. The update of the annotation for that resource is not performed.

        2.1.2. If the annotation in the spreadsheet contains 1 or more blanks, it is considered a modification and the update of the annotation will be performed and will blank out any existing annotation.

2.1.3. If the annotation in the spreadsheet contains non-space characters, it is considered a modification and the update of the annotation will be performed modifying the existing annotation.

2.1.4. If the annotation in the spreadsheet contains multiple lines then all lines will be taken as is and the resource annotation will retain those lines.

2.1.5. If the PhysicalType = 'TABLE' then the table annotation will be modified otherwise it is assumed to be a COLUMN annotation.

2.2. Location: /shared/ASAssets/BestPractices_v81/ManageAnnotations/updateAnnotations

Direction	Parameter Name	Parameter Type
IN	**performUpdate** – 0=do not perform update, 1=perform update environments. Basically it is used for testing purposes and developing new functionality.	INTEGER
IN	**inSqlStatement** – The table is provided by the user but the columns must match exactly as shown. This strategy allows the user to store there annotations in their own datasource [excel or database] in order to manage their annotations. Default Select if inSqlStatement is left null: select **PhysicalName, PhysicalType, ResourcePath, ResourceAnnotation** from /shared/ASAssets/BestPractices_v81/ManageAnnotations/Formatting/ResourceAnnotations	LONGVARCHAR
OUT	**rowsProcessed** – number of rows processed from the spreadsheet	INTEGER