



Data Relationship Management Integration with Planning and HPCM for EPMA Customers

ORACLE WHITEPAPER | OCTOBER 2019





Contents

Disclaimer and Safe Harbor	3
Who This Document is Intended For	3
Conventions	3
Copyright © and Trademarks ®™	3
Overview	4
Landscape and Workflow	5
Staging Outlines from EPMA and Planning in DRM	7
DRM Considerations for Preparing Outline Files	9
DRM Metadata Migration Templates.....	9
Preparing an Import file for DRM.....	10
Specific example of a DRM Import File	11
Preparing an ADS Generated File	15
Exporting Outlines to Planning	19
Import File in Planning.....	26
Hyperion Profitability & Cost Management Considerations.....	28
Automation Strategies	29
Command Syntax – DRM Batch Client	30
Command Syntax – Planning OLU	32
Automation Summary	33
Moving to EPM 11.2.x – Initial Release	34
Considerations	34
Parallel Environments	34
Migrating 11.1.2.4.x Applications to 11.2.x – Initial Release.....	34
Migrating an EPMA Outline Managed Planning Application to 11.2.x – Initial Release	34



Disclaimer and Safe Harbor

The following is intended to outline our general product usage. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Who This Document is Intended For

The following is intended to update administrators of EPMA and Planning Applications who possess experience with existing integrations between both products. These users are generally familiar with some of the verbiage used later in the document such as "Hierarchy", "Member", "Node", "Local Property", and "Relationship Property"; all used in the context of the application usage. This document is not intended as a tutorial for those who do not already administer or set up interactions between these products.

Conventions

Note also that this document employs additional terminology specific to DRM and/or Planning. Over the years some terminology has developed across user bases between the products. For example, DRM users typically refer to the structure content as "Nodes", and then Limb Nodes (can have children) or Leaf nodes (may not have children). Planning user tend to refer to the structural elements as "Members" with there being Parent Members (can have children) and Base Members (may not have children)" The terms are equivalent but can make for an interesting evening's discussion.

In many cases this is indicated by the use of a leading upper case letter. So the use of "Hierarchy" is intended to focus thinking on a DRM Hierarchy where "hierarchy" is used when referring to any generic hierarchical structure. Also at times, use of a leading upper case character is used to draw attention between information on a screenshot and the text associated with it in the explanation.

Copyright © and Trademarks ®™

All products mentioned in this document are either Copyright © and/or registered Trade Marks ®™ of Oracle Corporation and/or its Affiliates. All rights reserved; unless otherwise noted.

Overview

Customers have been using the freely licensed Oracle Hyperion Enterprise Performance Management Architect (EPMA) for integrating hierarchy member, member relationship structure and member property values (collectively termed, metadata) with other Oracle Hyperion products such as Oracle Hyperion Planning (Planning), Oracle Hyperion Profitability and Cost Management (HPCM), Oracle Hyperion Financial Management (HFM) and with the broader Oracle Hyperion Enterprise Performance Management (EPM) product base.

With the deprecation of EPMA, the question arises as to what other methods and tools are available that assist with integrations for metadata updates be performed across products and managed moving forward. One answer is for customers to use Oracle Data Relationship Management. Oracle Enterprise Data Relationship Management (DRM) is an enterprise change management solution for building and retaining consistency within master data assets despite the continuing changes necessary to support underlying transactional and analytical systems.

DRM is used by hundreds of customers across areas such as the corporate Financial Management area, Product Management and Human Resource. In addition to its full-featured Hierarchy structure management approach, member and property management features, such as formula based derivation of property values, it also provides for other optional features such as Analytical Metadata Management (Analytics) overview and Change Management Workflows through the Data Relationship Governance (DRG) functionality allowing governance oversight to be managed in the change process.

A utility is provided via My Oracle Support to assist customers with converting EPMA generated “.ADS” files for their Planning, HFM, HPCM and Essbase outlines to a DRM import format. This can be acquired from My Oracle Support as Patch Number 3069700: EPMA ADS FILE CONVERSION UTILITY FOR EPM 11.2

(<https://support.oracle.com/epmos/faces/PatchDetail?patchId=30695700>)

More details on DRM can be found at this [site](#)

(<https://www.oracle.com/technetwork/middleware/data-relationship-management/overview/index.html>).

Specifically, this document focuses on starting to use DRM as a substitute for EPMA. We'll do this by discussing the parts and pieces of a current EPMA to Planning landscape, transitioning initial content from EPMA to DRM and covering several approaches to transfer the metadata information from DRM to Planning.

Landscape and Workflow

The use of EPMA with products such as Planning generally takes one of several forms and there are at times permutations on any general approach being used. Below we outline a fairly typical case where the Planning system has been configured to use EPMA. This can be if it has a Shared Library and optionally there may be one or more ‘Local’ dimensions in any one application.

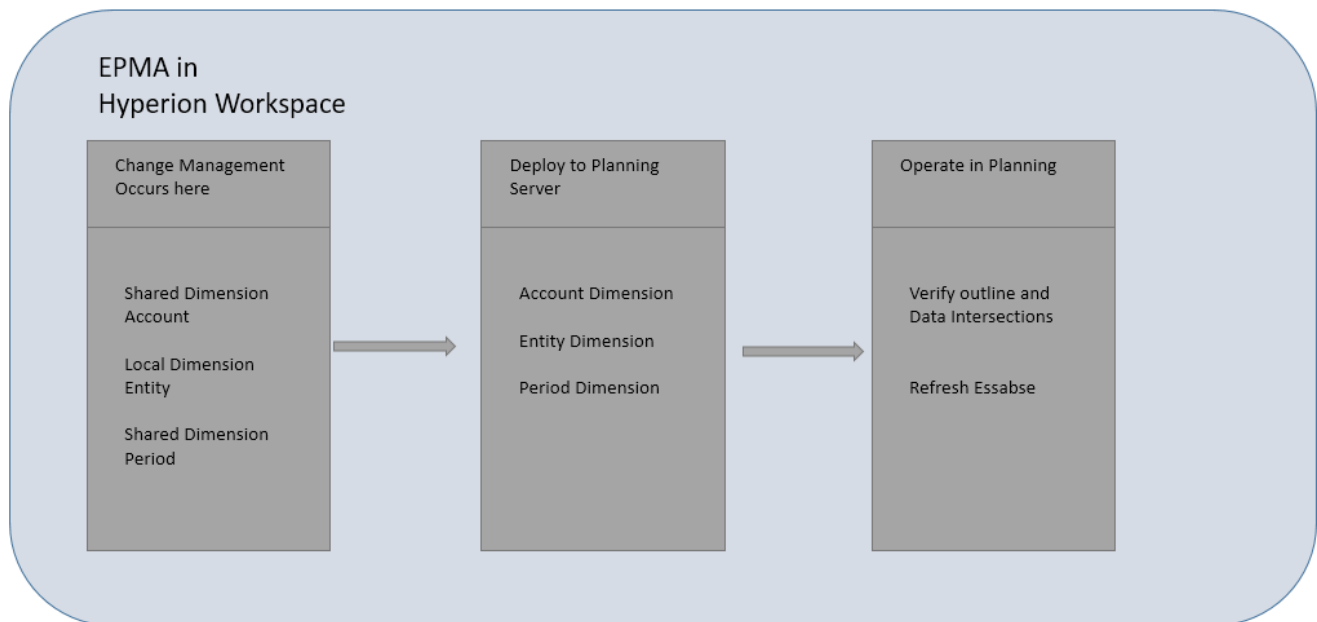
Generally, users manage outlines for Planning in EPMA by using

Navigate → Administer → Application Library

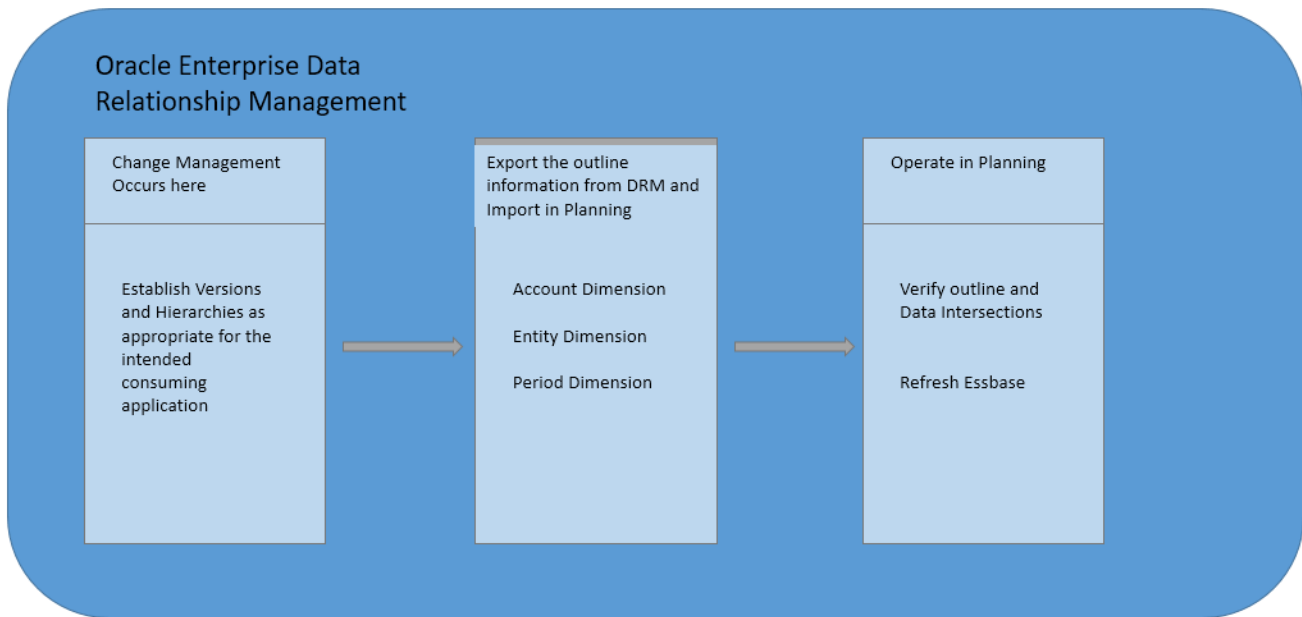
where they select an EPMA managed application and make desired edits to the dimension structures such as adding and moving members, managing SmartLists and updating property values. As all changes are finalized, changes can be deployed to an application such as Planning. This may be being done through managing a Shared Library in EPMA, Local Dimensions in the workspace or strategies that use a mix of Shared and Local Dimensions.

Going forward, EPMA customers will be able to use DRM to make these changes and submit an outline file directly to Planning. DRM allows for all these scenarios.

Presented graphically, an existing EPMA to Planning workflow may look like:




A DRM to Planning workflow may look like:



As you can see, they are virtually identical. While each of the applications has its specific features, DRM is generally considered much more rich in outline (metadata) management features providing for:

- A full featured Drag and Drop outline editor.
- In addition to standard Defined, List and Look-up properties and properties with Inheritance; DRM also provides Calculated (Derived) and Calculated Read-Write (Overrideable) Derived properties. For the latter case, not only can a value for a property of a member be calculated, it may be overridden provided the user is given sufficient access rights, in those cases where it is needed.
- Properties can be set up as applicable at the Version, Hierarchy and node levels.
- Several types of Validations where most, depending on type, can be configured to run Real-Time as users interact with the product and/or as part of a batched process where you may only want to run the checks after a several edits have been made.
- A robust formula system that is provided for as both a DRM-centric formula based property deriver system for the full DRM object model.
- A JavaScript extension to the formula systems also allows for formulas to be created with interaction to the DRM object model.
- Application templates for analytical apps such as Planning and HPCM, reporting templates for BI Apps and templates for interaction with FDMEE.
- In addition, DRM offers a Query capabilities for the metadata, so, for one small example, you can find all members where a specific value has been set.



Other DRM metadata management features include:

- The ability to compare two Hierarchies (perhaps a Primary and an Alternate) or compare the same Hierarchy across two separate Versions, for example, to compare similarities or differences between a May and June Version.
- A Blend capability that allows large scale merging of information across Version or Hierarchies.
- User management, including the ability to integrate to Hyperion Shared Services either in a stand-alone fashion for basic authentication or as a Foundation managed Common User Provisioning where both authentication and role authorization can be managed.
- DRM also makes available an API for custom interactions to be set up for activities such as task initiation and model interaction.
- DRM also provides Interactive, Batch and API capability to both Import and Export information from and to various sources and targets.
-

Finally a few terms used in this document that are common in DRM:

- Version: The primary container used in DRM to consolidate one or more Hierarchies of metadata. In addition to grouping, versions are often used to isolate point-in-time views of outlines.
- Hierarchy: A container that holds Parent-Child (Parent Member/Base Member) structure (relationships) between the nodes (members) and is the general interaction point for managing those relationships and setting property values on members.
- Node: Generically any member in DRM. DRM provides for conceptual Limb (Parent members because they may have Children or Descendants) and Leaf (Typically a Base Member; a node that is defined to not allow it to have Children or Descendants)
- Properties: Values set on a node, about the node, such as its Data Storage type Consolidation indicator, etc.

The interactive Import along with both the interactive and batch Export approaches will be discussed as the focus of this whitepaper.

Staging Outlines from EPMA and Planning in DRM

There is a customer base that today integrates from DRM through EPMA to Planning and other Applications indicating that the process chosen is quite flexible.

Outside of custom developed approaches, there are two general methodologies to instantiate DRM with metadata (the Outline) from either EPMA or directly from Planning. Importing the file from either is a relatively straightforward process.

The main difference comes in to play in that EPMA exports outline information in what is known as a .ADS formatted text file format providing sections for Dimensions (Versions and Hierarchies) and Hierarchies (member rollup structure), with member properties.

Planning, via the Administer → Import and Export menu approach or the Outline Load Utility (OLU) process, produces a single Dimensional text file in a Child – Parent - member property format.



We'll explore the details of each in the section following the Understanding DRM in Consideration for Preparing Outline Files topic.

DRM Considerations for Preparing Outline Files

Terminology sometimes gets a bit confusing when using terms such as data and metadata, especially keeping their usage in mind in the context of which application is being thought about. For example, metadata to DRM is generally considered to be the definitional elements such as those specified below. Metadata to an application, such as Planning, is often considered to be the values held by the DRM properties for any node (member) that will be sent to Planning. For example, a value of 'Never Shared' specified for the TD node (member) is 'data' in a DRM context but 'metadata' in the context of a Planning Application. I personally find the terms 'Configuration' and 'Content' less confusing.

DRM Metadata Migration Templates

We mentioned the availability of application templates (Templates) above. They are XML formatted definitional files that contain precisely specified information that instructs DRM how to create its own constructs.

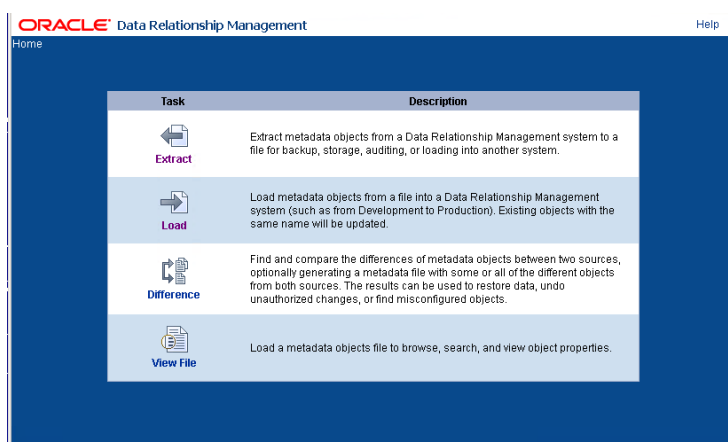
They can be of several types and examples of these are:

- Property Definitions
- Property Category Definitions
- Validation Definitions
- Import Specifications
- Export Specifications

among others.

For a specific example, the Template for Planning called `planning-app-template.xml` is by default installed in the

`install_drive:\Oracle\Middleware\EPMSysstem11R1\products\DataRelationshipManagement\server\app-templates` folder



These templates can be loaded into DRM via the DRM Metadata Migration Utility (MMU). It is access via a web browser generally at the <http://server-name/drm-migration-client/> URL.

This utility will allow a user to Load, Extract and/or view the XML files. So, it is a direct matter to load an initial configuration for a Planning Application to DRM via this utility.

Preparing an Import file for DRM

While it is certainly possible to write a small Java routine to read and parse an ADS file into a DRM expected format, moving from a product like EPMA to DRM is generally a one-off or one-time activity and it is typically more efficient to prepare the file in a spreadsheet interactively.

Files used to import a Version into DRM are at minimum expected to be comprised of at least two sections:

- A Hierarchy Section – typically referenced as [hier] in the import file for DRM
- A Relationship Section – typically referenced as [relation] in the import file for DRM

The Hierarchy section must specify the Hierarchy Name and the Top Node Name used in the Hierarchy. Optionally, it can contain values for properties of the hierarchy that you wish to load, such as a Description of the hierarchy, and any other valid System or User Defined Hierarchy Property values you wish to create Hierarchy Properties for. A typical example here is that if you know the Hierarchy will contain Shared Nodes, you will generally want to set the System Level Core.EnableSharedNodes value to True through the import to help minimize later warnings you may receive about Shared Members on import.

The Relationship Section is needed to specify the Parent-Child relationship structure between the Nodes and may optionally contain Local* Property values for each Node (Member). There is one Parent-Child specification per line. This is typically how you bring in values per node (member) for Data Storage, Consolidation Indicator, Two-Pass Calc, etc.

Optionally, you may specify a Node [node] section. This is sometimes helpful when you want to specify additional, Global property values for nodes (members).

Then, you can take advantage of the features in DRM to help you manage the metadata across Planning applications.

Additional detailed information can be found in the [Oracle Enterprise Data Relationship Management Documentation Guides](#).

Also, as indicated above, DRM ships with “Application Templates” associated with applications, such as Planning, that can be used to instantiate many DRM Property Definitions, Import and Export Specifications, Validations and other artifacts to help in this process.

We’ll discuss the DRM Import specification with an example next.

***Note:** There is a subtle difference between ‘Local’ properties in DRM and the ‘Relationship’ properties an EPMA or Planning user should be familiar with. In DRM, ‘Local’ means the value is local (or isolated) to the node WITHIN a Hierarchy, whereas in Planning, “relationship” it means it is consistent for any Parent/Child relationship across hierarchies, but within a dimension. While the difference is subtle, it may yield slightly different results on output than one might expect without this consideration in mind. In practice, it rarely causes an issue and if it does, can generally be mitigated by using DRM Formula logic.

‘Global’, in DRM, means that the node will hold only one value for a property for a node (member) regardless of its Relationship or Hierarchy. Generally, a property such as Account Type is global because its value for an Account node should not be different no matter its place in one or more (perhaps Alternate) Hierarchies.



```
|UDA|true
|Account|Corporate Accounts|true
Entity|Entity|true

on]
_NOLINK|Y|
_UDF|Y|
|No Account|Y|StoreData|N|N|Y|N|N|N|N|+|+|+|+|+|Currency|None|Revenue|None|NonExpense|Flow|N|None|FALSE|HSP_NOLINK|LastVal
|Statistics|Y|NeverShare|Y|Y|Y|N|N|N|N|~|~|+|+|+|+|+|NonCurrency|Plan1|SavedAssumption|None|NonExpense|Flow|N|None|FALSE|HSP_NOLINK|
ics|Revenue Driver Assumptions|Y|NeverShare|Y|Y|Y|N|N|N|N|+|~|+|+|+|+|NonCurrency|Plan1|SavedAssumption|None|NonExpense|Flow|N|None|FALSE|
Driver Assumptions|Units|Y|StoreData|Y|Y|Y|N|N|N|N|~|~|+|+|+|+|+|NonCurrency|Plan1|SavedAssumption|None|NonExpense|Flow|N|None|FALSE|
Total Entity|Y|NeverShare|Y|Y|Y|N|N|N|N|+|+|+|+|+|+|+|Unspecified|N|Total Department|Total Department|
ntity|TD|Y|NeverShare|Y|Y|Y|N|N|N|N|+|+|+|+|+|+|+|Unspecified|N|Total Department|Total Department|
StoreData|Y|Y|Y|N|N|N|N|+|+|+|+|+|+|+|Unspecified|N|No Department|No Department|
Y|NeverShare|Y|Y|Y|N|N|N|N|+|+|+|+|+|+|+|Unspecified|N|Resources|Resources|
|Y|StoreData|Y|Y|Y|N|N|N|N|+|+|+|+|+|+|+|Unspecified|N|Facilities Resources|Facilities Resources|
```

Column Number	Field	Example Values
1	Hierarchy Name	Account
2	Top Node (Member) name	Account
3	Hierarchy Description	Corporate Account
4	Enable Shared Nodes	true

For the [relation] section, while different sections of the file may have more columns than another, like the hierarchy section, each line must have the same number of columns within the section. The same delimiter, a pipe (“|”) in this example, must be used across the entire file. Note that in this example, there are 32 columns per line. Below is an example of the layout from the sample above and its logical mapping in a DRM Import specification.

Column Number	Field	Example Values
1	Parent	Account
2	Child	No Account
3	IsPrimary	Y
4	Data Storage	StoreData
5	Valid for Plan1	N
6	Valid for Plan2	N
7	Valid for Plan3	Y
8	Valid for CapEx	N
9	Valid for Workforce	N
10	Valid for Project	N
11	Aggregation for Plan 1	+
12	Aggregation for Plan 2	+
13	Aggregation for Plan 3	+
14	Aggregation for CapEx	+
15	Aggregation for Workforce	+
16	Aggregation for Project	+
17	Data Type	Currency
18	Source Plan Type	None
19	Account Type	Revenue
20	Exchange Rate Type	None
21	Expense Reporting	NonExpense
22	Time Balance	Flow
23	Smart List	
24	Two Pass Calc	N
25	Skip Value	None
26	Description	
27	Weeks Distribution	False
28	UDA	HSP_NOLINK
29	Alias Default	
30	Alias English	
31	Member Formula (BSO)	
32	Currency	LastVal

Here are some screenshots of a DRM Import Specification matched to the example above

This screenshot shows the 'Source' tab of the Oracle Data Relationship Management interface. The 'Device' is set to 'Client File'. The 'File Location' is empty, with a 'Browse...' button. Under 'Import Sections', 'Hierarchy' is selected, with 'Hier' set to 'Ex: [hier]' and 'Node' set to 'Ex: [relation]'. The 'File Format' section shows 'Character Encoding' as 'Unicode (UTF-8)', 'Strip Quotation Marks' is unchecked, 'Fixed Width' is selected, 'Delimited' is unchecked, and 'Skip First Rows In File' is set to '0'. There is also an option for 'No Section Header In File'.

This screenshot shows the 'Columns' tab. The 'Section/Relation' is set to 'Select Columns'. The 'Category' is 'System'. The 'Available' table lists columns: Label, Namespace, [Ignore Column], Description, Leaf, Node Approved, Sort Order. The 'Selected' table lists columns: Label, Namespace, 1* - Parent Node, 2* - Name, 3 - IsPrimary, 4 - Data Storage, 5 - Valid For Plan1, 6 - Valid For Plan2, 7 - Valid For Plan3, 8 - Valid For Capex, 9 - Valid For Workforce, 10 - Valid For Project, 11 - Aggregation For Plan1, 12 - Aggregation For Plan2, 13 - Aggregation For Plan3, 14 - Aggregation For Capital Asset Plan.

This screenshot shows the 'Target' tab. The 'Version Name' is 'TrialRun1'. The 'Version Description' is empty. The 'Max Errors' is set to '20'. The 'Assign to Variable' is empty. The 'Blender Options' section has 'Save Version To Repository' checked and 'Blend After Import' unchecked. The 'Blender Profile' is empty.

To summarize:

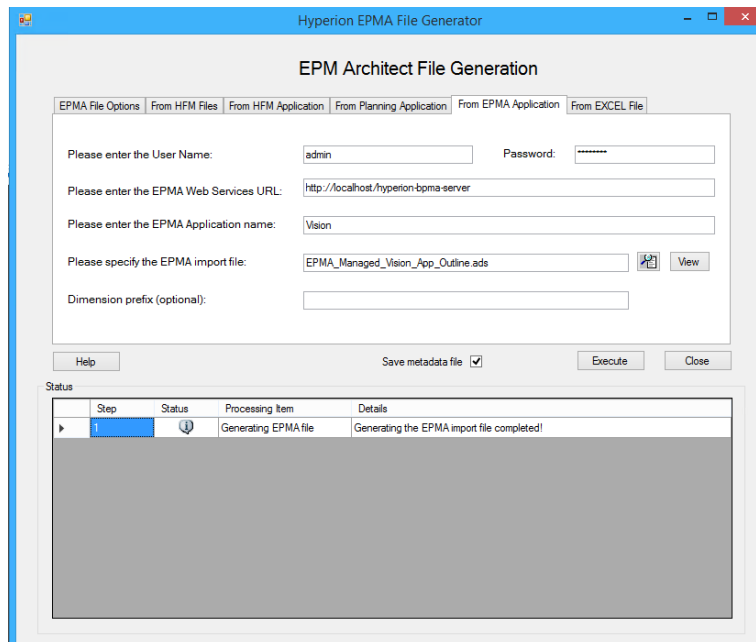
- Make the necessary modifications to the file from EPMA.
- Use or modify a dimension Import specification that was provided in the appropriate application template to match the layout of the modified file from EPMA or Planning where you will:
 - Point to the location of the file to be imported on the Source tab.
 - Use the Columns tab to specify the column order of values in the file layout.
 - Move to the Target tab and specify a name for the Version to be imported.

With this in mind, we will next discuss the options for an initial output of the metadata outlines from EPMA. This utility allows for the creation of an outline file from EPMA managed applications such as HFM, Planning and direct from classic mode Planning Applications to a .ADS formatted file.

Creating an ADS File from EPMA

EPMA provides a utility called the EPMA File generator, which normally has a default location of `%EPM_HOME%\products\Foundation\BPMA\EPMAFileGenerator\bin\EPMAFileGenerator.exe`

When you start the interface, you can navigate to the “From EPMA Application” tab, specify the credentials and connection then choose Execute to create the file.



Preparing an ADS Generated File

Now that we have an overview of the DRM Import paradigm and have gone through the process to generate the file, it is straightforward to prepare the .ADS file for import. As mentioned earlier while it is possible to write a Java routine to parse and re-array the file contents, generally it is a straightforward process to take the EPMA File Generator extracted file, bring it into a spreadsheet. To prepare for DRM, focus first on the Dimension Section (!Section=Dimensions) to determine the layout for the [hier] section of the DRM import, determining the Hierarchy Names and the property values desired to be mapped to the [hier] section of the DRM Import Specification.

Then, repeat the same process for the !Hierarchy= section in the .ADS file for the [relation] section of the DRM import file. In practice, very little generally needs changed, other than separating out the sections, one per Version. A key point is that if you change a Hierarchy Top Node Name, you must also change its value throughout the file.

It is even common to import all Hierarchies into a single Version in DRM where all Dimensions are Mastered together, but this is generally reserved to those customers that have developed a strong, isolated Node (Member) naming convention without collisions across the Dimensions.

But even then it is quite common for implementers to decide to prefix the Node (Member) names they bring into DRM, perhaps prefixing Account Members with ACCT_ and Entity members with CC_ or PC_, as appropriate. This allows Node Names to be unique within a DRM version. Then, the customer may choose to derive a Node Name value to be used in exports (by creating a Derived Formula property in DRM, stripping off the Prefix before sending it back to Planning); it just depends on the goal the customer wishes to achieve.

A general outline of the workflow is as follows:


Use the EPMA File Generator (Generally installed under
%EPM_ORACLE_HOME%\products\Foundation\BPMA\EPMAFileGenerator\bin\EPMAFileGenerator.exe)



<p>Choose the ‘From EPMA Application’ tab. Specify the credentials, the connection information and the output destination and ‘Execute’ the extract.</p>	
<p>A file in the ‘ADS’ file format will be created.</p>	<p>See this Oracle TechNetwork Tutorial for more information.</p>



The ADS file is a multi-section format file, generally consisting of several sections:	A two line file specification header indicating
	!FILE_FORMAT=ADS
	!VERSION=1.0
	One or more section headers in the format:
	!Section=Dimensions
	Or
	!Section=DimensionAssociations
	One or more Hierarchies part in the format:
	!Hierarchies=Account
	Followed by a pipe separated column header line in the format of:
	'Parent Child DataStorage ExchangeRateType MemberValidForPlan1 MemberValidForPlan2 MemberValidForPlan3 MemberValidForWorkforce MemberValidForCapex Plan1Aggregation Plan2Aggregation Plan3Aggregation WorkforceAggregation
Prepare the file information into a format that DRM is prepared to consume.	Followed by the Outline information itself
Create an Import Specification in DRM	#root 100 NeverShare Average Y Y Y N + + + +
Import the Dimension into DRM and save the Version.	Optionally, there may be Member information under the !Members= heading as in:
	!Members=Account
	And other sections such as a !PropertyArray= heading as in:
	!PropertyArray=Account



The ADS file may be manually prepared for DRM import via manipulating in a text editor or via a spreadsheet. Additionally, Oracle has provided a “.ADS” to DRM file conversion utilities to assist with initial preparation of the file for import into DRM.

This utility is provided via My Oracle Support to assist customers with converting EPMA generated “.ADS” files for their Planning, HFM, HPCM and Essbase outlines to a DRM import format. This can be acquired from My Oracle Support as Patch Number 3069700: EPMA ADS FILE CONVERSION UTILITY FOR EPM 11.2

(<https://support.oracle.com/epmos/faces/PatchDetail?patchId=30695700>)

Running the ADS_DRM.class file provided should provide a help screen. Typical instantiation syntax is

Java ADS_DRM

Exporting Outlines to Planning

After you've begun to manage your outline information in DRM, perhaps adding or rearranging members in a structure and updating property values, when you are ready to send the data back to Planning, you'll use the DRM Export functionality. Planning leverages a functionality it refers to as the Outline Load Utility (OLU). Specific documentation on the OLU may be found under the [Oracle Enterprise Performance Management System Documentation](#) and [Command Line Parameters for the Outline Load Utility](#) at [Oracle Docs](#).

The OLU expects a Comma Delimited File that contains a one-row header, wherein the columns to be imported are indicated with their order matching the layout of the content which is to follow. So, the top row contains the header and subsequent lines are the member information to import.

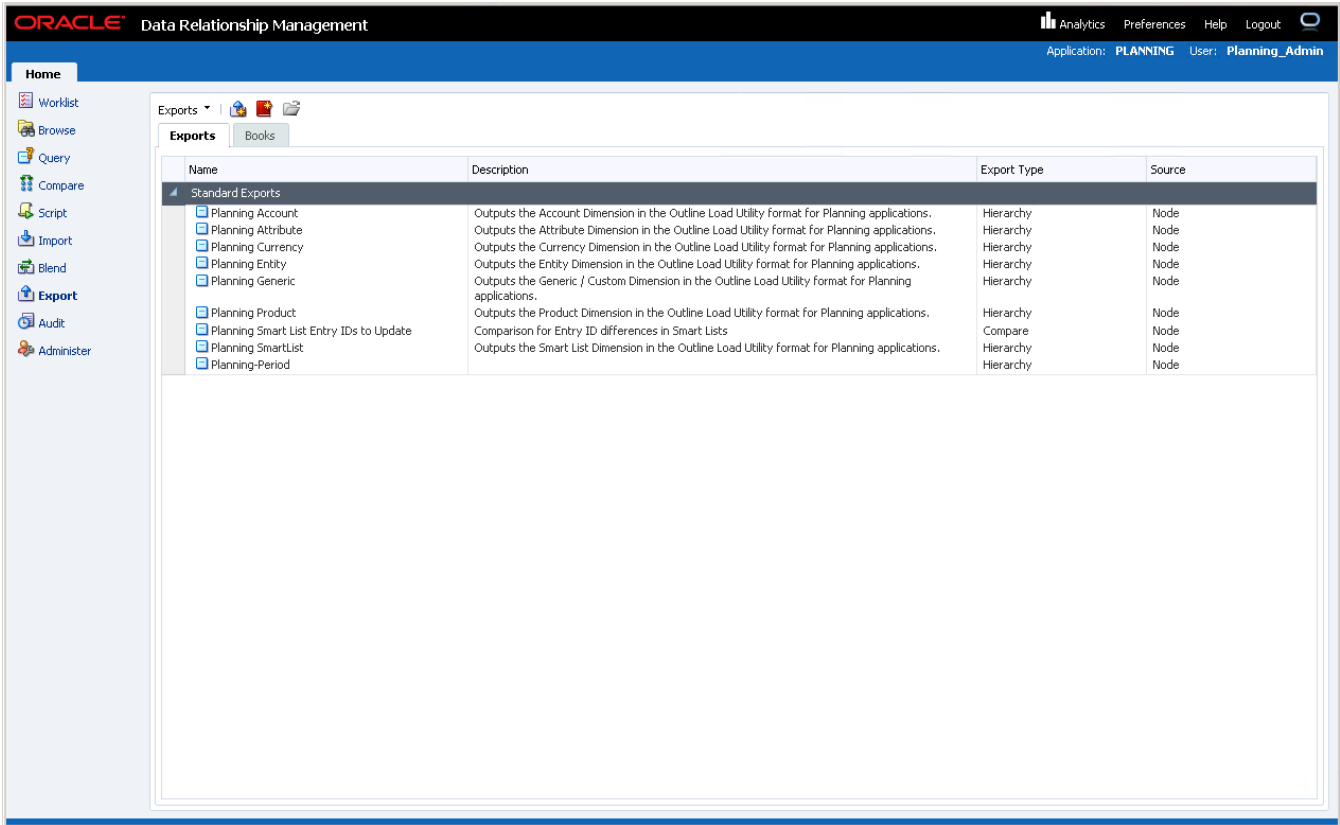
Typical OLU file example for an Entity Dimension																								
Entity, Parent, Alias: Default, Data Storage, Two Pass Calculation, Description, Formula, UDA, Smart List, Data Type, Base Currency, Plan Type (Plan1), Aggregation (Plan1), Data Storage (Plan1), Formula (Plan1), Plan Type (Plan2), Aggregation (Plan2), Data Storage (Plan2), Formula (Plan2), Plan Type (Plan3), Aggregation (Plan3), Data Storage (Plan3), Formula (Plan3)																								
"Total Entity","Entity","", "Never Share", "False", "", "<none>", "", "", "Unspecified", "USD", "True", "+", "Never Share", "<none>", "True", "+", "Never Share", "<none>", "True", "+", "Never Share", "<none>"																								
"TD","Total Entity","Total Department", "Never Share", "False", "", "<none>", "", "", "Unspecified", "USD", "True", "+", "Never Share", "<none>", "True", "+", "Never Share", "<none>"																								
"100","TD","Resources", "Never Share", "False", "", "<none>", "", "", "Unspecified", "USD", "True", "+", "Never Share", "<none>", "True", "+", "Never Share", "<none>"																								
"110","100","Facilities Resources", "Store", "False", "", "<none>", "", "", "Unspecified", "USD", "True", "+", "Store", "<none>", "True", "+", "Store", "<none>", "True", "+", "Store", "<none>"																								
"111","100","West Region Resources", "Store", "False", "", "<none>", "", "", "Unspecified", "USD", "True", "+", "Store", "<none>", "True", "+", "Store", "<none>", "True", "+", "Store", "<none>"																								

Note that above the information above is presented in a tabular format with each row of the file in one row of the table. This helps make it easier to distinguish separate lines in the file as the text wrapping is a limitation and artifact of printed page width restrictions. In actuality, each row of the table equates to one line in the .CSV file.



To create the file an export template specific to Planning is provided in the Planning Application template that provides several export layouts specific to out-of-the-box dimension configurations. Here, we will focus on the Planning Entity export definition reviewing its configuration and screens.

Below, we see the DRM export screen accessed via the Export menu link on the left on the Home tab. Here you will find several Export profiles pre-defined to serve as starting points for the exports to Planning



Choosing the Planning Entity export profile, we are taken to the detailed Export Definition screen. This page allows the user to choose the Version and Hierarchies to export. In this case, we have set up a Hierarchy Group to consolidate all Entity Hierarchies that comprise the Dimension that will be sent to Planning.

The screenshot shows the Oracle Data Relationship Management interface. The top navigation bar includes 'Analytics', 'Preferences', 'Help', and 'Logout'. The user is logged in as 'Planning_Admin' in the 'PLANNING' application. The main window has a tab for 'Planning Entity'. Below the tab are icons for 'Source', 'Style', 'Filter', 'Columns', and 'Target'. The 'Source' tab is active, displaying a form with the following fields:

- Version:** A dropdown menu with a 'Normal' link next to it.
- Hierarchy Selection:** A dropdown menu with 'Hierarchy Group' selected.
- Hierarchy Group Property:** A dropdown menu with 'HP Dimension' selected.
- Hierarchy Group:** A dropdown menu with 'Entity' selected.

Clicking on the Style tab, there are options specific to overall setting that can be configured for the export. This includes defaulting to all Nodes within the Hierarchy (as opposed to only Limb or Leaf nodes). Other options allow for starting at the Top Node of the Hierarchy, Including Inactive or Implicitly Shared members and even the ability to remove certain duplicates.

Also, sets of Validations can be configured to run before the export, so content issues can be caught and remedied before proceeding to send the file, along with a few lesser-used options.

The screenshot displays the Oracle Data Relationship Management (DRM) web application interface. The top navigation bar includes the Oracle logo, the title "Data Relationship Management", and links for Analytics, Preferences, Help, and Logout. Below this, a sub-header shows the current application as "PLANNING" and the user as "Planning_Admin". The main content area is titled "Planning Entity" and features a tabbed interface with "Source", "Style", "Filter", "Columns", and "Target". The "Style" tab is active, showing configuration options for data export. Under "Node Selection", the "All Nodes" radio button is selected, with options for "Limb Nodes Only" and "Leaf Nodes Only". The "Options" section includes checkboxes for "Recurse from Top Node" (checked), "Include Inactive Nodes" (checked), "Include Implicitly Shared Nodes" (unchecked), "Remove Duplicates Based on Key" (unchecked), and "Tabbed Output" (unchecked). A "Batch Validations" section contains a "Run Validations Before Export" dropdown menu set to "Selected". The "Repeat Parameters" section includes dropdowns for "Repeat Number Property" and "Bottom Level Property", a "Bottom Level Value" input field set to "0", and a "Repeat Nodes to Enforce Bottom Level" checkbox (checked) with radio button options for "Repeat Bottom Limb Node" (selected) and "Repeat Bottom Leaf Node".

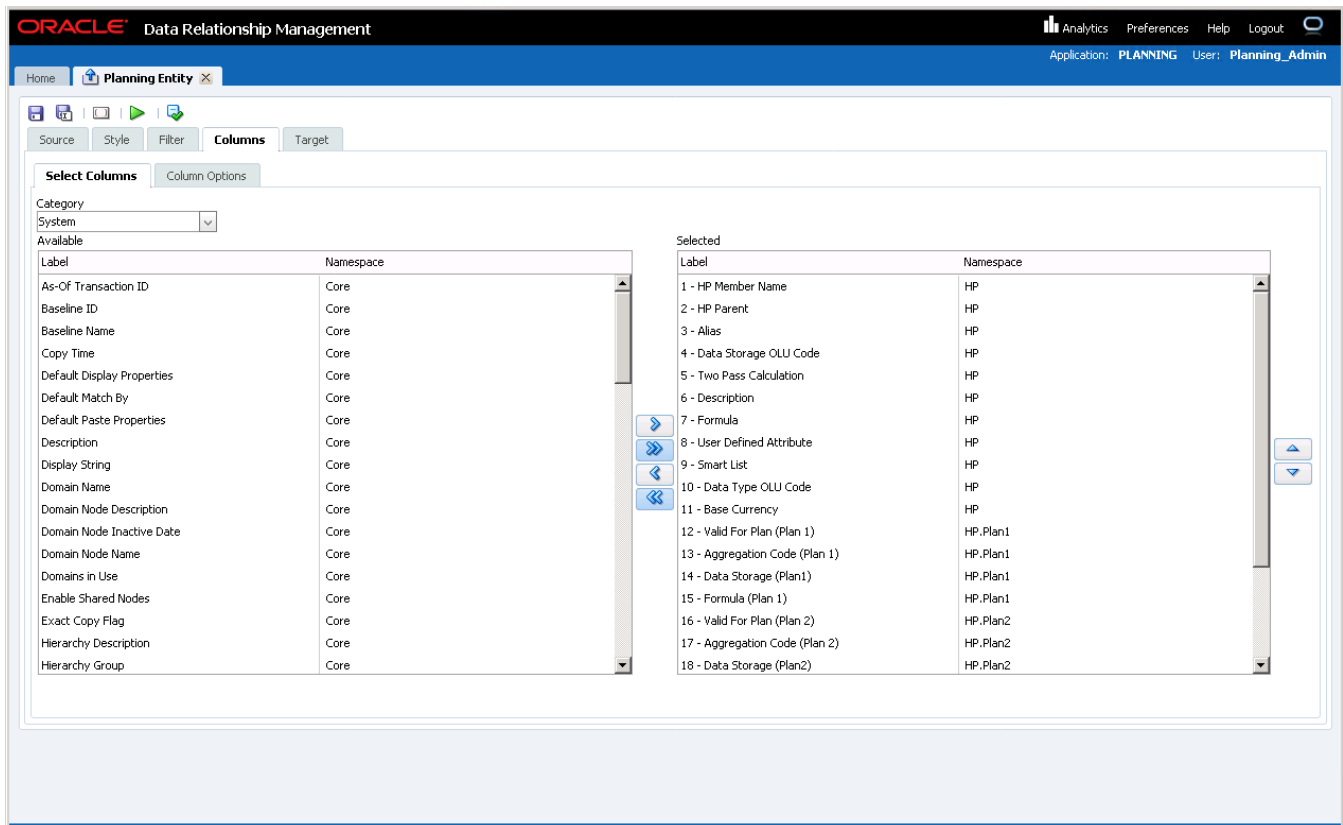
On the Filter tab, an additional Validation can be specified that is generally used where a Validation is set up to only include nodes that pass the validations to be included in the export.

Below, an Inclusion filter can be specified where a query can be specified to target an overall population set. In this example, it is defined to only include those members in the Hierarchy that have been specifically set as a Parent or Base Member for Planning. Note that it is specified as “Private Query”, which indicates it is only available to users configured with the correct roles and access setting to prevent general Interactive Users from unintentionally changing the setting.

The screenshot displays the Oracle Data Relationship Management (DRM) web application interface. The top navigation bar includes the Oracle logo, the title "Data Relationship Management", and links for Analytics, Preferences, Help, and Logout. Below this, a secondary bar shows the current application as "PLANNING" and the user as "Planning_Admin". The main interface has a tabbed menu with "Source", "Style", "Filter", "Columns", and "Target". The "Filter" tab is active, showing a "Validation" dropdown menu. Below this, the "Inclusion Filter" section contains a "Query" dropdown set to "[Private Query]" and a "Description" text area with the text "HP Membership In Parent Member,Base Member". A "Clear" button is located to the right of the description. The "Descendants File" section includes a checkbox for "Use text file to include/exclude descendant nodes.", a "* Connection" dropdown set to "Local_Planning_BackupToFile", and a "* File Name" text input field. Below these, there are radio buttons for "Include mode" and "Exclude mode".

On the Columns tab, here is where the properties values held in DRM are specified to be output in a specific, pre-defined order to the file to be imported by Planning. Items available are presented in the left pane while items already selected are presented in the right pane.

Interaction is simply a matter of double-clicking an item on the left to specify it for inclusion or highlighting as selection and using one of the bluish arrow shuttle controls to move a selection between panes. On the far right, the up/down arrows can be used to re-order a highlighted item or set of items within the selected list.



Finally, we are presented with the Target tab where users can specify the Device to be used. In this case it is set to Server File, because we intend to direct the output directly to the Planning server via a Connection we have set up earlier. We also specify the specific file name for the metadata set to be used in the File Name box in the top row.

Note we have other options available such as using the Column Headers as specified in the labels associated with the DRM property definitions but in this case, we have chosen to provide specific column layout information for the file in the Header section on the screen.

Below Header, we can specify a standard or custom delimiter to be used in the output file, in this case a Comma and that we want to terminate each line of the output with Carriage Return/LineFeed (CRLF) combination characters.

There is also an option available to Replace several characters “on-the-fly” during the export, though this is less commonly necessary.

The screenshot shows the Oracle Data Relationship Management (DRM) Target tab configuration interface. The top navigation bar includes 'ORACLE Data Relationship Management', 'Analytics', 'Preferences', 'Help', and 'Logout'. The application context is 'PLANNING' and the user is 'Planning_Admin'. The 'Target' tab is active, showing configuration options for the export target.

Target Configuration:

- Device:** Server File
- * Connection:** Planning_Application_Server
- * File Name:** Entity_From_DRM.csv
- Description:** Planning_Application_Server

Format Options:

- ☐ Column Headings
- ☒ Quoted Strings
- ☐ Fixed Width
- Character Encoding:** Unicode (UTF-8)

Header / Footer:

- ☐ Blank line between Header/Footer and Body
- Header:** Entity, Parent, Alias: Default, Data Storage, Two Pass Calculation, Description, Formula, UDA, Smart List, Data Type, Base Currency, Plan Type (Plan1), Aggregation (Plan1), Data Storage (Plan1), Formula (Plan1), Plan Type (Plan2), Aggregation (Plan2), Data Storage (Plan2), Formula (Plan2), Plan Type (Plan3), Aggregation (Plan3)
- Footer:**

Replace:

- (None) with (None)
- (None) with (None)
- (None) with (None)

Field Delimiter: (Comma)

Record Delimiter: (CRLF)

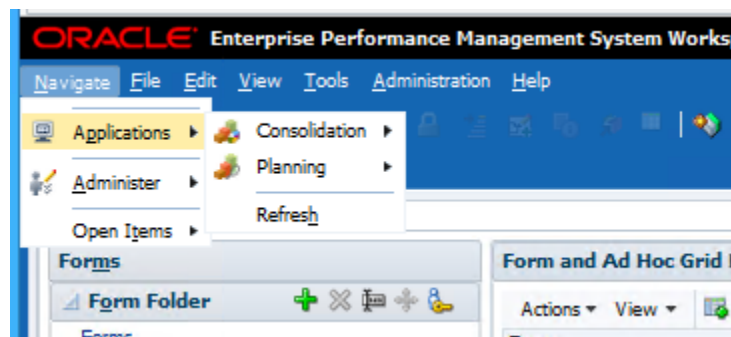
To Run the export, simply click the Green Right Arrow just under the Tab to initiate the export to Planning.

Import File in Planning

In Planning, the file can be imported via the Administration menu item in the classic interface and optionally imported via the OLU command line utility. We'll discuss the use of OLU in the next section where we'll expand the topic to discuss automation strategies.

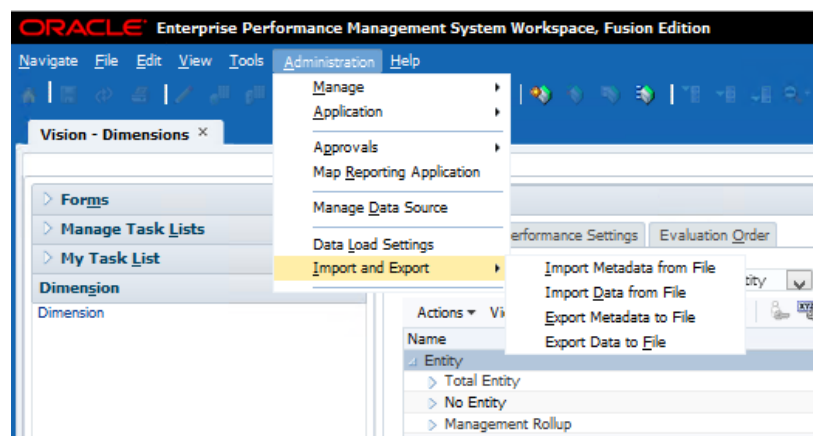
Interactively, it is a straightforward matter. Open the Planning application, generally accomplished by choosing

Navigate → Applications → Planning → and the specific Planning application you wish to open

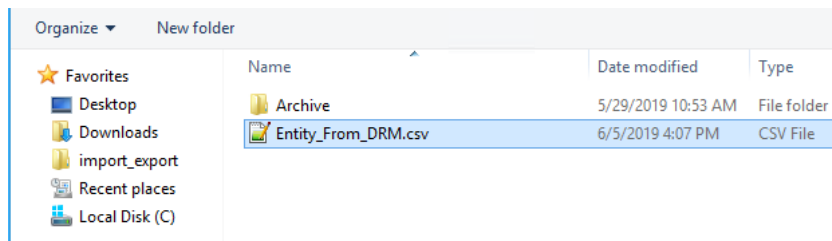
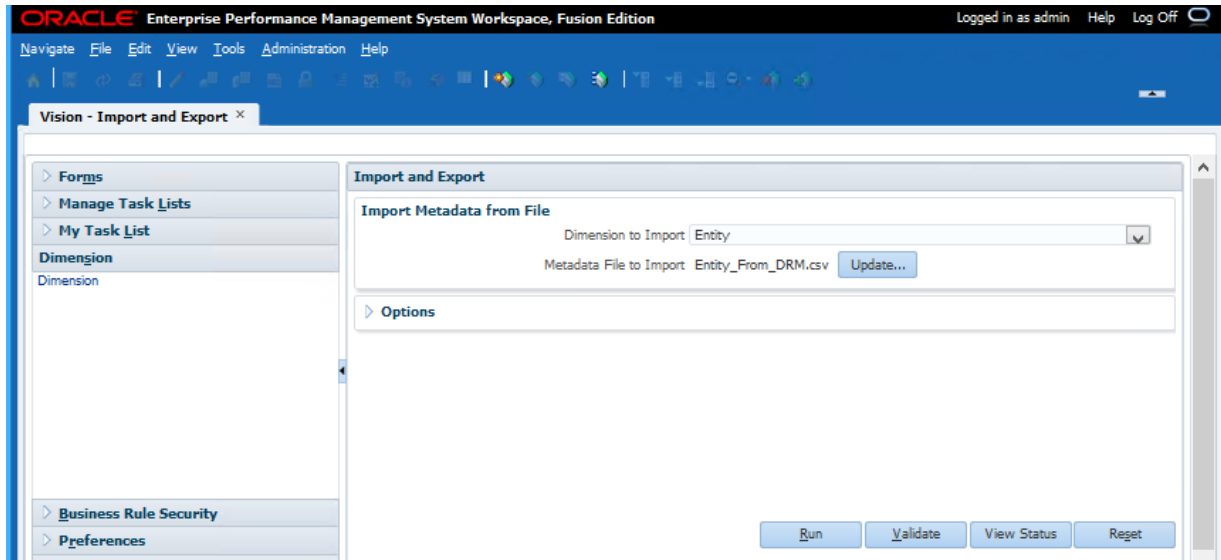


Then from the Workspace menu choose:

Administration → Import and Export → Import Metadata from File



To follow with this example scenario, choose Entity, Browse to the location where the file was delivered from DRM and select the file.



Next it is a matter to run a Validate (button on lower right) which will check the file for obvious format issues. Once complete, click on the Run button and the outline file will be loaded into Planning.

Hyperion Profitability & Cost Management Considerations

An EPMA extract from an HPCM application is similar to that of one from a Planning application with the addition of a 'Members Section' that may provide additional properties.

The key consideration here is that the top hierarchy node in a DRM hierarchy represents the top dimension member in HPCM applications. Therefore, it is important to import the properties for the top node for each dimension in the Members section into the DRM application so that when exporting back to HPCM, the needed attribution can be sent.

Regarding exports, HPCM has a requirement that when sending text values back to HPCM, they must be wrapped in double quotes (") in the export file however, property values that are empty (blank or never initialized) may NOT be wrapped in double quotes. To accomplish this, a set of derived DRM properties specifically intended for export must exist that simply take the value of a text property and wrap it in double quotes if that property has a value and that is what is sent back to HPCM.

As an example, if the property COMMENT may or may not have a value, then a supporting or 'shadow' property to be used in the DRM export back to HPCM should be created. As an example, for the Comment property then a derived formula property such as EX_Comment may be created and can contain logic similar to the following:

```
If(GreaterThan(Length(PropValue(HPCM.Comment)),0),Concat(",PropValue(HPCM.Comment)",PropValue(HPCM.Comment))
```

In this example, the logic in Ex_Comment simply determines if the HPCM.Comment property has a value, and if it does, wraps it in double quotes using the Concat function, otherwise produces output with no value. To assist with this, a HPCM application template is provided with the DRM release that contains many common properties and the associated 'shadow' properties that can be used for exports.

Since all implementations vary in their level of complexity and customization, the templates serve as a starting point, containing many of the common HPCM properties.



Automation Strategies

There are two components of interest here. DRM installs include the `drm-batch-client.exe` (Batch Client) that allow for the batch scripting of several DRM functionalities, such as Imports, Action Scripts and Exports.

Planning provides the Outline Load Utility (OLU) command line utility to, as the name indicates, load outline files to a dimension.

Since both are command line utilities, they are often used in batch file scripts to automate extract and load processes. The general strategy here is to create an export command using the Batch Client and an import command using the OLU.

The DRM Batch Client can be a separate install to the Windows® based the DRM Application Server and/or the Planning Application Server.

The OLU is typically deployed just to the Planning Application Server.

If the DRM Batch Client is installed to the Planning Application Server, script-based automation can easily be set up to be orchestrated from one location.

Next, we'll discuss the format and syntax of both a DRM Batch Client command and an OLU command.

Command Syntax – DRM Batch Client

For the DRM Batch Client, there is a specific syntax for each operation desired. For example, an Import command has one syntax, an Action Script command has its syntax and Exports have their unique syntax.

On the DRM Application Server, for a typical installation, open a console window and navigate to the *Install_Drive:\Oracle\Middleware\EPMSys11R1\products\DataRelationshipManagement\client\batch-client* folder.

In addition to the [DRM User Guide](#) documentation, a help screen can be initiated by typing

```
drm-batch-client.exe /?
```


In the console windows to get a help screen where the syntax per operation is provided.

For an export from DRM using the Batch Client, below is an example command line:

DRM Batch Client Example Sytnax	
<pre>drm-batch-client.exe /u=admin /pw=s3cr3tp\$\$\$w0rd /url=net.tcp://servername:6789/Oracle/Drm/Engine /objectaccess=System /log=c:\PlanningExportBatchClientlog.txt /op=Export /xtype=E /xname="Planning Entity" /cver=Planning-Entity /pver=Planning-Entity</pre>	

Like other examples above, the example is bounded by a table to make it easier to read, but parameters should be entered as all one line with a Return keypress being used to execute the command.

Command Part	Explanation
drm-batch-client.exe	Name of the command file
/u=admin	The DRM Username to use for the connection. See next note.
/pw= s3cr3tp\$\$\$w0rd	The password of that user. NOTE: DRM also provides a drm-batch-client-credentials.exe utility that allows for keystore storage of both an encrypted username and password that are associated to the account logging into the Application Server. If that is set up, then credentials are secured and do not need to be specified on the command line or in a batch file.
/url=net.tcp://severname:6789/Oracle/Drm/	This value is specific to the application you have set up in DRM and can be found in the DRM Console for the specific application you want to interact with, under the Host Machines → Engine tabs for that application.
/objectaccess=System	This specifies the Object Access Level (Authority) under which the customer has staged the Export specification definition
/log=PlanningExportBatchClientlog.txt	Specifies a file name for the log file that gets generated when the export is executed. An accessible path may be specified.
/op=Export	Instructs the batch client that you wish to perform an export
/xtype=E	Instructs the batch client that you wish to perform a standard export as opposed to a book (combined file) export
/xname="Planning Entity"	The name of the defined export to run. Note that since the export name contains a space, it must be wrapped in a pair of double quotation marks.
/cver=Planning-Entity	Current Version name. The name of the version to be exported
/pver=Planning-Entity	Previous Version name. Note that even for cases where a hierarchy export is chosen and no comparison is needed, a previous must be specified and should be the same as the /cver value



Run the command line and the file will be generated and output to the destination as defined in the export definition. Note that there are other options available that can be researched in the DRM Admin and User Guides

Command Syntax – Planning OLU

The Outline Load Utility (a .cmd file) has over three dozen possible parameter when viewed in the context of the overarching HspOutlineLoad Java that it calls. For example, a separate password file for user authentication(s) is generally expected to be set up. This example will illustrate the basics.

Outline Load Utility Example Syntax	
OutlineLoad	/A:Vision /U:admin /M /N /I:c:\File_From_DRM.csv /D:Entity /L:c:/outlineLoad.log /X:c:/outlineLoad.exc

Command Part	Explanation
outlineLoad.cmd	Name of the command file
/A:Vision	The name of the Application to load the file to
/U:admin	The Planning Username for the connection
/M	Specifies that the load is a Metadata load
/N	Used to validate the load file. Should be removed or specified as /-N to perform the actual load
/D:Entity	The name of the Dimension to load
/L:c:/outlineLoad.log	Specifies a file name for the log file that gets generated when the export is executed. An accessible path may be specified.
/X:c:/outlineLoad.exc	Specifies a file name for an exceptions that may get generated when the export is executed. An accessible path may be specified.

An example batch file would simply be execution of the two utilities from within one batch process as follows:

Example Batch File	
Line	Command
1	REM – Export the Entity information file from DRM
2	drm-batch-client.exe /u=admin /pw=s3cr3tp@\$w0rd /url=net.tcp://servername:6789/Oracle/Drm/Engine /objectaccess=System /log=c:\PlanningExportBatchClientlog.txt /op=Export /xtype=E /xname="Planning Entity" /cver=Planning-Entity /pver=Planning-Entity
3	REM – Import the Entity dimension into the Planning Vision Application
4	call OutlineLoad /A:Vision /U:admin /M /N /I:c:\ File From DRM /D:Entity /L:c:/outlineLoad.log /X:c:/outlineLoad.exc

Of course, a practical implementation would add lines such as a manageable directory structure to store the files, checks for file existence before process execution, checking for error conditions, time and date stamping the files and archiving the files processed.



Automation Summary

So, with the DRM Batch Client (and potentially the Batch Client installed on the Planning Application Server) and the Planning Outline Load Utility, batch file case be created to call both processes and orchestrate the entire process.

It is important to note that for this process, other DRM components such as the DRM Application Server and the DRM Metadata Migration Utility do not need to be and generally should not be installed and must not be running on the Planning Application Server unless there are other considerations at play, such as an integration with Foundation Shared Services or directed to do so by an Oracle support representative.



Moving to EPM 11.2.x – Initial Release

Considerations

Given that EPMA is not planned to be available in EPM 11.2 and future releases and given that the EPM 11.2 portfolio is being positioned as a technology upgrade, to provide support for newer operating system, as opposed to providing additional features, the following approach is offered for consideration.

Parallel Environments

Again, since the key driver behind the 11.2.x line is upgraded technology support, it is reasonable to assume that it will be instantiated in a new hardware (or at least new virtual machines) and software landscape. Therefore it is expected the 11.2.x line will be available in parallel with any 11.1.2.4.x implementations.


Migrating 11.1.2.4.x Applications to 11.2.x – Initial Release

Because of the need for parallel environment, there is not a direct software directed upgrade path (no “one-button” upgrade option) anticipated because of the landscape, customers will need to migrate their applications from their 11.1.2.4.x environments to their new 11.2.x environments.

With the deprecation of EPMA, that clearly means that EPMA will not be available in the 11.2.x environments so customers will need to develop a strategy to and implement a process to migrate applications.

While there may be several strategies that can be developed we present an outline for one approach along with some detailed steps for review by presenting an approach for Planning Applications.

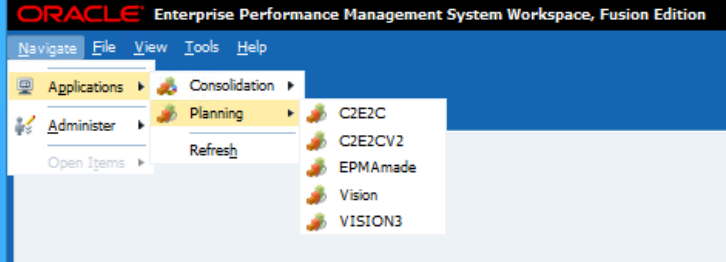
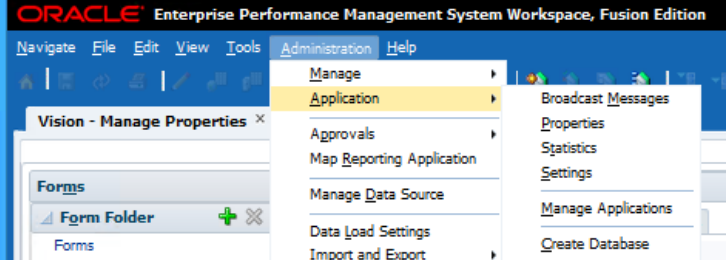
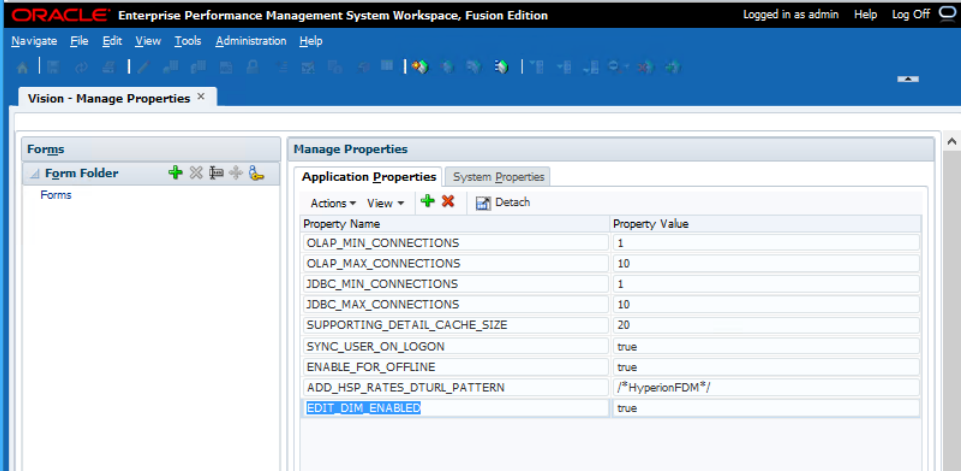
Migrating an EPMA Outline Managed Planning Application to 11.2.x – Initial Release



As Workspace/EPMA users and administrators should be aware, outlines managed via EPMA can be edited only in the EPMA perspective of the outline. Edits to the outline is disabled in both the Planning ‘Classic’ and Simplified Interfaces. Since EPMA will not be available in the 11.2.x landscape, the Planning Application must be converted back to the mode where editing will be permitted directly in Planning. This is accomplished by setting the value of an application level Property termed EDIT_DIM_ENABLE to true

Below is an outline of these steps with some associated screenshots:



<p>Regardless of:</p> <ul style="list-style-type: none">a Planning Application being initially set up as a standalone Planning App first then Converted to an EPMA Managed App via the Navigate → Administer → Transform Classic to EPM Architect	<p>Or:</p> <ul style="list-style-type: none">a Planning Application created through the Workspace/EPMA Navigate → Administer → Application then the File → New → Application pathway, then deployed
The same common approach can be used for each application to be converted:	
Steps	Screenshot
<p>Open the Planning App from Workspace menu via the Navigate → Applications → Planning → <i>application_name</i></p>	
<p>Once opened, from the menu choose Administration → Application → Properties</p>	
<p>From the Manage Properties window:</p> <ul style="list-style-type: none">If the EDIT_DIM_ENABLE Property does not exist, add it, or otherwise edit it to:Set its Property Value to true	

- Be sure NOT to alter any other Property values available as the system may become unusable.
- Save the change
- Close all tabs and Log off from Workspace
- Fully Stop then Restart all the EPM System Services

Once the system has been restarted:

- You will be able to edit outlines directly in either the Planning specific Classic (Workspace based) or the Simplified Interface
- DO NOT make any edits via the EPMA interface and DO NOT DEPLOY again from EPMA
- While the application may still appear available as an EPMA Managed Application in Workspace, only make edits in the Planning specific Classic or Simplified Interface
- IF YOU DEPLOY THE APPLICATION AFTER THIS CHANGE SUCH AS FROM THE Navigate → Administer → Application Library → Deploy Action: YOU WILL OVERWRITE ALL DIMENSION OUTLINES and the application may become unusable
- DO NOT delete the Application from the EPMA Application Library

After the Applications have been validated, you may make the necessary LCMs and post them to Apps in the 11.2.x environment.

OPTIONAL:

It is possible to remove the reference in Workspace to a formerly EPMA managed version of a Planning Application; those that show in the Navigate → Administer → Application Library listing. This will remove its visibility in Workspace under EPMA. Please contact Oracle Support for guidance.