

Add-on for Oracle Hyperion EPM Documentation

- User Guide
 - Introduction
 - Hyperion dimensions in the EBX® repository
 - Extracting financial data
 - Configuring data mapping
 - Import and export procedures
 - Comparing dimensions
 - Mapping data generation procedures
 - Report of import and export execution
 - Business rules
 - API and software resources
 - File format
 - Error management
 - Data staging
 - Installation
 - Appendix
- Release Notes
 - Version 4.6.0
 - All release notes

User Guide

- Introduction
- Hyperion dimensions in the EBX® repository
- Extracting financial data
- Configuring data mapping

- [Import and export procedures](#)
- [Comparing dimensions](#)
- [Mapping data generation procedures](#)
- [Report of import and export execution](#)
- [Business rules](#)
- [API and software resources](#)
- [File format](#)
- [Error management](#)
- [Data staging](#)
- [Installation](#)
- [Appendix](#)

Introduction

- [Overview](#)
- [Intended users of the add-on](#)
- [User prerequisites](#)

Overview

The TIBCO EBX® Add-on for Oracle Hyperion EPM facilitates management of financial data dimensions within your repository. You can leverage all TIBCO EBX® features for secure data governance, such as permission and version management (data space, data set), approval workflows, and audit-trails.

The EBX® Add-on for Oracle Hyperion EPM allows you to import and export data between your repository and any financial application (For Hyperion these are: HFM, Essbase and Planning). Configurations determine how dimensions in your repository map to financial application data during import and export.

Another key component to dimension management is comparing dimension versions. This process enables you to understand how a dimension has changed over time. You can use the add-on's 'Compare' service to accomplish this.

SPECIAL NOTATION KEY:	
☐	Important recommendation for the use of the feature
☐	This feature is not yet available in the current release

SPECIAL NOTATION:	
☐	The add-on is specialized in Oracle® Hyperion EPM. Other financial products will be provided in a further release.
☐	From the version 1.5.0, the add-on can be integrated with the TIBCO EBX® Match and Cleanse Add-on functions and with the TIBCO EBX® Rules Portfolio Add-on. The first add-on allows you to de-duplicate and clean up the financial data. The second one is used to enforce a transparent management of the business and permission rules applied to the financial data (rules portfolio, rules configuration, traceability of the rules execution). Please refer to the related appendix for more information.

Intended users of the add-on

The following types of users interact with this add-on:

- Business users who manage the dimensions and members in the financial application such as Hyperion. These users interact with the add-on in order to prepare, check, and query dimensions before exporting them to the financial applications.
- The IT user who prepares the data configurations required to drive the mapping between EBX® data structures and the financial dimensions.

User prerequisites

Knowledge of EBX® is a prerequisite to use the add-on.

Hyperion dimensions in the EBX® repository

- Add-on datasets
- Representation of Hyperion dimensions in EBX®
- Navigating relationships in the repository
- Active and '#root' fields
- 'Send to end' a node in a data hierarchy
 - Applied on a data hierarchy view
 - Applied on a table view
 - Applied on a record view
 - A record is child of more than one parent records
 - Record with no parent
 - Record in the group 'Active unset'

Add-on datasets

The add-on is based on four main data sets:

- 'Oracle Hyperion EPM Dimensions' that collects all business information of the financial data.
- 'Oracle Hyperion EPM Extract Configuration' that allows you to extract data based on configuration settings.
- 'Oracle Hyperion EPM Mapping' that defines data mapping between EBX® and Hyperion applications.
- 'Oracle Hyperion EPM Report' that collects all import/export result information.



Every data set created in the **Oracle Hyperion EPM Dimensions** dataset is based on a group of fields named 'Financial Data'.



Below is an expanded view of the 'EPM Mapping':



Representation of Hyperion dimensions in EBX®

A Hyperion dimension is defined in the EBX® repository as four tables, as follows:

- The main table contains records with all the fields of the dimension grouped per type of application (HFM, Essbase, Planning). For example, the table 'Account' includes the field's name, description, etc. In the main group 'Hyperion financial management', the shared fields between all types of application are located in the group named 'Shared information'.
- The hierarchy table contains all parent-child relationships between records in a dimension. The naming convention for the label of the table appends "Hierarchy" to the name of the dimension. For example, for the 'Account' hierarchy, the technical name of the table becomes "AccountHierarchy".
- The description table contains all the localized names of records for a dimension. The naming convention appends "Description" to the dimension name. For example, the technical name of the description table of 'Account' becomes "AccountDescription".
- The setting table contains all technical parameters used by Hyperion applications. This data is grouped by 'Dimension Attributes' and 'Dimension Properties'.

Some dimensions do not have any relationship definitions or descriptions and thus only include the main table. This is the case for the dimensions ICP, Security Class, View, Alias.

Navigating relationships in the repository

For each dimension with managed relationships, a hierarchical table view is configured in the EBX® repository on the main table of the dimension. This hierarchical table view is named 'Parent to child'. Using this tree view, it is easy to access and handle relationships, for example, to attach, detach, move, delete, and create children.

Active and '#root' fields

To enable the management of Hyperion relationships within the hierarchy view in EBX®, the add-on uses two fields:

- **active**: a boolean field that is automatically initialized to 'true' when the add-on creates a new relationship for a dimension. To omit a relationship from the hierarchy, set this field in the relationship to 'false'. During normal usage of the add-on, only the 'true' branch of the hierarchy is used. The user service 'Switch active', available on each hierarchy table, allows selecting multiple records and activating or deactivating them all at once. During the export procedure, relationships set to 'false' are not exported.
- **#root**: the first level of the hierarchy.



'Send to end' a node in a data hierarchy

This service is used to send one or more records to the last children position in a the data hierarchy. It works on data hierarchy, table and record views.

Note: the service 'Send to end' is available for the add-on only. This is not a generic EBX® feature.

Applied on a data hierarchy view

'Send to end' for multiple nodes

- Select some nodes and use the service 'Send to end':



- After the execution of the 'Send to end':



'Send to end' for one node only

- On one node, the service 'Send to end' is executed:



- After the execution of the 'Send to end':



Applied on a table view

The behavioral is the same than one applied on the data hierarchy views.

- Select some records and execute the 'Send to end':



- Open the Data Hierarchy view to check the result of the 'Send to end':



Applied on a record view

The record is not in the group 'Active unset'

On a table records view, when the service 'Send to end' is executed, the system applies the same procedure than on saw in previous section. For example:

- On one record, use the service 'Send to end':



- Return to the data hierarchy view to check the result:



The record is in the group 'Active unset'

This case is unsupported because all records in the group "Active unset" don't have neither parent nor child.

A record is child of more than one parent records

When a record which is a child of many parent records is selected, all nodes containing this record as a child are selected on the data hierarchy view:



Then, all of these nodes are sent to end:



Record with no parent

It is impossible to send to end a record without parent. For example, the record '614100' have no parent and the execution of the service 'Send to end' raises an error.



Record in the group 'Active unset'

This case is unsupported because all records in the group "Active unset" don't have neither parent nor child.

Extracting financial data

- [Overview](#)
- [Creating an extract configuration](#)
 - [EBX® repository data structure](#)
 - [Property](#)
 - [File group](#)
 - [Extract file](#)
 - [Extract column](#)
 - [Value map](#)

- Map pair
- Running the Extract service
 - Extracting data for all files
 - Extracting all files in group(s)
 - Extracting selected files
 - Extracting one file on a hierarchy
 - Extracting a file in 'Extract file' record view

Overview

The add-on allows you to extract data from an EBX® Add-on for Oracle Hyperion EPM financial dataset and export it to a file. You can define a configuration that specifies content, file format, and any specific data filters. Use the 'Oracle Hyperion EPM Extract Configuration' dataset to specify configuration options and then run the 'Extract' service to export the data.

Creating an extract configuration

The 'Oracle Hyperion EPM Extract Configuration' dataset allows you to define the target financial dataset with the 'Property' group and store the file configuration using the 'Extract file' table. Additionally, this table allows you to:

- Define the file format options, such as, file name, header, footer, and file type.
- Define file content, such as, columns and dimension.
- Filter the content for the file: top nodes and filter.

You can categorize your Extract files using the 'File group' table and override your financial data on your file using the 'Map pair' and 'Value map' table. See the section below, *EBX® repository data structure*, for detailed information about each option in the dataset.

Once configuration is complete, you can run the 'Extract' service from several locations. See *Running the Extract service* for more information.

EBX® repository data structure

The descriptions below provide an overview of the table structure in the repository.

Property

GROUP	FIELD	DEFINITION
Property	Data space	Target data space/ financial data set to extract data.
Financial data set		

File group

TABLE	FIELD	DEFINITION
File group	Name	Name of the group. This field is mandatory and unique.
Description	Description for the group. You can input all the description fields in multiple languages.	
Extract files	Association field. A list of extract files which have FK to the record.	

Extract file

TABLE	FIELD	DEFINITION
Extract file	File group	FK File group. Select group for the Extract file.
Dimension	Mandatory. Select a dimension for the file. Each extract file has only one dimension.	
File name	Name of the file. This field is mandatory and unique.	
Top nodes	Multiple values. Extract top nodes and its descendants.	

TABLE	FIELD	DEFINITION
Filter	Filter the extract result. We provided a component as the filter component of EBX®.	
File type	Select the display style for the file. There are 3 types: - Parent/Child: include parent and child hierarchy nodes - Child: only leaf nodes will be extracted. - Flattened: like the rainbow file with level columns determined dynamically by the actual depth of the hierarchy.	
Delimiter	Mandatory. Select "PIPE" to extract the TXT file, "COMMA" to extract the CSV file.	
Include header	Include file header or not.	
Quoted string	Select 'Yes' to wrap the value by quoted character.	
Header	Input header for the file.	
Footer	Input footer for the file.	
Blank line HF	If yes, display a blank line between body - header or body - footer.	
Extract Columns	Association field to display the extract columns.	

Extract column

TABLE	FIELD	DEFINITION
Extract column	Extract file	Mandatory. FK to Extract file.

TABLE	FIELD	DEFINITION
Column name	Name of column in the extracted file. If you do not input the name, the add-on will extract title of the field on the financial data set.	
Table type	Type of the table for the Field path. The options of Field path component will be generated after selecting the table type. There are 3 table types: - Node: hierarchy table - Member: member table - Member extension: member table on extension data set.	
Field path	Store the MDM field path.	
Override shared	If Override shared is set to 'Yes', the value on shared value will be extracted instead of the value on financial data set.	
Shared value		
Value map	Source and target value to extract.	
Index	Order of the column on extracted file.	

Value map

TABLE	FIELD	DEFINITION
Value map	Name	Value map name. This field is mandatory and unique.

TABLE	FIELD	DEFINITION
Description	Description for the value map. You can input in multiple languages.	
Map pairs	Association field to display a list of map pairs.	

Map pair

TABLE	FIELD	DEFINITION
Map pair	Value map	Mandatory. FK to Value map.
Source value	Value of the field on the financial data set.	
Target value	Value of the field on extracted file.	

Running the Extract service

The next sections demonstrate the outcomes of running the 'Extract' service in different contexts.

Extracting data for all files

- Navigate to the 'Oracle Hyperion EPM' dataspace and select the 'Oracle Hyperion EPM Extract Configuration' dataset.
- From the 'Actions' menu of the dataset, click 'Oracle Hyperion EPM' > 'Extract'.



- Result page:



Extracting all files in group(s)

- Navigate to the 'Oracle Hyperion EPM' data space and select the 'Oracle Hyperion EPM Extract Configuration' data set.

- Select one or more group(s) in the 'File group' table. From the table's 'Actions' menu, click 'Extract' to extract files for the selected group(s). Alternatively, you can access a record in the 'File group' table and execute the service from the record's 'Actions' menu.



- Result page: All the file groups will be listed including the files under the groups. Click on the 'Download' button to download the corresponding files.



Extracting selected files

- Navigate to the 'Oracle Hyperion EPM' data space and select the 'Oracle Hyperion EPM Extract Configuration' dataset.
- Select one or more file(s) in the 'Extract file' table. From the table's 'Actions' menu, click 'Extract' to extract all selected files.



- Result page: All the files will be listed. Click on the 'Download' button to download corresponding files.



Extracting one file on a hierarchy

- Navigate to the 'Oracle Hyperion EPM' data space and select the 'Oracle Hyperion EPM Extract Configuration' data set.
- On the service list on hierarchy view of the 'Extract file' table, click on the 'Extract' service to extract the file.



- Result page: the file will be displayed on the result page. Click on the 'Download' button to download your file.



Extracting a file in 'Extract file' record view

- Navigate to the 'Oracle Hyperion EPM' data space and select the 'Oracle Hyperion EPM Extract Configuration' dataset.

- Access a record in 'Extract file' table. From the record's 'Actions' menu, click 'Extract' to extract the file.



- Result page: the file will be displayed on the result page. Click on the 'Download' button to download your file.



Configuring data mapping

- Configuration Overview
- EBX® repository data structure
 - MDM table
 - MDM field
 - Path
- Mapping definition
 - Application
 - Mapping table
 - Mapping field
 - Dimension
 - Application dimension
 - XML path
 - Version
 - Language mapping
 - Validation report on mapping data set
- Default data configuration
- Additional configurations
 - Aggregation weight display format
 - Default select all on export
 - Default language
 - Export full relationship nodes

- Default export file extension
- Export order for hierarchy
- Action for import missing value
- Flat csv version
- Export #root
- Export empty field for app setting
- Other CSV file configuration properties
- Use TIBCO EBX™ Match and Cleanse Add-on
- Application table
- Dynamically handle the hierarchy levels of Essbase rainbow file
- Switch the display of boolean fields in exported Excel file between T/F and Y/N
- Align predefined rules data to the current version of TIBCO EBX™ Rules Portfolio Add-on
- Add default configuration for 'Update dimension and field order' on import service
- Regular expression for names

Configuration Overview

The data domain called 'Administrative data' is located in dataset 'Oracle Hyperion EPM Mapping'. It encompasses all the information required to define data mapping between the EBX® and Hyperion data structures.

This allows data mapping rules to be readily adapted depending on how EBX® data structures are designed and which XML data formats need to be integrated.

The data mapping is defined in the context of each Hyperion application, such as HFM, or Essbase.

An example of data mapping values is highlighted in appendix.

It is possible to manage different versions of data mapping by using datasets. When the import and export procedures are executed, it is possible to select a dataset where the data mapping is retrieved by the add-on.

EBX® repository data structure

The descriptions below provide an overview of the table structure in the repository.

MDM table

TABLE	FIELD	DEFINITION
MDM table	Name	Name of the table in EBX® For example, "Account", "AccountHierarchy", "AccountDescription"
Path	Absolute path of the table starting with /root in the EBX® repository data structure For example, /root/Hyperion/Financial/Account	

MDM field

TABLE	FIELD	DEFINITION
MDM field	MDM table	Name of its containing table in EBX® For example, "Account"
Field name	Field name in the EBX® table For example, "Name"	
Type	Data type For example, String, Boolean.	
Path	Relative path of the field relative to the EBX® table path For example, /Name	
Is filtering	No: the field is used when exporting data Yes: the field is ignored when exporting data	
Default value	This property is only used when the 'Type' field is set to Boolean and allows you to specify a default value for the field. This value is used	

TABLE	FIELD	DEFINITION
	during import, update and creation. If a value is not declared the field is considered undefined and the system automatically gets the default value declared in the data model.	

Path

TABLE	FIELD	DEFINITION
Path	Path	Path in the EBX® repository data structure. Starts with either '/root' for an absolute path, or '/' for a relative path Example: /root/Hyperion/ Financial/Account / CalcAttribute

Mapping definition

All the tables described below are located in the group 'Mapping' in the Administrative Data domain. To complete these tables it is recommended to have mastered XML data structure.

Application

TABLE	FIELD	DEFINITION
Application	Name	Name of Hyperion application available when importing and exporting data. For example HFM, Essbase, Planning, EPMA. The list of possible applications is retrieved from the 'Application' table declared in the 'TIBCO EBX™ Add-on for Oracle Hyperion EPM' data space,

TABLE	FIELD	DEFINITION
		located in EBX® 'Administration' tab.
Has version	Allow for identifying if the application has version for import/export procedure or not. If it has version, the import/export procedure will process the version in table 'Version' as the tag 'Schema_version' on pivot XML file	

Mapping table

TABLE	FIELD	DEFINITION
Mapping table	Application	Application for which the mapping is defined For example, HFM
MDM table	Table name in the EBX® repository	
XML path	Relative path in the XML file for the table For example, / MEMBERS/MEMBER	
Dimension	Dimension on which the table is mapped. The dimension description gives its XML name (see below Table Dimension) For example, 'Account' The list of available dimensions depends on the selected application.	
XML path in Planning	Relative path in the PLANNING XML file for the table. For example, /Member	

TABLE	FIELD	DEFINITION
Table type	Enumeration value: Member, Node, Description, Reference Member: dimension. Eg. "Account" Node: hierarchy table. Eg. "AccountHierarchy" Description: description table. Eg. "AccountDescription" Reference: tables in group Financial/Reference. Eg. 'ConsolidationAccountType', 'View'	
Is extension table	Configure is the table is an extension or not	

Mapping field

TABLE	FIELD	DEFINITION
Mapping field	Application	Application for which the mapping is defined For example, HFM
Mapping table	Mapping table which includes the field For example, 'HFM-Account' (first the application name, and then the table name) The tables list depends on the selected application.	
MDM field	Field name in the EBX® repository For example, 'CalcAttribute' The fields list depends on the selected table.	
XML Path	Relative path to the field in XML file including its name For example, /AT	

TABLE	FIELD	DEFINITION
Essbase spread sheet column name	Name of the column corresponding to the field in the spreadsheet used to import and export data for Essbase. This field is not used when not importing from or exporting to Essbase.	
XML path in Planning	Relative path in the PLANNING XML file for the field For example, / Plan5PerfOrder	
Planning CSV column name	Name of the corresponding field in CSV file use only to export data for Planning. For example: "Aggregation (Flow)"	
Flat CSV name	Name of corresponding field in the CSV Flat file used to export HFM Flat CSV file. For example: "EnableCustom3Aggr"	
Flat ADS name	Name of the corresponding field in the ADS Flat file used to export the EPMA flat ADS file. For example: "SmartList"	
Field order	Place of the field in XML file within member. The order value 0 means the first position.	
Field type	Enumeration value: Name, DefaultParent, Attribute, Description, Parent, Child, Language, Description - Language, Data Storage, UDA - Name: Label. -	

TABLE	FIELD	DEFINITION
	DefaultParent: DefaultParent. - Attribute: AT. - Description: Description. - Parent: hierarchy Parent. - Child: hierarchy Child. - Language: Language - dimension Language. - Description - Language: attribute Language in tag Description. - Data Storage: Data Storage. - UDA: UDA. - AttributeICP: AT - Name = "ICPTopMember". - Attribute security class: AT - Name = "SecurityClass". - Weeks distribution member: attribute WeeksDistributionMember. - Skip value: attribute SkipValue. - Scale: attribute Scale. - Symbol: attribute Symbol.	

Dimension

TABLE	FIELD	DEFINITION
Dimension	Name	Dimension name
	Essbase sheet name	Name of the sheet corresponding to the dimension in the spreadsheet used to import and export data for Essbase. This field is not used when not importing from or exporting to Essbase.

TABLE	FIELD	DEFINITION
	XML path in Planning	Relative path in the PLANNING XML file for the dimension For example: <pre> /DIMENSIONS/ Dimension[@csversion="*" @density="*" @dimensionName="*" @name="Account" @dimensionType="*" @origname="*" @DimensionAlias="*" @comment="*"] </pre>
	XML path	Absolute path for the dimension in the XML file including its name For example, // <pre> DIMENSION[@Name="Account"] </pre>
	Dimension type	Enumeration value: Business dimension, Currencies, Application setting, Consol method, Language • Business dimension: including Account, Custom1...4, Entity, Scenario, Period, Year. • Currencies: dimension Currencies.

TABLE	FIELD	DEFINITION
		• Application setting: dimension MISC - Application setting. • Consol method: dimension MISC - Consolidation method. • Language: dimension Language.
	Is custom dimension	Allow for identifying if the dimension is a Custom one. Some internal treatment can be performed such as creation of a default record with value 'ALL'
	Flat ADS name	Name of the corresponding dimension in the ADS Flat file used to export the EPMA flat ADS file. For example: "planning_SampApp_Entity"

Application dimension

TABLE	FIELD	DEFINITION
Application Dimension	Application	Application for which the mapping is defined For example, HFM
Dimension	A dimension held by the application	
Dimension order	Place of the dimension in XML file. The order value '0' means the first position	

TABLE	FIELD	DEFINITION
Is default	Allow to identify if the dimension is an automatic import/export dimension. If 'Is default' is set 'True', then the check-box for it selection will be disabled during import and export procedure.	

XML path

TABLE	FIELD	DEFINITION
XML Path	XML Path	Path in the XML file data structure For example, /AT

Version

TABLE	FIELD	DEFINITION
Version	Version	Version in XML file.
Update time	Time of last update	
Application	Application containing this version	

Language mapping

TABLE	FIELD	DEFINITION
Language mapping	Language	The Language in XML file
EBX® locale	Enumerated value for the locale of EBX®: English, French ...	

Validation report on mapping data set

Validation reports run on the 'Oracle Hyperion EPM Mapping' data set are enriched to check the following aspects of configuration integrity:

- Duplication Paths (warning)
- Duplicated paths of MDM table (warning)
- Duplicated MDM tables and Paths of MDM fields (warning)
- Duplication of XML paths (warning)
- Path defined as a space string (warning)

To run a validation report, navigate to the 'Oracle Hyperion EPM Mapping' data set and from the 'Actions' drop-down menu select 'Validate'.



Default data configuration

The Oracle Hyperion EPM comes with a default, ready-to-use data configuration for each Hyperion application.

When necessary, some reference data values are also provided directly within the 'Financial data' domain such as the table 'Consolidation account type'.

The default data for 'Financial data' domain is provided as an archive named "ebx-addon-hmfh.ebx" (only data for Reference data). The default mapping is provided as an archive named "ebx-addon-hmfh-mapping.ebx".

Please refer to the section Installation of this user guide to get further information.

Additional configurations

The 'TIBCO EBX™ Add-on for Oracle Hyperion EPM' data set of the EBX® Administration area brings a set of additional configurations:

- 'Mail' is used to configure email information required when the execution of the import and export procedures is asynchronous. See the sections related to the import and export procedures for more information.
- 'Logging' is used to configure the data mapping logging. See Appendix - Data mapping logging for more information.
- 'Basis' and 'Path' domains are described in this section.



Aggregation weight display format

Define the export format for the field 'AggregationWeight' in the dimension Custom. This format is defined follows 'Java - Decimal Format'. Value must conform to the regular expression "?+?*?*".

Here are some examples:

VALUE	PATTERN	OUTPUT	COMMENT
123456.789	###,###.###	123,456.789	The pound sign (#) denotes a digit, the comma is a placeholder for the grouping separator, and the period is a placeholder for the decimal separator.
123456.789	###.##	123456.79	The value has three digits to the right of the decimal point, but the pattern has only two. The format method handles this by rounding up.
123.78	000000.000	000123.780	The pattern specifies leading and trailing zeros, because the 0 character is used instead of the pound sign (#).
123.00	0.#	123	The pound sign (#) at the end of the pattern denotes the digits after the decimal point of the output without trailing zeros.

Default select all on export

This is an option for the default 'select all dimension(s)' when using the export service.

Default select all on Export = 'Yes':

☐

Default select all on Export = 'No':

☐

Default language

Define the default language used when the user create a description of a dimension member.

If the 'Default language' is filled (E.g. English):

Then the add-on uses it as the default language when creating a new member description through the user interface:

☐

If the 'Default language' is empty:

Then the system uses the EBX® current language as the default language when creating a new member description through the user interface:

☐

Export full relationship nodes

Allow to export all relationship nodes declarations when the export procedure is executed with the Excel format.

Export full relationship nodes = 'Yes':

The complete node declaration (including all descendants) is exported in every node.

☐

Export full relationship node = 'No':

The complete node declaration is exported only one time. After, only the parent node is exported.

☐

Default export file extension

☐
☐

On the service 'Hyperion export', after selecting the application, the option 'Choose dimension(s) for export' requests the user to select the export file type(s). The file types can be XML, Excel 2007-2010 and CSV.

The configuration 'Default export file extension' allows to configure the default selected file types. For example, if the configuration 'Default export file extension' is as follows:

Then the result for the export UI is highlighted below:



Export order for hierarchy

This field provides two options: 'Order using Child order' and 'Same order as Data hierarchy view'. It allows for deciding the order in the hierarchy part of the export file when using the Hyperion export service.



If the option 'Order using Child order' is selected, then the order of the hierarchy part in the export file is ordered by the field 'Child order'. This field is located on the hierarchy tables in the financial data set (Ex: Account hierarchy, Custom1 hierarchy, etc.).

If the option 'Same order as Data hierarchy view' is selected, then the system will arrange the order in the export file in the same order than one displayed in the data hierarchy view.



For example, when the export procedure is executed on this hierarchy, the selection of the two options provide the results highlighted in the next pages.



The corresponding Child orders of this hierarchy.

Export order for hierarchy = 'Order using Child order':



Export order for hierarchy = 'Same order as Data hierarchy':



Action for import missing value

When importing data to EBX®, it can exist missing values. For example, the missing value for AggrWeight:

- On Excel file: the empty or null value for AggrWeight or the missing of the column AggrWeight.

- On XML file: the empty value for tag <AT Name-"AggrWeight"></AT>



In order to handle these cases of missing values, the add-on provides an option in TIBCO EBX™ Add-on for Oracle Hyperion EPM to decide the action when importing the missing value field(s) in mode 'Import'.



- 'Preserve attribute existing values'. The system will keep current value of the missing value fields.
- 'Set existing attribute values to null'. The system will overwrite current value of the missing value fields as null.

Example:

For the field 'AggrWeight' in the dimension Custom1:



Action for import missing value = 'Preserve attribute existing values'.

- When importing in mode 'Import' with the file missing value for AggrWeight, the system will automatically keep the current value for field AggrWeight.



Action for import missing value = 'Set existing attribute values to null':

- When importing in mode 'Import' with the file missing value for AggrWeight, the system will automatically overwrite the current value for field AggrWeight as null.



Flat csv version

Define field 'Version' used by the export procedure with HFM FLAT CSV file.



Export #root

Allow to use or not the '#root' in HFM FLAT CSV file when exporting data.

Export #root = 'Yes':



Export #root = 'No':



Export empty field for app setting

Decide to export the empty field(s) in table 'Application Setting' or not when using the FLAT CSV file.

Export empty field for AppSetting = 'No':

☐

Export empty field for AppSetting = 'Yes':

☐

Other CSV file configuration properties

These additional properties are available to configure the export of flat CSV file:

PROPERTY	DESCRIPTION
Header's label of default parent	Allow one to provide a label for the header of the default parent column in the flat CSV file. This property is not used in case there is no mapping for the field "Default parent" or when the value is empty.
Header's label of description	Allow one to provide a label for the header of the description column in the flat CSV. This property is not used in case there is no mapping for the field "Description" or when the value is empty.
Is export hierarchy header	Boolean value. Allow one to configure if the hierarchy headers are exported in the flat CSV file or not. 'True': The exported CSV file contains the header of hierarchy. 'False': The exported CSV file does not contain the header of hierarchy.

PROPERTY	DESCRIPTION
	This property is not used in case there is no mapping for the hierarchy fields.
Is export custom dimension order	Boolean value. Allow one to configure if the item '!CUSTOM_ORDER=Custom1, Custom2...' is exported or not. 'True': the item 'CUSTOM_ORDER' is exported in the CSV file. 'False': the item 'CUSTOM_ORDER' is not exported in the CSV file.

Examples:

- Step 1: Configure export CSV options as follows:

☐

- Step 2: Execute the export CSV file operation

☐
☐
☐

The header's label of default parent will be 'Default parent' and the header's label of description will be 'Description(s)' as configured:

☐

The hierarchy header will be removed since the configuration 'Is export hierarchy header' is set as 'No'.

☐

In case the headers contain the delimiter of export files, the add-on will raise an error:

☐

In case there is only value for the property "Header's label of description" as highlighted now:

☐

Then, the header's label of description will be display as 'Description(s)'. The header's label of default parent column will be missing since the value of "Header's label of default parent" is blank.

☐

Use TIBCO EBX™ Match and Cleanse Add-on

From the version 1.5.0, it is possible to use the TIBCO EBX® Match and Cleanse Add-on on financial data. When the EBX® Match and Cleanse Add-on is not installed in your environment then the option is no longer available.

When the 'Use TIBCO EBX™ Match and Cleanse Add-on' is set to 'True', the add-on creates a data space for the EBX Match and Cleanse Add-on with the following datasets: 'Oracle Hyperion EPM dimensions' (on which EBX Match and Cleanse Add-on functionality is available), 'Oracle Hyperion Mapping' and 'Oracle Hyperion Report'.

On the 'Oracle Hyperion EPM Dimensions' dataset, the financial tables are extended with the metadata type used by the EBX Match and Cleanse Add-on. To enable data transfer to and from a standard 'Oracle Hyperion EPM Dimensions' dataset to the one using the EBX Match and Cleanse Add-on, use one of the following services: 'Copy to use TIBCO EBX™ Match and Cleanse Add-on' and 'Copy to remove TIBCO EBX™ Match and Cleanse Add-on'. Refer 'Integration with TIBCO EBX™ Match and Cleanse Add-on' in the user guide for more information).



'ebx-addon-hmfh-include-daqa.ebx' archive is provided as the default reference data of "Oracle Hyperion EPM Dimensions" data set in "Oracle Hyperion EPM with Match and Cleanse Add-on metadata" data space.

Application table

As shown in the following image, the list of possible applications is created in the 'Administration' tab → 'TIBCO EBX® Add-on for Oracle Hyperion EPM' → 'Application' table.



As shown below, when you create a data mapping the list of applications is supplied from the 'TIBCO EBX™ Add-on for Oracle Hyperion EPM' data space.



Dynamically handle the hierarchy levels of Essbase rainbow file

Overview

The add-on dynamically handles the number of hierarchy levels exported in an Essbase rainbow file. You only need to set the default number of hierarchy levels, and the number on the exported file dynamically updates based on your input (minimum is your default number).

Handle procedure

- On the 'Basis' group under the 'Hyperion Configuration' data set of the 'Administration' data space, set the value of the 'Default minimum number of generated levels' to the desired number. In this example we use five.



- On the 'Member' type table under the Oracle Hyperion EPM Dimensions data set, create some records. For example : 3 records (A1, A11, A12) in table 'Account'.



- From the 'Actions' menu under the Oracle Hyperion EPM Dimensions data set, run the 'Export' service with input data as below:

- Application: Essbase.
- Dimension: the dimension of the table that you have just imported data (Account).
- Export type: Excel.



- Now the value of 'Generate levels dynamically' is displayed as your setting in the Hyperion Configuration data set. You can change this value here if you want.
- Click the 'Export' button to export your data. Then download the exported Excel file.
- Open the exported file. The number of hierarchy levels is 5.



Migration of the 'Mapping data set'

The 'Migrate mapping data set' service is located in the 'Oracle Hyperion EPM' data space and is available to any child data space. This service migrates the existing data mapping configuration into a version that is compatible with the new extension management integrated from the 1.3.0 version of the add-on. Any 'Mapping table' with an undefined 'Is extension table' field is automatically set to 'No'.



This service migrates the existing data mapping configuration into a version that is compatible with the new validation report integrated from the GA 1.6.0 version of the add-on. The service automatically inserts the following new data mapping declarations for the EPMA application:

- **MDM table, MDM field:** new tables and fields used for storing EPMA data.
- **Application:** new application EPMA.

- **Dimension:** new dimensions for EPMA (Currency, Alias, Version); ADS name for dimensions that can be exported into EPMA ADS file.
- **Application dimension:** define which dimensions can be exported into EPMA ADS file (Account, Custom1 as Generic, Entity, Scenario, Period, Year, Currency, Version, Alias).
- **Mapping tables:** EPMA data mapping for the tables.
- **Mapping fields:** EPMA data mapping for the fields.

Switch the display of boolean fields in exported Excel file between T/F and Y/N

Overview

This feature supports user to switch the display of boolean values between T/F and Y/N in output file of the Export service.

Changing the Boolean display format

- From the Basis group under the TIBCO EBX™ Add-on for Oracle Hyperion EPM data set of the Administration data space, set the value of the 'Boolean values display format' field to 'Y/N' or 'T/F'.



- Create records in the Account table under the Oracle Hyperion EPM Dimensions data set. Then navigate to the Actions menu and run the Export service with the following configuration.
- Application: HFM
- Dimension: Account
- Export type: Select all
- In General properties: the default value of 'Boolean values display format' is 'Y/N'.
- Download the exported file then check the result.

- ADS file



- Excel file



- XML file

☐	☐

Align predefined rules data to the current version of TIBCO EBX™ Rules Portfolio Add-on

Overview

TIBCO EBX® Rules Portfolio Add-on 1.5.0 provides a feature to display D.E.C as hierarchy. Therefore, it's needed to update the predefined data for D.E.C on EBX® Add-on for Oracle Hyperion EPM to be aligned to this version of EBX® Rules Portfolio Add-on.

- Activating trigger when creating predefined rules. (reuse the Rules trigger to create data.)
- Adding D.E.C data for data set into predefined data of D.E.C.
- Remove simple expression data in predefined rules to avoid creating unused data since simple expression java classes are not supported.

Add default configuration for 'Update dimension and field order' on import service

Overview

Currently, we provide a new option in the Configuration section that allows users to set the default value themselves then they can modify it on Import configuration screen of 'Import' service



Regular expression for names

Overview

This feature allows users to add naming validation rule for name fields of financial dimension records. The following sections describe the steps to apply the rule.

Set rule in the configuration

To set a validation rule, go to Administration -> TIBCO EBX® Add-on for Oracle Hyperion EPM -> Basis.



Validation rule in effect

This rule affects the following tables: Account, Currency, Custom1, Custom2, Custom3, Custom4, Entity, Period, Consolidation account type, Consolidation method, Application setting, ICP, Scenario, Year.

When users create a record to one of above tables, if its name does not match the validation rule, a warning will be displayed, and they won't be able to save until the name conforms to the rule. For example, with this rule set:



Results in the following screen:



Import and export procedures

- Data formats
- Information order when exporting data
 - Nodes hierarchy order
 - Dimensions and fields order
 - Records order
- Import procedure
 - Choose data set mapping
 - Import with deletion or not
 - Update dimension and field order
 - Using TIBCO EBX® Add-on for Oracle Hyperion EPM - Logging
 - Automatic Hyperion application detection
 - Selecting a dimension for the EBX® hierarchical view
 - Choose delimiter (for ADS only)
 - Action for import missing value
 - Run in background
 - Import result
- Export procedure
 - Choose data set mapping

- Export type
- Using TIBCO EBX® Add-on for Oracle Hyperion EPM - Logging
- Hyperion application selection
- Advanced properties
- Run in background
- Export result file
- Import and export when using extension data models
 - Configuring a mapping extension
 - Import file when using extension data models
 - Export file when using extension data models
 - Displaying extension information
- Export from multiple nodes on the hierarchy view
 - Overview
 - Export nodes

Data formats

This table highlights the different formats of data accepted for import and export procedures.

APPLICATION	PROCEDURE (FORMAT)	SOURCE -> TARGET
HFM	Import (XML, XLSX)	XML (id=hfm)-> EBX® XLS/ XLSX->XML (id=hfm)->EBX®
Export (XML, XLSX, CSV)	EBX®-> XML (id=hfm) EBX®-> XML (id=hfm)->XLSX EBX®-> XML (id=hfm)-> CSV	
Essbase	Import (XML, XLSX)	XML (id=essbase)->EBX® XLS/XLSX->XML (id=essbase)->EBX®
Export (XML, XLSX)	EBX®->XML (id=essbase) EBX®->XML (id=essbase)-> XLSX	

APPLICATION	PROCEDURE (FORMAT)	SOURCE -> TARGET
Planning	Import (XML)	XML (planning) -> XML (id=planning)->EBX®
Export (XML, CSV)	EBX® -> XML (id=planning) -> XML(planning) EBX® -> XML (id=planning) -> CSV(planning)	
EPMA	Import (XML, ADS)	XML (id=epma) -> EBX® ADS -> XML (id=epma) -> EBX®
Export (XML, ADS)	EBX® -> XML (id=epma) EBX® -> XML (id=epma) -> ADS	

Note: XML (id= *application code*) is the HFM XML data schema that the add-on reuses as an internal pivot format.

Information order when exporting data

With Hyperion products, it is important to ensure that order of information in export file follows some key principles that are described in this section.

Nodes hierarchy order

The order of nodes in the EBX® data hierarchies follows one kept during the import procedure. The order in the export file follows one of the data hierarchy.

Dimensions and fields order

The order of dimensions and fields are saved in the dataset mapping information during the import procedure: see 'Mapping field' and 'Application dimension' tables, fields 'Dimension order' and 'Field order' respectively.

The order generated by the export procedure follows the order saved during the import. Consequently, the dataset used to get the data mapping configuration during the import time should be the same of one used at export time.

When the order value is 'null' for a field, it means that the field is located at the end of the list. For example, the fields 'Default parent' and 'Description' are systematically located in last position.

Records order

The order of fields and dimensions are saved in the dataset mapping information during the import procedure.

The export order is built by following the nodes data hierarchy order.

Example:

In EBX® data hierarchy for Account dimension:

- LEGAL
 - AE07
 - AE0700
 - AE08
 - AE0800
 - AE0700

Then the order of members in the XML export file is as follows:

1. LEGAL
2. AE07
3. AE0700
4. AE08
5. AE0800

The duplicate one 'AE0700' is put only one time in the XML export file.

Import procedure

This procedure imports data from a Hyperion application into the EBX® repository.

In the user interface, it can be accessed from the navigation pane by selecting **Oracle Hyperion EPM** then using the **Services** menu.



Choose data set mapping

Select the dataset to use for the data mapping configuration. It is possible to configure different data mappings and then selecting one when importing data.

The add-on automatically ignores this step if there is only one dataset mapping.

Import with deletion or not

The following options are available when importing data from Hyperion into the EBX® repository:

- *Import* : does not delete existing data. Only create and update operations are enforced against the current EBX® repository.
- *Delete all before import* : EBX® tables related to dimension with relations are cleaned before importing data. It means that those tables are not removed: ICP, Security Class, Currency, View, Alias.

Update dimension and field order

When importing data, you can decide to keep the ordering of dimensions and fields based on the imported file and reuse this order when re-exporting the file. This information is stored as the field 'Field order' in the table 'Mapping Field' and the field 'Dimension order' in the table 'Application Dimension' in the Administrative Data domain.

These fields can be manually changed to set a different order for an exported file. During normal use of the add-on, this modification is not required. In most situations, the order of dimensions and fields provided by the imported file can be reused for the exported file.

Using TIBCO EBX® Add-on for Oracle Hyperion EPM - Logging

When this option is activated, the add-on logs the data mapping execution to facilitate debugging operations when data mapping configuration is on failure (see Appendix - Data mapping logging).

Automatic Hyperion application detection

After selecting the file to import, the system automatically identifies the source Hyperion application. If this identification fails, the import procedure stops. This could occur if the application header in the file is not an expected application name, the XML encoding is incorrect, or the Excel file is not recognized.

Selecting a dimension for the EBX® hierarchical view

Based on the Hyperion application detected, the add-on displays all possible dimensions that can be imported. One or multiple dimensions may be selected. Special dimensions that are driven by the main dimensions are not displayed, but are managed automatically by the add-on. This is the case for the dimensions ICP, Security Class, Currency, View, Alias.

Choose delimiter (for ADS only)

This option only displays when importing an ADS file. In version 1.6.0 only the pipe (|) delimiter is supported.

Action for import missing value

When importing data to EBX®, missing values can be managed in different ways. The possible choices are displayed depending on the 'Action for import missing value' configuration in the 'TIBCO EBX® Add-on for Oracle Hyperion EPM' data space (located in the EBX® Administration tab).

Run in background

The option 'Run in background' allows to launch the import procedure in asynchronous mode. The user is notified by email with the result as highlighted below.



Please refer to the installation section to get further information about email configuration.

Import result

After running an import, a report UI is provided showing the number of creations and updated records for each table. The full import result is saved in a dedicated data set named 'Oracle Hyperion EPM Report'.

Export procedure

This procedure exports data from the EBX® repository into Hyperion applications.

In the user interface, it can be accessed from the navigation pane by selecting **Oracle Hyperion EPM** then using the **Services** menu.



Choose data set mapping

Select the dataset to use for the data mapping configuration. It is possible to configure different data mappings and then selecting one when exporting data.

The add-on automatically ignores this step if there is only one dataset mapping.

Export type

The following options are available when exporting from the EBX® repository into Hyperion:

- *Export*: export all data.
- *Export with filtering*: export all data except fields that have an 'is filtering' value of 'true' in the table 'MDM field' of the Administrative Data domain. The following fields cannot be filtered: primary key (name), default parent, description.

Using TIBCO EBX® Add-on for Oracle Hyperion EPM - Logging

When this option is activated, the add-on logs the data mapping execution to facilitate debugging operations when data mapping configuration is on failure (see Appendix - Data mapping logging).

Hyperion application selection

The user selects one Hyperion application at a time for which to perform an export. When selecting an application, such as HFM or Essbase, the user can choose which dimensions of the application to export to.

When several export file formats are possible for a given application, the user can select one or multiple targeted formats. The file formats that can be exported to depend on the destination application.

Advanced properties

The available choices depend on how the 'TIBCO EBX® Add-on for Oracle Hyperion EPM' data space (located under the 'Administration' tab) is configured. Default values are retrieved from this configuration.

When you use the ADS export, specific choices are available to drive data mapping:

- Flat ADS version: sets the ADS file version (default value '1.0').
- Selection of the sections to export into the ADS file. Three mandatory sections are systematically selected and cannot be removed: 'Dimensions', 'DimensionAssociations', and 'Hierarchies'. The 'Members' and 'PropertyArray' sections are optional.

- When you select one or more additional sections and click 'Export', the system moves to the 'Export - Preference' screen (see details in Appendix - Migration and Back-up procedures). This screen allows you to set the compatible section for fields collected from the 'Mapping field' table in the 'Oracle Hyperion EPM Mapping' data set. These fields are the fields mapped for 'Member' and 'Description' tables. The fields with the field types 'Name' or 'Description - Language' are not displayed in this list.

Run in background

The option 'Run in background' allows to launch the export procedure in asynchronous mode. The user is notified by email with the result (see figure below).



Please refer to the installation section to get more information about email addresses configuration.

Export result file

For each application selected for export, a downloadable file is generated using the naming convention "<data set name>-< applicationName>-<date>-<time>".

For the Essbase application, only the dimensions themselves are exported. Other tables related to the configuration, such as 'Currency', 'Appsetting', and 'Consolmethod' are not exported because the Excel template does not handle them. The ability to export these tables may be added in future releases of the add-on.

The full export result is saved in a dedicated data set named 'Oracle Hyperion EPM Report'.

Import and export when using extension data models

Configuring a mapping extension

Step 1 : Create a new data set corresponding to your extended data model. This model must declare the tables and foreign keys that allow the system to integrate it with the core EPM financial data. Please refer to the 'sampleExtensionTemplate.xsd' sample.

For example, the 'Account' dimension can be extended by three tables in the extended data set: 'Account extension', 'Account hierarchy extension' and 'Account setting extension'.

A table used as an extension must have an atomic primary key that is configured as a foreign key. The foreign key points to the related main table in the financial data set.

The extended data set could be any data set in the 'EBX® Add-on for Oracle Hyperion EPM' data space except any previously created by the add-on: 'Oracle Hyperion EPM Dimensions', 'Oracle Hyperion EPM Mapping' and 'Oracle Hyperion EPM Report' and their potential children.

Data model extension is possible for the 'Member', 'Hierarchy' and 'Setting' tables. The 'Description' table cannot be extended.

SPECIAL NOTATION:	
☐	The extended data set cannot be configured to link to a dedicated Oracle Hyperion EPM financial data set. The link is declared in your extended data set at the modeling level. If you need to handle different extended data sets to different Oracle Hyperion EPM financial data set, you must duplicate the data model and configure the foreign keys accordingly. In such case, you will no longer benefit from the inheritance features for the extended data model.

Step 2: Configure the MDM table(s) and MDM field(s) in the Oracle Hyperion EPM Mapping as illustrated in the following figure.



Step 3 : Configure the data mapping.



Import file when using extension data models

In order to import a file containing extended information, it is necessary to enrich the 'Mapping table' with the configuration of the extended tables and fields. The property 'Is Extension Table' is then set to 'True'.



In the properties available at the level of the 'Financial data' group, the 'Data set extension' is selected to import the data related to the extended tables and fields.



During the import procedure, select the data set mapping that contains the mapping for the extended tables and fields, as illustrated below:



Export file when using extension data models

During the export procedure, the selected 'Data set extension' is used. As explained for the import procedure (see previous section), it is defined as a property at the level of the 'Financial data' group.

If the 'Data set extension' is 'not defined' then the export of extended tables and fields is not applied.

Displaying extension information

The extension data is displayed through the 'Extension' tab as illustrated below.



SPECIAL NOTATION KEY:	
	The ability to change the 'Extension' tab label is not yet implemented.

Export from multiple nodes on the hierarchy view

Overview

This service allows user to export only nodes that they had selected themselves, instead of all nodes or all child of selected nodes.

The add-on only permits export of selected nodes. Additionally, records without relationships cannot be exported. When you try to run the service on these type of records, the following error message displays as below.



Export nodes

- Navigate to the Account table under the Oracle Hyperion EPM Dimensions data set, select the desired records (except those in the 'Active unset' group).



- From the 'Actions' menu, run the 'Export from selected nodes' service. The system will display the 'Hyperion Export - Choose data set mapping' screen.





- Select the data set mapping that contains the mapping data you would like to use for this export procedure, then click on the 'Configure export' button.
- On the Export configuration page, select one application, then click on the 'Dimension' button.
- The system will display the Choose dimension page. On this screen, you can select file types and customize information necessary for each type. After that, click on the 'Export' button.



- The service executes and returns the Export result to allow you to download your exported files.

Comparing dimensions

- [Overview of comparison](#)
- [Creating a comparison](#)
- [Expand/collapse one level for all nodes](#)
 - [Overview](#)
 - [Expand/Collapse procedure](#)

Overview of comparison

To ensure conformity with business objectives and forecast the impact of changing dimensions, you can use the 'Compare' service to quickly compare past, present, and future versions of dimensions. You can compare dimensions in up-to-date data spaces and those from previously taken snapshots. Specifically, you can compare between dimensions in:

- Snapshots
- A snapshot and a data space
- Data spaces
- Data sets (including a data set and its child)

When viewing a comparison, the source dimension displays on the left and the target on the right. If you expand a node in either side, its counterpart expands. As shown below, icons call out how nodes differ between source and target:

- A green '+' icon represents an addition.
- A red 'X' icon represents a removal.

- An orange '!' icon represents a modification to the node's hierarchy position. This notice displays when a node's 'Child order' property value differs between the source and target. Keep in mind that this difference may not always affect the order in which the nodes display. Using the image below as an example, the 'a21' and 'a22' nodes in the source dimension may have their 'Child order' properties set to '1' and '2', respectively. As shown in the upper image, this causes 'a21' to display above 'a22'. In the target dimension the node's respective property values may be changed to '5' and '4', which changes the node order.



- The 'i' button only displays for nodes whose 'Child order' property differs between source and target. As shown below, clicking the icon opens a pop-up outlining these differences.



Creating a comparison

To compare a source and target's versions of a dimension:

- Navigate to the Oracle Hyperion EPM data space and select the data set containing the dimension you want to use as the source for this comparison.
- From the 'Actions' menu, select 'Oracle Hyperion EPM' → 'Compare'. Note that after executing the 'Compare' service, you cannot change the 'Source data set' unless you exit and open a different data set.
- Specify the 'Target data set' using the supplied fields.



- Under the 'Dimension information' heading, select the 'Application' and then the 'Dimension' to compare. The selected application determines which dimensions you can select from.
- After clicking 'Next', the dimension displays as contained in each data set with the source on the left and target on the right. The 'Display mode' drop-down list contains the following options to adjust dimension display:
 - Display all
 - Display all differences only
 - Display deleted nodes only
 - Display new nodes only
 - Display differences in node order only



Expand/collapse one level for all nodes

Overview

This feature support users to expand/collapse one level by one level for all nodes which has the same level.

Expand/Collapse procedure

- From the Oracle Hyperion EPM Dimensions data set, input your own data then click on the 'Next' button to view the comparison.



- On the 'Comparing dimension' screen, click on either the 'Expand' button to view the records comparison at a higher level, or the 'Collapse' button to view the records comparison at a lower level.
- After finishing, you click on the 'Export to Excel' button to export the comparison's result to an Excel file.

Mapping data generation procedures

- [Generate mapping for custom dimension](#)
 - [Overview of custom dimension and its mapping data generation](#)
 - [Generate mapping for custom dimension procedure](#)
- [Generate extension mapping](#)
 - [Overview of extension mapping data generation](#)
 - [Generate extension mapping procedure](#)
- [Generate mapping data](#)
 - [Overview of mapping data generation](#)
 - [Generate mapping data procedure](#)

Generate mapping for custom dimension

Overview of custom dimension and its mapping data generation

The EBX® Add-on for Oracle Hyperion EPM allows users to create a custom data set. New custom dimensions can be created by adding a custom dimension group to each custom dimension, then create mapping data for custom dimension. For more information about what custom dimension is, and how to add a new custom dimension, see the Appendix Adding custom dimension.

The 'Generate mapping for custom dimension' service supports automatic mapping data for extra custom dimension. This service facilitates you to generate mapping data for all custom dimensions and all applications, and the following actions:

- Quickly creating a child mapping data set with unique name and making it the target mapping data set that you want to insert new mapping data.
- Select a target dimension. The service will duplicate and customize its mapping data to adapt it to your custom dimension. This will save your time rather than manual work.
- Select or deselect custom dimensions/applications. You may choose to generate mapping data for every custom dimensions and applications at the same time, or one by one.

After complete running the service, mapping data can be inserted into the target mapping data set that had been selected before. When viewing the result page, there will be a table that statistically analyzes the number of records that has been inserted into each table, such as:

- MDM table
- MDM field
- Mapping table
- Mapping field
- Path
- XML path

Generate mapping for custom dimension procedure

To completely generate mapping data for an application and dimension:

- Navigate to the Oracle Hyperion EPM dataspace and select the custom data set.
- From the 'Actions' menu, select 'Oracle Hyperion EPM' → 'Generates mapping extra custom'.



- Specify the target mapping data set. If you want to create a new child mapping data set under current mapping data set and make it as the target, click on the 'Create child data set' button. Note that after doing this, you are unable to change the 'Parent data set' unless you click on the 'Cancel' button on the popup frame and change it right then. Next, you should input unique name for the child mapping data set, specify its 'Owner' and 'Label and description' then click on the 'Create' button to save.



- Specify the target dimension you want to you to clone mapping data. Note that the target dimension must have the mapping data. Otherwise, service produces no results.
- Select the custom dimension and application that you would like to generate mapping data for. You may select some, or all.



- When you are finished with your selections, click the 'Generate' button to begin generating and inserting mapping data into the selected mapping data set.
- After finishing, the service displays a screen to show you the number of records that had been inserted/ modified in your mapping data set. You can quickly check the result by clicking on the 'Preview' button on the right side of the mapping data set's name.



Generate extension mapping

Overview of extension mapping data generation

For more information about what extension field is, see Appendix/Extension. The add-on allows you to automatically create mapping data for them by using the service 'Generate mapping extension'. You can run this service once to generate mapping data for each extension field with all applications. Additionally, running this service allows you to:

- Quickly creating a child mapping data set with a unique name and make it the target mapping data set for insertion of new mapping data.
- Select or deselect applications. You can save time by choosing to generate mapping data for every application at the same time, or one by one.

After running the service, the add-on inserts mapping data into the previously selected target mapping data set. When viewing the result page, you can click the 'Preview' button to see the changes in the mapping data set.

Generate extension mapping procedure

To generate mapping data for an application and dimension:

- Navigate to the Oracle Hyperion EPM data space and select the Referenced table of the Extension field that had been setup in the Extension Data model during creation of the extension field. Prior to running the service, make sure that the referenced table is mapped with your selected applications, or the service will not execute properly. The add-on adds the extension field to this table after running the service.
- From the table's 'Actions' menu, select 'Oracle Hyperion EPM' → 'Generate extension mapping'.



- Specify the target mapping data set. If you want to create a new child mapping data set under the current mapping data set and make it the target, click on the 'Create child data set' button. Next, you should enter a unique name for the child mapping data set, specify its 'Owner' and 'Label and description'. Before clicking 'Create', you should note that after this action completes, you can no longer change the 'Parent data set'. So, be sure this is the data set you want to use, or click 'Cancel' to use a new parent. Click the 'Create' button to save.



- Specify the extension data set that you would like to generate mapping data for. Then click the 'Choose applications' button.
- Select the applications you want to map with the extension field. After clicking 'Generate' the add-on generates and inserts the mapping data.



- After finishing, a results screen displays. You can click the 'Preview' button to see more details.



Generate mapping data

Overview of mapping data generation

If you have administrative access, you can use the 'Generate mapping data' service to quickly generate and customize the mapping data for dimensions and applications. With the 'Generate mapping data' service, you can:

- Quickly create a child mapping data set with a unique name and make it the target mapping data set for insertion of new mapping data.

- Select the application and dimension required to create mapping data. If they are not mapped with each other, you can define this mapping from here.
- Select or deselect only the MDM tables & MDM fields of the selected dimension that are required to create mapping data, which enables faster service execution.
- Decide whether to keep the default mapping data by the TIBCO EBX® Add-on for Oracle Hyperion EPM, by the mapped MDM tables & MDM fields, or customize them.

After running the service, the add-on inserts the mapping data into the previously selected target mapping data set. When viewing the result page, there will be a table that statistically analyzes the number of records that has been inserted into each table, such as:

- MDM table
- MDM field
- Mapping table
- Mapping field
- Path
- XML path

After the service runs, you can click the 'Preview' button to see execution results.

Generate mapping data procedure

To generate mapping data for an application and dimension:

- Navigate to the Oracle Hyperion EPM data space and select the data set that contains the dimension you want to generate mapping data for.
- From the 'Actions' menu, select 'Oracle Hyperion EPM' → 'Generates mapping data'.
- Select the target mapping data set. If you want to create a new child mapping data set under current mapping data set and make it the target, click on the 'Create child data set' button. Note that after doing this, you are unable to change the 'Parent data set' unless you click on the 'Cancel' button on the pop-up frame and change it. You should enter a unique name for the child mapping data set, specify its 'Owner' and 'Label and description', then click on the 'Create' button to save your action's result.



- Specify the application. The list of available dimensions changes depending on your selection. If you want to add a dimension that is not available yet, click the 'Preview' button to display the

'Application dimension' table. From this table you can map your dimension with your application.



- After finishing, click the 'Next' button to start mapping tables. The service displays dimension's list of 'MDM tables' whether or not they have been mapped. You may select, deselect or change the 'Table type' of any 'MDM table', but note that you have to select at least one 'MDM table' to continue the service. Then click the 'Next' button.



- Now you can customize 'Mapping table' record information for the 'MDM tables' that you previously selected. The information includes 'XML path' and 'XML path in Planning'. With 'MDM tables' that had been already mapped, there will be an 'Overwrite' button in the first column of each row, which allows you to customize the mapped data if necessary. For the 'MDM tables' that had not been mapped yet, its row will display the default data in the TIBCO EBX® Add-on for Oracle Hyperion EPM, and this is always customizable for you without clicking any button. After finishing, click on the 'Next' button to start mapping fields.
 - Enable mapping data customization by clicking the 'Overwrite' button in the first column. As highlighted below, the 'Wizard' button displays next to editable fields.



- Click the 'Wizard' button to set the field's default mapping data.



- Repeat steps 5 and 6 for 'MDM fields' of the dimension: select, deselect, or change the 'Field type', click on the 'Next' button to customize other information of the 'Mapping field' record such as: Essbase spreadsheet column name, XML path in Planning, Planning CSV column name, Flat CSV name, Flat ADS name. After finishing, click on the 'Generate' button, the service will execute its process, start inserting your customized mapping data into the selected mapping data set. One note for this step is that you may deselect all 'MDM field' when you do not need to generate mapping data for them, and the service will execute its process right then, without displaying the 'Mapping field' screen.



- After finishing, the service will display the 'Generate mapping result' screen to show you the number of records that had been inserted/modified in the mapping data set. You are now able to quickly check these changes by clicking on the 'Preview' button beside the mapping data set's name.



Report of import and export execution

The "Oracle Hyperion EPM Report" data set stores all import and export results (refer to appendix). The report can log an error when the data mapping or the Hyperion application file is not correct.

Business rules

Business rules can be implemented with the EBX® features (triggers, services development). Moreover, based on the native EBX® behavioral, many data quality issues are automatically addressed, such as problems with integrity constraints in foreign keys, incorrect data formats, and orphaned records.

SPECIAL NOTATION:	
	From the version EBX® Add-on for Oracle Hyperion EPM 1.5.0, the add-on can be integrated with the EBX® Rules Portfolio Add-on. This add-on is used to enforce a transparent management of the business and permission rules applied to financial data (rules portfolio, rules configuration, traceability of the rules execution). Please refer to the related appendix for more information.

API and software resources

- [API](#)
- [Paths configuration](#)
 - [Table name for service return](#)
 - [Path of alias table](#)
 - [Path of field alias/language](#)
 - [Path of field active](#)
 - [Path of field child order](#)
 - [Path of field default data set mapping](#)
 - [Path of field data set extension](#)

API

It is possible to integrate Import/Export Oracle® Hyperion EPM data in programmatic services via the Java API. See the javadoc provided with the add-on directly.

Paths configuration

The 'Path configuration' is available in the dataset TIBCO EBX™ Add-on for Oracle Hyperion EPM, located in the Administration area of EBX®. The paths of fields are used for the software development.

☐

Table name for service return

This field is used to declare the table which the import service will return to. The name stems from the table 'MDM Table' in the dataset Mapping.

☐

Path of alias table

☐

Path of field alias/language

☐

Path of field active

☐

Path of field child order

☐

Path of field default data set mapping

☐

Path of field data set extension

This field is used to declare the table which the import service will return to. The name stems from the table 'MDM Table' in the dataset Mapping.

☐

File format

- XML
- Excel for Essbase
- ADS

XML

The XML format used is defined by the HFM application. The add-on reuses this format for Essbase and Planning data values as an internal pivot format.

For Planning XML import and export, the add-on transforms Planning XML into the new HFM XML structure (see diagram below).



Excel for Essbase

An Excel template must be used to export data from Essbase and import data into EBX®. The same template is used to generate data in a format readable by Essbase.

In this template:

- Each dimension is managed as a sheet.
- Each row is a member.
- Columns from 'Gen 1' to 'Gen 20' are used to declared relationships between a parent and its children.

An example of this spreadsheet template is provided with the add-on. Also see the Appendix - Excel Essbase format and use for a full description of the template.

The Essbase user must configure the reporting tool in Essbase to generate this Excel file.

ADS

An ADS template is used to export data from EPMA and import it into EBX®. The same template is used to generate data in a format readable by EPMA.

In this template:

- Dimension settings are managed in one specified section: !Section=Dimensions.
- Associations between dimensions are managed in section: !Section=DimensionAssociations

- Each dimension's hierarchy is managed in one compatible hierarchy section. For example, the 'Account' dimension's hierarchy information is managed in section:
Hierarchies=planning_SampApp_Account.
- Depending on your environment, there are three ways to manage the hierarchy node information on each dimension:
 - In only the hierarchy section
 - In two sections:
 - Hierarchies: to manage the hierarchy information
 - Members: to manage the node information
 - In three sections:
 - Hierarchies: to manage the hierarchy information
 - Members: to manage the node information
 - PropertyArray: to manage specific node information

Please refer to the appendix for a full description of the template.

Error management

- [Internal errors raised by import and export procedures](#)
- [Data integrity failures](#)

Internal errors raised by import and export procedures

When an error occurs during the import or export procedures, all modifications are cancelled and data is reverted to the state before the process was launched.

The only situation where the process continues executing even if an error occurs is when a dimension is not recognized. In this case, the procedure carries on with the next dimension.

Data integrity failures

When data integrity failures occur, the add-on leaves data validation to EBX®. For instance, if a relation is broken between two dimensions, the import procedure carries on and EBX® logs an error in the validation report.

Data staging

Data models of the domains Financial data and Administrative data are provided with the add-on. It means that the creation of data spaces and data sets are managed by core EBX® features.

The user can then import data into a data set in a data space, then create an inherited data space to allow users to query and update dimensions and hierarchies. Comparisons between initial data values and modification can be performed with EBX®. Export procedures can be performed based on a specific data set in a data space depending on how workflows need to be organized.

Below is an example using data sets to define several applications in different Hyperion products and a quick overview of a comparison of an account from the Essbase ASO and Essbase BSO applications.



Many uses cases can be built depending on a company's needs. For example, when a hierarchy for a same dimension has differences between two applications, it is possible to create a parent data set and two child data set. Each data set inherits from hierarchy relationships defined in the parent, and each child data set can occult irrelevant relationships or created new ones unknown from the other application. When exporting data, occulted relationships are ignored.



Installation

- [Setting archives](#)
- [Email configuration](#)

Setting archives

Six default archives are provided with the add-on:

- 'ebx-addon-hmfh.ebx' for the Financial dimension data set (data for Reference tables).
- 'ebx-addon-hmfh-mapping.ebx' for the Mapping data set (data mapping).
- 'ebx-addon-hmfh-configuration.ebx' for the TIBCO EBX® Add-on for Oracle Hyperion EPM data set.
- 'ebx-addon-hmfh-logging.ebx' for the Hyperion Logging data set.
- 'ebx-addon-hmfh-report.ebx' for the Hyperion Report data set.
- 'ebx-addon-hmfh-include-daqa.ebx' for the Financial dimension data set integrated metadata of the TIBCO EBX® Match and Cleanse Add-on (data for Reference tables).

To import archives for Financial dimension data set and Mapping data set:

- a. Copy the archives 'ebx-addon-hmfh.ebx' and 'ebx-addon-hmfh-mapping.ebx' into the archives directory. See Administration Guide > Technical administration > Repository Administration in the main EBX® documentation for more information about the archives directory.
- b. Navigate to the Data Spaces area then import the archives into the data space "Oracle Hyperion EPM". Click Actions -> Import archive.
- c. Select the archive to import:
 - 'ebx-addon-hmfh.ebx' contains data for the Financial data set (only data for Reference data).
 - 'ebx-addon-hmfh-mapping.ebx' contains data for the Mapping data set.
 - 'ebx-addon-hmfh-report.ebx' contains data for the Report data set.

To import archives for TIBCO EBX® Add-on for Oracle Hyperion EPM dimension data set and Hyperion Logging data set:

- a. Copy the archives 'ebx-addon-hmfh-configuration.ebx' and 'ebx-addon-hmfh-logging.ebx' into the archives directory. See Administration Guide > Technical administration > Repository Administration in the main EBX® documentation for more information about the archives directory.
- b. Navigate to the Administration area then TIBCO EBX® Add-on for Oracle Hyperion EPM data set or Hyperion Logging data set. Click Actions -> Archives -> Import
- c. Select the archive to import:
 - 'ebx-addon-hmfh-configuration.ebx' for the TIBCO EBX® Add-on for Oracle Hyperion EPM data set.
 - 'ebx-addon-hmfh-logging.ebx' for the Hyperion Logging data set.

To import archives for the Financial dimension data set integrated in the EBX® Match and Cleanse Add-on, the steps are the same as the steps to import archives for Financial dimension.

Remember to create the 'Oracle Hyperion EPM with Match and Cleanse Add-on metadata' data space by setting 'Use TIBCO EBX® Match and Cleanse Add-on' to 'Yes' in the 'Hyperion Configuration' data set.

During the server start-up, the add-on tries to create a new dataspace for itself as a child of the Reference data space, if one does not already exist. This dataspace creation uses the standard EBX® procedure. Thus, if the repository has existing fatal validation errors in any dataset, this step

will fail. Furthermore, any dataset belonging to the reference dataspace will also be present in the add-on's dataspace.

When a new Financial dataset in another dataspace is created with 'ebx-addon-hmfh.xsd', the mapping dataset and report dataset in this dataspace must also be created with the data models delivered with the add-on, namely 'ebx-addon-hmfh-mapping.xsd' and 'ebx-addon-hmfh-report.xsd' respectively.

Email configuration

To make the option 'run in background' available for the import and export procedures, it is necessary to configure the email addresses used for sending the results as described below. This 'TIBCO EBX® Add-on for Oracle Hyperion EPM' screen is available in the Administration area of EBX®.



The technical and security parameters associated with this email account are configured in the ebx.properties configuration file (refer to EBX® documentation).

It is possible to define a localized content template for each import and export procedures for the locales configured in EBX®.



The email address used as the recipient of the notifications is the one attached to the user session, as defined in the user directory (refer to EBX® documentation).

Appendix

- [Excel Essbase format and use](#)
 - [Identification of Essbase application](#)
 - [Header structure](#)
 - [Root member structure](#)
 - [Hierarchy structure](#)
 - [Alias columns](#)
 - [Examples of invalid spreadsheet format](#)
 - [Translation into Excel and XML format](#)
 - [EBX® data structure for Essbase](#)
 - [Data mapping configuration for Essbase](#)

- EPMA ADS format and use
 - Identification of EPMA application
 - ADS structure
 - Mapping for ADS name
 - Mapping for sections on 'Export - Preference' screen
- Samples
- Administrative data
 - Logical data model
 - Declaration of the MDM data structure
 - Table mapping
 - Field mapping
 - Dimension mapping
 - Language mapping
- Data mapping logging
 - Logging Overview
 - Configuration
 - Data structure to apply the logging
- Import and export report
 - Report usage and location
 - Data architecture used to log the import and export report
 - Import result
 - Import history
 - Export result
 - Export history
 - Reference data
 - Purge of the report data
- Migration & Back-up procedures
 - Migration requirements

- Backup
- Restore
- How to upgrade version n to version n+1
- How to upgrade from one server to another
- Special case of upgrading extra custom to version 1.6.0
- Special cases in restoring
- Changes management
 - Customization options
 - Change on the mapping dataset
 - Change on the Hyperion financial data structure
 - Inclusion of the Hyperion data model
 - Changes applied on the permission
 - Changes applied on the data hierarchy views
- Adding custom dimension
- Extension
 - Create an extension model
 - Mapping for extension model
- Integration with TIBCO EBX® Match and Cleanse Add-on
 - Configuration
 - Data transfer to the datasets using the EBX Match and Cleanse Add-on and opposite direction
- Integration with TIBCO EBX® Rules Portfolio Add-on
 - Migration from Hyperion rules to EBX® Rules Portfolio Add-on
 - Configuration
 - Creation of pre-defined rules
- Known limitations
 - Error management
 - Migration procedures

Excel Essbase format and use

Identification of Essbase application

☐

Header structure

☐

Root member structure

☐

Hierarchy structure

☐

Alias columns

☐

Examples of invalid spreadsheet format

☐
☐
☐

Translation into Excel and XML format

Based on the input Essbase spreadsheet file, the Essbase import procedure generates an XML file with the same format used by HFM application, with following adjustments:

- PRODUCT_ID is set to ESSBASE.
- XML contains only Dimension. It doesn't contain any of the following tags: CURRENCIES, MISC, LANGUAGES, SCHEMA_VERSION, REPORT.

When using the Essbase export procedure in EBX®, an XML file is first generated based on HFM format. This XML file can contain the above tags, but the process to convert to an Essbase spreadsheet does not take this information into account. Then, the Essbase spreadsheet is generated automatically from this XML file.

EBX® data structure for Essbase

Essbase group of fields in each dimension

For each dimension table, a group of fields is included at the end of the data structure. It contains all the Essbase fields. See the example above.



Relationship field

The field 'Data Storage' is added to each hierarchy table. Its value depends on the relation between two members. See the example above for the 'Account data' hierarchy.



Data mapping configuration for Essbase

Essbase and HFM does not share the same data mapping configuration.

In order to import data from an Essbase XML file into EBX®, a specific data mapping configuration must be enforced:

- Add a new application named "ESSBASE" in 'Application'.
- Add a sheet name corresponding to each dimension in the table 'Dimension'.
- Add a mapping in Application > Dimension to map the dimension with the Essbase application.
- Add a data mapping to the table 'MDM table' to map the table with a path.
- Add a data mapping to the table 'MDM field' to map the Essbase field with a path.
- Add a data mapping to the table 'Mapping table' to map the Essbase application with the dimension table.
- Add a data mapping to the table 'Mapping field' to map the Essbase application to the dimension table and the Essbase field. In this table, enter the header of each property column in the spreadsheet.

There are some fields in the add-on that must be completed:

- The field 'Name' in tables 'Account', 'Custom1', 'Custom2',...
- The child field in the hierarchy table of 'Account', 'Custom1', 'Custom2',...

EPMA ADS format and use

Identification of EPMA application



ADS structure



Mapping for ADS name

Dimension mapping: Depending on the DimensionClass, you can get the ADS name of the dimension in the ADS file.



Table mapping and Field mapping: This is the same as other applications. You need to declare the ADS name for field mapping when defining the mapping field.

- Mapping table: define mapping for all tables of each dimension.



- Mapping field: define ADS name for all fields to import/export.



Dimension association: In version 1.6.0 of the add-on, the dimension association is imported into the 'TIBCO EBX® Add-on for Oracle Hyperion EPM' data space when importing. On export, the system automatically gets that dimension association to include in the exported ADS file.

Mapping for sections on 'Export - Preference' screen

When you select more than just the default sections to export, you are automatically taken to the 'Export - Preference' screen. The following image shows selection of additional sections:



The 'Export - Preference' screen is shown below:



- **Export preference:** select the preference to export. This field displays 'No preference' if this is the first export.
- **Section:** select the section to set to the selection fields in the 'Mapping field' table. Click the 'Set' button to set the selected section for the checked fields in the 'Mapping field' table.

- **Mapping field:** display the table of 'Mapping fields' for ADS export. This table contains only fields that are mapped for 'Mapping table' in the 'Member' type or 'Description'. The fields in types 'Name' and 'Description - Language' are also hidden in this table.
- **Save preference:**
 - No: do not save the preference.
 - Update current preference:
 - Displays only when the 'Export preference' is not set to 'No preference'.
 - Updates the content of the currently selected preference in the TIBCO EBX® Add-on for Oracle Hyperion EPM > Preference table.
 - New preference:
 - Creates a record to store the 'Mapping field' table configuration in the 'Export - Preference' screen in the TIBCO EBX® Add-on for Oracle Hyperion EPM Preference table.



Samples

The following samples are provided with the add-on.

FILE	DESCRIPTION
template-Essbase.xlsx template-Essbase.xls	Templates used when managing import and export procedures for Essbase
hfm-example.xml	Sample of data values for HFM in XML format
Planning-Entity-example.xml	Sample of data values for Planning in XML format

Administrative data

Logical data model



The data mapping configuration is used during the import and export procedures to drive the transformation from the source data format into target data format. The add-on uses an XLM pivot format based on the grammar of HFM XML (XML pivot).

Declaration of the MDM data structure

MDM table

☐

MDM field

☐

Table mapping

Applied to HFM and Essbase

A. Dimension, Table type = member

☐

B. Hierarchy, Table type = node

☐

Applied to Planning

A. Table type = member & node

☐

B. Table type = DimensionSetting

☐

Description

Table type = Description

☐

Mapping extension

☐

Field mapping

Regular field

☐

Field in a hierarchy



Field description



Field mapping for Planning



Dimension mapping



Language mapping



Special case - mapping "Default" language as "English":

- Select the "Mapping field" table
- Filter FieldType = 'Description' and change XML path value from /DESCRIPTION to / DESCRIPTION



- Select "Language Mapping":



Data mapping logging

Logging Overview

The data mapping logging is activated to track technical information on data mapping execution. When a trouble is raised during the data mapping execution, either at import or export time, then analyzing the data mapping logs is helpful to fix the issue.

This log mechanism is useful at the time when data mapping configuration must be tested. Once this configuration is validated, the log mechanism should be deactivated.

Configuration

In the administration area of EBX®, and in the Hyperion configuration, it is possible to configure the maximum number of data mapping instances to track. For example, if the 'Depth logging' property

is set to '10', then the first 10 dimensions, and on each dimension the first 10 members, hierarchy nodes and settings are logged. If the limit is set to '-1' the logging is executed for the whole data.

It is recommended to set the value to '10' since the data mapping configuration is fully tested with this level of depth.



Data structure to apply the logging

The data mapping logging uses a set of tables located in the administration area of EBX®, named 'TIBCO EBX® Add-on for Oracle Hyperion EPM - Logging'.

The logging information is deleted before every new execution. This is not an audit trail that the system keeps in memory in a sustainable way. This is a technical tracking of data mapping execution that is valuable to fix issues that might appear due to wrong data mapping configuration.

The logging report is available through a data hierarchy defined in the table 'Logging report'. The view is as follows. For each data mapping operation, a set of information is provided to track the value of the source data against its target data. A data hierarchy view is provided to facilitate the use of the logging data. For example, the parent node 'Account' is structured with as many child nodes as fields in the dimension.



Root log

This table contains general logging information.

FIELD NAME	DESCRIPTION
Application	The current application which is being imported/exported
Procedure	The current procedure which is running: - Convert - Import - Export
Stage source	The current stage, link to table Stage. For example: - Source = XML, Target = EBX (import HFM xml file) - Source = EXCEL, Target = XML (convert ESSBASE spreadsheet to HFM xml)

FIELD NAME	DESCRIPTION
Stage target	
Date	The current date time when running the service
User	The current user who runs the service
UUID	The uuid string to separate the current session getting from EBX®
Dataset	The current dataset which is being imported/ exported
Data space	The current data space which is being imported/exported

Stage

FIELD NAME	DESCRIPTION
Type	The information to use for the source and the target in the stage: - XML - EBX® - EXCEL - CSV

Instance

This table stores the hierarchy information for the Object (Logging report).

FIELD NAME	DESCRIPTION
Object	Parent object
Object child	Child object

Object type

FIELD NAME	DESCRIPTION
Type	The information to use for the source log and the target log in the Object: - Dimension - EBXTable - EBXField - DimensionXML -

FIELD NAME	DESCRIPTION
	XMLTag - SpreadSheet_Sheet - SpreadSheet_Cell - CSV_Column
Description	The description for each type

Object (Logging report)

This table stores the detail logging data for each object (dimension, member, hierarchy, setting, field). The view 'Logging report' is configured on this table.

FIELD NAME	DESCRIPTION
Root log	The current Root log, reference from table 'Root log'
Source log	The current log for current stage. For example: - Stage Source = XML, Stage Target = EBX, Source log = Dimension XML, Target log = Dimension (log for the import dimension in stage import HFM xml file to EBX®)
Target log	
Source path	The current source path
Target path	The current target path
Value	The value which is being processed
Free log	Free log

Import and export report

Report usage and location

The import and export report is used to log information during the import and export procedures. The information is stored in a dedicated data set allowing to request and purge the logs over time. It is located in the data space 'Oracle Hyperion EPM' and the data set 'Oracle Hyperion EPM Report'.

Data architecture used to log the import and export report



The information logged during the import and export procedures is saved through this logical data model.

Import result

The import result logs all import parameters and the execution status.

FIELD NAME	DESCRIPTION
User	The user who execute the import procedure.
File	The file that is imported
File path	Path of the imported file.
Import type	Type of the import (Eg. Delete all before import)
Update dimension and field order	'True': when the order of dimensions and fields is updated after the import. 'False': when the order of dimensions and fields is not updated after the import.

FIELD NAME	DESCRIPTION
Application	Application involved in the import procedure.
Dataset financial	Name of the financial dataset.
Dataset extension	Name of the extension dataset.
Dataset mapping	Name of the mapping dataset.
Selected dimension	List of the dimensions that are selected.
Execution time	Execution time of the import procedure.
CSV version	The CSV version of the exported file.

FIELD NAME	DESCRIPTION
Export type	Type of the export.
Export root in CSV flat file	'True': the root is exported for CSV file. 'False': the root is not exported for CSV file.
Delimiter	Delimiter character used for exporting CSV file.
Export empty field app setting	'True': empty fields app setting is exported. 'False': empty fields app setting is not exported.
Export full relationship	'True': full relationship is exported. 'False': full relationship is not exported.
Export #root	'True': the '#root' is exported for HFM CSV file. 'False': the '#root' is not exported for HFM CSV file.
Aggregation weight display format	Define the export format for the field 'AggregationWeight' in the dimension Custom. This format applies the 'Java - Decimal Format'. Value must conform to the regular expression "[-]?[0#,.]+[.]?[0#]*[E]?[0]*".
Default select all export	'True': 'select all dimension(s)' check box is selected by default. 'False': 'select all dimension(s)' check box is not selected by default.

FIELD NAME	DESCRIPTION
Export order for hierarchy	Selection of the order management applied to the hierarchy: • 'Order using Child order' - hierarchy on export is based on 'Child order'. • 'Same order as Data hierarchy view' - hierarchy on export is based on the Data hierarchy view.
Status	'True': the import procedure is a success. 'False': the import procedure has failed.
Use TIBCO EBX® Add-on for Oracle Hyperion EPM - Logging	'True': the TIBCO EBX® Add-on for Oracle Hyperion EPM - Logging has been used. 'False': the TIBCO EBX® Add-on for Oracle Hyperion EPM - Logging has not been used.
Run in background	'True': the import procedure has been executed in background 'False': the import procedure has not been executed in background.
Error message	If case the import procedure fails, then the error message is logged.
Unsupported tag	List of XML tags that are defined in the data mapping.
Not imported dimension	List of name of dimensions that are not imported during the procedure.

Import history

The import history logs all detailed information of the import per dimension of the financial data model as illustrated below.

FIELD NAME	DESCRIPTION
Import result	Link to the corresponding 'Import result' procedure.
Dimension	Dimension name that is imported.
Table	Table name that is imported.
Number of created record	Number of created records after the execution of the import procedure.
Number of modified record	Number of modified records after the execution of the import procedure.
Extension table name	Name of extension table that is imported.

Export result

The export result logs all export parameters and the execution status.

FIELD NAME	DESCRIPTION
User	The user who execute the export procedure.
Version	The version of current dimensions.
CSV version	The CSV version of the exported file.
Export type	Type of the export.
Export root in CSV flat file	'True': the root is exported for CSV file. 'False': the root is not exported for CSV file.
Delimiter	Delimiter character used for exporting CSV file.
Export empty field app setting	'True': empty fields app setting is exported. 'False': empty fields app setting is not exported.

FIELD NAME	DESCRIPTION
Export full relationship	'True': full relationship is exported. 'False': full relationship is not exported.
Export #root	'True': the '#root' is exported for HFM CSV file. 'False': the '#root' is not exported for HFM CSV file.
Aggregation weight display format	Define the export format for the field 'AggregationWeight' in the dimension Custom. This format applies the 'Java - Decimal Format'. Value must conform to the regular expression "[-]?[0#],[.]?[0#]*[E]?[0]*".
Default select all export	'True': 'select all dimension(s)' check box is selected by default. 'False': 'select all dimension(s)' check box is not selected by default.
Export order for hierarchy	Selection of the order management applied to the hierarchy: • 'Order using Child order' - hierarchy on export is based on 'Child order'. • 'Same order as Data hierarchy view' - hierarchy on export is based on the Data hierarchy view.

FIELD NAME	DESCRIPTION
Application	Application involved in the export procedure.
Dataset financial	Name of the financial dataset.

FIELD NAME	DESCRIPTION
Dataset extension	Name of the extension dataset.
Dataset mapping	Name of the mapping dataset.
Selected dimension	List of the dimensions that are selected.
Execution time	Execution time of the export procedure.
Status	'True': the export procedure is a success. 'False': the export procedure has failed.
Use TIBCO EBX® Add-on for Oracle Hyperion EPM - Logging	'True': the TIBCO EBX® Add-on for Oracle Hyperion EPM - Logging has been used. 'False': the TIBCO EBX® Add-on for Oracle Hyperion EPM - Logging has not been used.
Run in background	'True': the export procedure has been executed in background 'False': the export procedure has not been executed in background.
Error message	If case the export procedure fails, then the error message is logged.

Export history

The export history logs all detailed information of the export per dimension of the financial data model as illustrated below.

FIELD NAME	DESCRIPTION
Export result	Link to the corresponding 'Export result' procedure.
Dimension	Dimension name that is exported.
File	The file that is exported.

Reference data

Application

FIELD NAME	DESCRIPTION
Name	Name of the Hyperion application

Dimension

FIELD NAME	DESCRIPTION
Name	Name of the dimension

Table

FIELD NAME	DESCRIPTION
Name	Name of the table in the financial data model

File type

FIELD NAME	DESCRIPTION
Name	File extension (Eg. '.xml')
Label	Label of the file extension

File

FIELD NAME	DESCRIPTION
Name	Name of the file that is imported or exported
Type	The type of the file among the list provided by the table 'File type'

Purge of the report data

At the level of the data set 'Oracle Hyperion EPM report', the service 'Purge report' allows to remove all or part of the report data.



Migration & Back-up procedures

Migration requirements

From the version 1.0.2 of the add-on, a migration procedure is provided to facilitate the upgrades. If modifications have been done in the data mapping or in the financial data model itself, then specific steps must be achieved as explained in this appendix.

The migration procedure relies on two services:

- **Backup service.** This service is executed on the version (n) to backup all configurations (structure, permission, reference data) and the Financial dataset. It must be executed on the old version (version n) before upgrading the add-on to the new version. This service executes the following steps:
 - Backup the structure and all permissions of the Financial dataset.
 - Backup data of all reference tables (in Reference data group).
 - Use the mapping dataset of version (n) to backup financial data.
- **Restore service.** After the upgrade of the add-on to the version (n+1), the restore service is launched on this new version to restore the backup. This service executes the following steps:
 - Restore the structure and permission of the Financial dataset.
 - Restore data of all reference tables.
 - Use the mapping dataset of version (n+1) to restore financial data.

In order to avoid any error when using the backup and restore services, you can delete unused records on the table 'Application' (in group 'Administrative Data' on Oracle Hyperion EPM 1.0.x or in dataset mapping on Oracle Hyperion EPM 1.1.x). For example, if you don't use the application Essbase, you can delete the record 'ESSBASE' in the table 'Application' in every Oracle Hyperion EPM dataset. Then, the backup service will work and export without data related to Essbase.

Beginning in version 1.6.0 of the add-on, the migration procedure has been upgraded. The backup and restore services allows you to backup multiple data spaces. The restore service is also available to restore the backup folder from versions prior to 1.6.0.

The migration procedure does not backup and restore data in the Oracle Hyperion EPM Extension dataset. This data migration must be achieved manually with the upgrade of the add-on.

Backup

For versions prior to the 1.6.0 release

- Step 1: Select a Oracle Hyperion EPM data space.
- Step 2: From the 'Services' drop-down menu select 'Oracle Hyperion EPM' → 'Backup'.
- 
- Step 3: Enter a name in the 'Backup folder name' field.
- 
- Step 4: Click the 'Export' button to backup data.
- Step 5: EBX® returns the path of the backup folder on server. For example, this folder is located at {ebx.temp.directory}/ebx.platform.modules/ebx-addon-hmfh/backup). The path returned by EBX® is ebx-addon-hmfh\backup\Hmfh1.0.1_325026.
- 
- Step 6: Save the path.

From version 1.6.0

- Step 1: On Dataspace level, choose one Oracle Hyperion EPM dataspace.
- Step 2: Choose the service Backup.
- 
- Step 3: Enter a folder name for exporting.
 - Backup folder name: enter the name for backup folder.
 - Data spaces: select the data spaces to backup from data spaces' tree. If the children are selected, the parent is also selected.
 - Application: select application user want to backup.
- 
- Step 4: Click the export to backup data.



- Step 5: EBX® will return the path of the backup folder on server (for example, this folder is located at {ebx.temp.directory}/ebx.platform.modules/ebx-addon-hmfh/backup). For example: ebx-addon-hmfh\backup\Hmfh1.0.1_943079.
- Step 6: Save the path.

Restore

Before version 1.6.0

- Step 1: From the 'Services' drop-down menu select 'Oracle Hyperion EPM' → 'Restore'.



- Step 2: Enter the path of the backup folder.



- Step 3: Click the 'Import' button to restore data.
- Step 4: EBX® will restore the data in the backup folder.



From version 1.6.0

- Step 1: From the 'Actions' drop-down menu select 'Oracle Hyperion EPM' → 'Restore'.



- Step 2: Select folder name



- Step 3: Select data space
- Step 4: If you select data space that exists in the system, a duplicated data space screen displays:



There are two available actions for a duplicated data space:

- **Ignore** : Keep current data space in system and do nothing.
- **Replace with data space in backup folder**: Delete the duplicated data space in the system and replace it with the data space in the backup folder.
- Step 5: If you choose ' **Replace with data space in backup folder**' the system shows the warning screen to inform you that the duplicated data spaces will be erased from the EBX® system before restoring.



- Step 6: Click the 'Submit' button to start the restore process.

How to upgrade version n to version n+1

- **Run the 'Backup' service from version n**
 - Perform the steps outlined above in the "Backup" section to backup your data.
- **Upgrade to the add-on version (n + 1)**
 - Step 1: Shut down the server. *Perform a backup of the repository.*
 - Step 2: Update the Oracle Hyperion EPM version (n) repository with the new version of add-on (n+1).
 - Step 3: Start EBX®.
- **Restore**
 - Run the 'Migrate data set mapping' service (if needed).
 - Perform the steps outlined in the above "Restore" section to restore your data.

How to upgrade from one server to another

- **Backup**
 - Perform the steps outlined above in the "Backup" section to backup your data.
- **Install the new add-on version (n+1)**
 - Step 1: Install the new version of the add-on (n+1) on a new server.
 - Step 2: Start EBX®.
- **Restore**
 - Step 1: On a *new repository* with the add-on version (n+1).
 - Step 2: Copy the backup folder on the server from Oracle Hyperion EPM version (n) to the compatible backup folder on the server add-on version (n+1) (using the path which was saved on the Backup).
 - Step 3:
 - Run the 'Migrate data set mapping' service (if needed).
 - Perform the steps outlined above in the "Restore" section to restore your data.

Special case of upgrading extra custom to version 1.6.0

- Since there are newly added tables in 1.6.0 of the add-on, you need to add new tables to a custom model before running the migration process.

- The following image highlights these new tables:



- Create the following tables: 'Currency hierarchy', 'Currency setting', 'Alias hierarchy', 'Alias setting', 'Version', 'Version hierarchy', 'Version setting', 'Version description' and configure their primary keys:



NAME	LABEL	INCLUDED DATA MODELS	PRIMARY KEY
CurrencyHierarchy	Currency Hierarchy	Currency hierarchy	/FKCurrency / FKChildCurrency / active
CurrencySetting	Currency Setting	Currency setting	/Attributes/Name
AliasHierarchy	Alias Hierarchy	Alias hierarchy	/FKAlias / FKChildAlias /active
AliasSetting	Alias Setting	Alias setting	/Attributes/Name
Version	Version	Version	/Name
VersionHierarchy	Version Hierarchy	Version hierarchy	/FKVersion / FKChildVersion / active
VersionSetting	Version Setting	Version setting	/Attributes/Name
VersionDescription	Version Description	Version description	/FKVersion /FKAlias

- Publish this model and run the migration process.

Special cases in restoring

In certain situations the backup folder may contain data spaces having the same home key as the data spaces set to backup. The table below describes how to treat these cases.

NO.	IN SYSTEM	DATA SPACE ON WHICH THE RESTORE IS LAUNCHED	IN BACKUP FOLDER	RESULT
1	ebx-addon-hmfh -> A ----> B -----> C	ebx-addon-hmfh	ebx-addon-hmfh -> A -> B ----> C ----> D ----> E	Success. Depending on the user's selection for the data space tree in backup folder, the result could be : ebx-addon-hmfh -> A ----> B -----> C -----> D -----> E
2	ebx-addon-hmfh -> A ----> B -----> C	A	ebx-addon-hmfh -> A -> B ----> D ----> E ----> C	Failed. System shows error message.
3	Master reference -> ebx-addon-hmfh -> B -> C	ebx-addon-hmfh	ebx-addon-hmfh -> C ----> B -----> A	Failed. System shows error message.

Changes management

Customization options

The add-on can be customized as follows:

- 'Mapping dataset' to adapt the import and export data flux depending on applications used by a company. This type of change is achieved by a parametrization of the tables used for the data mapping. These tables are reference data managed by EBX®.
- 'Hyperion financial data structure' to add or remove data with help of the modeling function of EBX® (DMA: Data Modeler Assistant).

- 'Inclusion of the Hyperion data model' into a user data model to overload the default data brought by the add-on. This inclusion relies on the EBX® feature 'include' available with the DMA.
- 'Permission' to adapt the policies. This type of change is achieved by the permission management function brought by EBX®.
- 'Data views' to adapt the labels used by default. This type of change is achieved by using the EBX® data views feature.

These customizations are not mandatory. They depend on the context and needs of every company. Once changes are applied, it is necessary to keep them safe when a new version of the add-on is installed. This appendix explains, for each type of change listed below, how to upgrade the add-on without removing the changes.

Change on the mapping dataset

CASE	CHANGE APPLIES ON		DESCRIPTION
Value	Structure		
#1	X		Value change in the mapping dataset
#2		X	Data structure change in the mapping dataset

A user of the add-on can only change the value of the data mapping.

The changes applied on the data structure is strictly under the responsibility of Cloud Software Group, Inc.. A user should not change the data structure of the data mapping. This type of change is not kept when a new version of the add-on is installed.

Case #1 - Value change in the mapping dataset

#1.1 - Changes done by the user of the add-on

The following steps must be executed to keep the changes in the mapping data set and associated data:

- **Step 1** : On version (n), export archive for only Oracle Hyperion EPM Mapping data set to '.ebx' file (Data Spaces > Actions > Archives > Export).
- **Step 2** : Run Backup service (Data Spaces > Services > Backup) (see Appendix - Migration and Back-up procedures).
- **Step 3** : Upgrade Oracle Hyperion EPM to version (n+1).
- **Step 4** : Import archive file for mapping dataset by using exported '.ebx' file in step 1. It means that the mapping archive of version (n+1) is not used on this step.
- **Step 5** : Create a child dataset for mapping dataset. After this step, both parent mapping dataset and child mapping dataset contains the mapping data of version (n).



- **Step 6** : Import archive file for parent mapping data set by using archive file in Oracle Hyperion EPM version (n+1). After this step :
 - Parent mapping dataset contains new data mapping of Oracle Hyperion EPM version (n+1).
 - Child mapping data set contains old data mapping of Oracle Hyperion EPM version (n).



- **Step 7** : Deactivate dataset parent.



- **Step 8** : Optimize dataset then go to child mapping data set and get the differences from the old data mapping. The child dataset mapping contains both the modification of the user of the add-on and the new modification of Oracle Hyperion EPM version (n+1). The user can work on child dataset mapping to control all of these different.



- **Step 9** : Use the child dataset mapping to restore data by using the Restore service of the add-on. (Data Spaces > Services > Restore) (see Appendix - Migration and Back-up procedures).

#1.2 - Changes done by Cloud Software Group, Inc.

In this case, only Cloud Software Group, Inc. has changed the data mapping. In other words, it means that the user of the add-on still works on the default data mapping of the add-on version (n).

Then, the migration procedure is used without additional procedure (see Appendix - Migration and Back-up procedures).

Case #2 - Data structure change in the mapping dataset

The data structure of the mapping dataset can be modified by Cloud Software Group, Inc. only. Cloud Software Group, Inc. tries to ensure a forward compatibility of the user mapping dataset changes against the new data structure. In case of need, a migration procedure can be provided to realign data values with the new data structure.

Change on the Hyperion financial data structure

The Hyperion financial data structure can be modified by Cloud Software Group, Inc. only. The user of the add-on cannot modify this structure because every change will be lost during the installation of a new version of the add-on.

But the user of the add-on can extend the Hyperion financial data structure to link the Hyperion data with other tables.

CASE	CHANGE ON THE HYPERION FINANCIAL DATA STRUCTURE BY		DESCRIPTION
User	Cloud Software Group, Inc.		
#1		X	Modification of the Hyperion financial data structure.
#2	X		Extension of the financial data structure.

Case #1 - Modification of the Hyperion financial data structure

This type of modification can be done by Cloud Software Group, Inc. only.

CASE	CHANGE ON		DESCRIPTION
Table	Field		
#1.1	X		New table.
#1.2	X		Deletion of a table.

CASE	CHANGE ON		DESCRIPTION
#1.3		X	New field.
#1.4		X	Field name change.
#1.5		X	Field group owner change.
#1.6		X	Change a field into a foreign key.
#1.7		X	Deletion of a field.

#1.1 - New table

Add the path for new table to Path table:

☐

Register this table to the add-on by creating a new declaration in the 'MDM Table':

☐

Register all fields of this table to the add-on by creating their reference data in the 'MDM Field' table:

☐

Enter the mapping declaration of the table with one application: 'Mapping Table':

☐

Mapping all fields (MDM Field) of this table (Mapping Table) with the fields listed below by creating new data mapping in the 'Mapping Field' table:

- XML path: XML path in HFM XML file.
- Essbase spreadsheet column name: Name of column in Essbase spreadsheet.
- XML path in Planning: XML path in Planning XML file.
- Planning CSV column name: Column name in Planning CSV file.
- Flat CSV column name: Column name in HFM flat file.

☐

#1.2 - Deletion of a table

Delete the data related to this table on the table 'MDM Table':

☐

Delete the data related to this table on the table 'MDM Field':

☐

Delete the data related to this table on the table 'Path':

☐

Delete its reference data on tables mapping

- 'Mapping Table'

☐

- 'Mapping Field'

☐

#1.3 - New field

Create a new declaration on the table 'MDM Field':

☐

Create a new mapping for the new 'MDM field':

☐

Create a new mapping for the new 'MDM field':

#1.4 - Field name change

Example: change field "Name" to "Name1".

Create a new path for the field Name1.

☐☐

Align the corresponding data on the table 'MDM Field' (Path, Name):

☐

#1.5 - Field group owner change

Example: move the field "Name" to the group "Reference"

Create a new path for the new field Name:

☐

Align the corresponding data on the table 'MDM Field' (Path, Name).

☐

#1.6 - Change a field into a foreign key

Align data on the table 'MDM field':



#1.7 - Deletion of a field

Example: deletion of the field "Name" in the table Account.

Delete the corresponding data on the table 'MDM Field' table:



Delete the corresponding data mapping for this MDM field in the 'Mapping Field' table:



Case #2 - Extension of the Hyperion financial data structure

Two types of extension are permitted: use of external foreign keys, use of an extension mechanism provided by the add-on.

CASE	CHANGE ON			DESCRIPTION
Structure	Permission	View		
#2.1	X			Use of an external foreign key (FK) to a table in the Hyperion financial data structure
#2.2	X			Extension from the Hyperion financial data structure to another data structure

#2.1 - Extension by using an external foreign key (FK)

The use of external foreign keys (FK) from a user's data model to the Hyperion financial data model is permitted. It does not entail specific procedure when upgrading the add-on.

Thus, the migration procedure can be used to keep financial data safe when a new version of the add-on is installed (see Appendix - Migration and Back-up procedures).

#2.2 - Extension by using the extended data

#2.2.1 - Description



The add-on provides a dataset for extension data which contains one extension table for each dimension. For example, the dimension 'Account' has its own extension table: 'AccountExtension'.



This data model is then modified to add extension fields. The end-user access the extended information from the financial dataset by using the link Extension as highlighted below (example with a record 'T1' Account).



Limitation of the extension mechanism:

- The parent and children dataspace and datasets of dataset financial use the same dataset extension.

#2.2.2 - Upgrade procedure

Once the extension mechanism is used, then the following procedure must be executed to install a new version of the add-on:

- **Step 1** : Export Oracle Hyperion EPM extension data model to xsd file.



- **Step 2** : Export Oracle Hyperion EPM extension dataset (contains data) to archive file.



- **Step 3** : Use Migration procedure to save financial data on Oracle Hyperion EPM dataset (see Appendix - Migration and Back-up procedures).
- **Step 4** : Because after upgrading Hyperion to version (n+1), the Oracle Hyperion EPM extension data model will be overwritten by the default one, the user needs to import Hyperion extension data model to xsd file which was exported on step 1 to restore the old model.
- **Step 5** : Import the archive file, which was exported in step 2, into the Oracle Hyperion EPM extension dataset to restore the data in this dataset.

Inclusion of the Hyperion data model

From the version 5.3.0 of EBX® it is possible to include data models into another one. You can either include the Oracle Hyperion EPM data model into a user data model or include within the Oracle Hyperion EPM data model a user data model.

CASE	CHANGE ON			DESCRIPTION
Structure	Permission	View		
#1	X			Include the Hyperion data model into a user data model
#2	X			Include a user data model into the Hyperion data model

Case #1 - Include the Hyperion data model into a user data model

The data model provided by the add-on relies on complex data types that can be reused to build bespoke tables by using the inclusion of types. Here is the procedure:

- **Step 1** : In the DMA (Data Model Assistant), go to Configuration > Included data models, create new record as shown below:



- **Step 2** : After including the Hyperion data model, all complex data types of Hyperion are included into the user's data model. Open the "Included complex data types" to check:



- **Step 3** : Go to Configuration > Data model properties, set value for Module name as shown below:



- **Step 4** : Create a new group in Data structure (for example, group Company Financial Data).
- **Step 5** : In order to use the service Backup/Restore of the migration procedure on this dataset, user must set information for root node to "Hyperion".

- **Step 6** : Create new tables for all dimensions (in the created group in step 4) by using the complex data type of Hyperion. Each dimension requires 4 tables.
 - For example: dimension Account require 4 tables:
 - Account: stores all members of the Account dimension.
 - AccountHierarchy: contains all relation between one member to others. The add-on uses this table to create hierarchy view for this dimension.
 - AccountDescription: contains all descriptions for each member.
 - AccountSetting: contains the dimension setting used for Planning application.

These tables must be created for all dimensions: Account, Custom1, Custom2, Custom3, Custom4, Entity, Scenario, Period, Year.

These shared tables must be created: ICP, Security Class, Currency, Currency Description, Application Setting, Consolidation Method, Consolidation Method Description, Alias.

The tables must reuse the same names and data organization (levels and sub-levels) than ones used by the add-on. All tables must be created otherwise errors could be raised either at the time of EBX® validation and/or when importing and exporting data with the add-on (data mapping concerns).



- **Step 7** : Create Reference Data group to store reference tables by using ReferenceDataType. This group must be located in the same model level, same group with the tables declared in the previous step.
- **Step 8** : Go to table 'MDM Table' in the mapping dataset, change the path of each MDM table by the real path in company's financial dataset:



Impact when upgrading the version of the add-on

Because the backup and restore services brought by the migration procedure are designed to work with the default data model, the migration procedure to backup/restore data is no longer usable. When inclusion of the data model is used as explained above, data is backup and restore manually with help of this procedure:

- **Step 1** : Backup data in dataset manually.
- **Step 2** : Upgrade Oracle Hyperion EPM from version (n) to version (n+1). After the upgrade, the complex data types are automatically updated.

- **Step 3** : Restore data in dataset manually.
- **Step 4** : Follow the user guide in section 'Change on dataset mapping' to manage the changes on dataset mapping.

Case #2 - Include a user data model into the Hyperion data model

It is possible to include a user data model into the Hyperion data model directly. This inclusion is achieved by following the procedure explained below:

- **Step 1** : Create a data model to include to the Hyperion data model.



- **Step 2** : When a data model is included, EBX® will include only the complex data types, triggers and value functions. EBX® will not include tables in the data model. Then it is required to create the complex data types for the tables, the groups or the fields that will be integrated to the Hyperion data model. For example, in the following data model, a new complex data type is created and contains a new group (group1) and one field (field1):



- **Step 3** : Open the Hyperion data model, go to Configuration > Included data models, create new record as the following figure:



- **Step 4** : After including the user data model into the Hyperion data model, all complex data types are included into the Hyperion data model. To go the "Included complex data types" to check.
- **Step 5** : Create new tables, add new groups or add new fields into the Hyperion data model by using the included complex data types. For example, a new group in Account table is created using the complex data type in the step 2.



- **Step 6** : To allow the add-on to take into consideration the new tables and fields, it is required to update the data mapping with these changes (see "Change on mapping dataset"). For example, to take into account the new group and field added on step 5, a new record in table 'MDM Field' is created:



Then the data mapping for the new field is created:



Impact when upgrading the version of the add-on

Even though the inclusion of a data model into the Hyperion data model, it still remains possible to use the migration procedure:

- **Step 1** : Use the Backup service to backup data in financial dataset (see Appendix - Migration and Back-up procedures).
- **Step 2** : Upgrade the Oracle Hyperion EPM from the version (n) to the version (n+1). After upgrading, the complex data types are automatically updated.
- **Step 3** : Follow all steps explained in the section "How to include user data model into Hyperion data model" to re-include user's data model into Hyperion data model.
- **Step 4** : Use the Restore service to restore data in financial dataset (see Appendix - Migration and Back-up procedures).
- **Step 5** : Follow the user guide in section "Change on mapping dataset" to manage the changes on the mapping.

Changes applied on the permission

CASE	CHANGE ON			DESCRIPTION
Structure	Permission	View		
#1		X		Permission change applied on a dataset
#2		X		Permission change applied on the data model

Case #1 - Permission change applied on a dataset

The migration procedure can be used to backup and restore the permission on a dataset.

Case #2 - Permission change applied on the data model

The changes applied on the permission at the level of the data model are not kept when the add-on is upgraded. These changes must be reapplied manually after the upgrade process.

To change permission on the data model, these steps must be applied:

- **Step 1** : In the DMA (Data Modeler Assistant), go to Action > Permission to open the table permission for the data model.



- **Step 2** : Add a new record to set a new permission.



Changes applied on the data hierarchy views

The data hierarchy views' labels provided by the add-on should not be changed directly. When such changes are needed, it is recommended to duplicate the data hierarchy views and then applied the changes on these news views.

When upgrading from the version (n) to the version (n+1), it is possible to export bespoke views from version (n), then importing them into the version (n+1).

Duplication of a view

- **Step 1** : Select the duplicate service on the view.



- **Step 2** : Set a new name for the duplicated view.



- **Step 3** : Set a new label for the duplicated view.



Export and import the views

- **Step 1** : Go to Administration > Views configuration.
- **Step 2** : Find the view with help of its documentation, then export/import it.



Create the views manually

When the Hyperion data model is included into a user data model, the predefined views brought by the add-on must be created manually. The procedure is described below.

#1. Data hierarchy view

A hierarchy view is used for each dimension in order to manage the members through a parent and child relationship. For example, create the hierarchy view for the Account dimension:

- **Step 1** : Go to the Account table, choose View > Create a new custom view.



- **Step 2** : Enter the view name, owner and set view mode to Hierarchy view.



- **Step 3** : On the 'Dimension' field, select Active to classify member by active value.



- **Step 4** : Enter a label for the view and select Child order as Ordering node.



- **Step 5** : Click on Create and apply button to create the view.

#2. The view used to display the active relationships

The active view is used on each table hierarchy for displaying only records which has 'active' value Yes or No. For example, create the view 'Active yes' for the Account hierarchy:

- **Step 1** : Go to Account Hierarchy table, choose View > Create a new custom view.

- **Step 2** : Enter the view name, owner and set view mode to Simple tabular view.



- **Step 3** : On Columns field, select column to display. And then, on Filter field, add new criterion.



- **Step 4** : Select Field 'Active', Operator '=', Value 'Yes'.



- **Step 5** : Click on Create and apply button to create the view.



Adding custom dimension

From the EBX® Add-on for Oracle Hyperion EPM 1.5.0 version, it is possible to add any bespoke custom dimension. The process to configure a new custom dimension is described below.

To create an extension model, create a new data model and import the 'includedHyperion.xsd' file. The model populates with Oracle Hyperion EPM dimension tables. After importing, you can publish the model to add any bespoke custom dimension.

- **Step 1**: create a new data model and include the 'ebx-addon-hmfh.xsd data model'.



- **Step 2:** add a new custom dimension, for instance 'Custom 5' of data type 'Custom'



- **Step 3:** data mapping configuration

In the 'MDM Table' table:



In the 'MDM Field' table:



In the 'Dimension' table:



In the 'Mapping table' table:



In the 'Mapping field' table:



In the 'Application dimension' table:



- **Step 4:** create Data Hierarchy view



Extension

This feature allows you to add additional fields to Oracle Hyperion EPM dimension tables. Even though the add-on integrates the data model, you can create an extension model, add a data set based on this model and perform a couple of mapping tasks to enable import/export of extension data. The following sections describe this process

Create an extension model

To create an extension model, create a new data model and import the 'sampleExtensionTemplate.xsd' file. The model populates with extension tables for each Oracle Hyperion EPM Dimensions data set table. After importing, you can publish the model and create a data set.



The 'com.orchestranetworks.addon.hmfh.DefaultExtensionTableTrigger' is provided for every extension table. This trigger automatically fills the primary key for an extension record when displaying in the 'Extension' tab.



As shown below on the working dimensions data set, set the 'Data set extension' property to the extension data set you just created.



Each record now contains a new tab for the extension data:



In order to work with the extension data on this tab, you will need to add the necessary mapping in the mapping data set. The following section describes how to define the 'Default data set mapping'.

Mapping for extension model

Mapping for extension tables

Firstly, you'll need to create extension data on the 'MDM table' table. As a best practice, you may want to apply the same naming convention used by the corresponding table in the data model. As shown below, you can use the path from the model for the MDM table.



From the record detail page, you can create a 'Mapping table' for the extension table:



The value of 'Table type' and 'XML path' should be the same as the table for which you want to create the extension. For example, the 'AccountExtension' table extends the 'Account' table. So, 'Table type' and 'XML path' should be set to the same values used in the 'Account' table.

After you create the mapping data, you can use the 'Extension' tab to access the extension data.



Mapping for extension fields

In order to import/export extension data, you must define mapping data for extension fields.

You can create mapping data for extension fields from an 'MDM table' record in the mapping data set:



The field name and data type should be similar with those used in the data model.

The path for the MDM field can come from the extension model.



On the newly created MDM field record, create a new mapping field record, as shown below:



The XML path for the extension fields can be the same as normal attribute.



After you save and close, you can export/import the extension data using the Oracle Hyperion EPM services.

Integration with TIBCO EBX® Match and Cleanse Add-on

Configuration

In the configuration tab, the property 'Use TIBCO EBX® Match and Cleanse Add-on' must be set to 'True'. If the EBX® Match and Cleanse Add-on is not installed on your environment then it is no longer possible to configure this property.



Don't forget to import the archives for EBX Match and Cleanse Add-on before working on EPM-Matching data set. The two archives are also provided as a default configuration for EPM-Matching data set:

ebx-addon-hmfh-matching-configuration.ebx

ebx-addon-hmfh-include-daqa.ebx

Data transfer to the datasets using the EBX Match and Cleanse Add-on and opposite direction

When the 'Use TIBCO EBX® Match and Cleanse Add-on' configuration is activated, then the data space 'Oracle Hyperion EPM with Match and Cleanse Add-on metadata' is created automatically. It

is used to receive the financial data from the originate data space in order to apply the cleansing and matching operations.

The data transfer from the financial data space to one use for the EBX Match and Cleanse Add-on is performed through the service 'Copy to use TIBCO EBX™ Match and Cleanse Add-on' at the level of the data set 'Oracle Hyperion EPM Dimensions'. After the execution of this process, it is possible to transfer the data into the initial data space with the service 'Copy to remove TIBCO EBX™ Match and Cleanse Add-on' at the level of the data set 'Oracle Hyperion EPM Dimensions' of the data space used for the EBX Match and Cleanse Add-on.

Service 'Copy to use TIBCO EBX® Match and Cleanse Add-on'



When the property 'Create target data space' is set to 'True', then it becomes possible to enter the name of a new target data space that will be created under the selected data space.



After the selection of the target data space, it is needed to decide how the data transfer is performed when it already exists data for the underlying data sets.



Integration with TIBCO EBX® Rules Portfolio Add-on

Migration from Hyperion rules to EBX® Rules Portfolio Add-on

Before the version 1.5.0, the data set 'Hyperion rules portfolio' was used for storing the rules configuration. From the version 1.5.0, this portfolio has been transferred into the EBX Rule Portfolio Add-on. The rules configuration is now stored in the data set 'Rules portfolio', and the former data set 'Hyperion rules portfolio' is no longer used.

When upgrading EBX® Add-on for Oracle Hyperion EPM to the version 1.5.0, it is needed to execute the migration procedure as described below. It transfers the rules configuration from the 'Hyperion rules portfolio' data set to the 'Rules portfolio' data set. The former 'Hyperion rules portfolio' data set is then deleted.

Configuration

Be sure the EBX Rules Portfolio Add-on is installed and publish any rules using the 'Publish rules portfolio' service. This service is located under *Administration > TIBCO EBX® Rules Portfolio Add-on > Actions*.



If you don't want to use all the created rules, rule publications can be deleted from *Administration > TIBCO EBX® Rules Portfolio Add-on - Production > Rule publication > Actions*.



Creation of pre-defined rules

The three services shown in the following image allow you to create sample rules and generate predefined financial data rules. The use of these rules are not mandatory as you can use the EBX Rules Portfolio Add-on to create custom business and permission rules. See that add-on's documentation for more information.



Sample rules portfolio

NO	D.E.C.	RULE TYPE	NAME	DESCRIPTION
1	Account	Automated rule	For accounts, if Data Type is set to any value except Currency, you must set Exchange Rate Type to None	When create/modify account, if DataType is set to any value except Currency then the ExchangeRateType will be set to 'None'
2	Account	Automated rule	When create new account, automatically create new Custom4 member with the same name under "Source"	When create new account, automatically create new Custom4 member with the same name under "Source"
3	Account	Automated rule	If AccountType is REVENUE and "Submission group" not equal to 1 then	If AccountType is REVENUE and "Submission group" not equal to 1 then

NO	D.E.C.	RULE TYPE	NAME	DESCRIPTION
			automatically set "Submission group" to 1	automatically set "Submission group" to 1
4	Account	Manual validation rule	Only base-level accounts can be calculated	Only base account (account which doesn't have any child) can have IsCalculate = 'Y'.
5	Account	Manual validation rule	Account description must be defined	Account must have at least 1 description which not empty
6	Account	Table set rule	All account must be descendant of #root	If an account isn't descendant of #root then automatically put it under #root
7	Account	Validation rule	If account is intercompany account then the plug account must be defined.	If IsICP = 'Y' then the field PlugAccount must be defined.
8	Account	Table set rule	If AccountType is REVENUE then "Submission group" must be 1	If AccountType is REVENUE then "Submission group" must be 1

NO	D.E.C.	RULE TYPE	NAME	DESCRIPTION
9	Custom1	Table set rule	Custom1 having name starts with 'TX' must be child of 'TAXES'	Custom1 having name starts with 'TX' must be child of 'TAXES'
10	Custom2	Manual validation rule	Descendance of Non_Prod_types must have description starts with "Non_Prod"	If a custom2 member is descendant of "Non_Prod_types" then it's description must start with "Non_Prod"
11	Custom3 Hierarchy	Validation rule	Child of PORTLAND must having name ends with "Portland"	Descendance of PORTLAND must having name ends with "Portland"
12	Entity	Validation rule	Entity default currency must be defined	Field Currency in Entity must be defined.

Financial rules portfolio

RULE TYPE	RULE NAME
Automated rule	Rule 01: Custom3 needs to be created in source
Automated rule	Rule 02: Entity must have parent 'APMMCONTRIB'
Automated rule	Rule 02: Entity must have parent 'APMMCONTRIBDKK'
Automated rule	Rule 03: An entity should be in APMTOWN

RULE TYPE	RULE NAME
Manual validation rule	Rule 04: Account end with C should be set IsCalculated as 'Yes'
Validation rule	Rule 05: Account Description must be unique
Validation rule	Rule 06: TopMember level of parent account should lower than or equal to TopMember level of child account
Validation rule	Rule 06: TopMember level of parent account should lower than or equal to TopMember level of child account - Update
Manual validation rule	Rule 08: New validation account should be placed both in ValidsubX and in ValidationX
Manual validation rule	Rule 09: Custom1 member must has child which matches regular expression
Automated rule	Rule 10: The activity should be also placed in Custom2 under ICBASS
Automated rule	Rule 11: If entity has userdefined1 marked with X-CloA, X-CloE or X-CloI, change description to "CLOSED:" + entity description
Manual validation rule	Rule 12: The description of entity, which has userdefined1 with format X-CloA, X-CloE or X-CloI, must start with "CLOSED:"
Validation rule	Rule 13: Value of field "Plug account" must end with "P"
Validation rule	Rule 14: When account end with '9', value of field IsICP should be "Y" or "R"
Validation rule	Rule 15: When account end with '9', field ICP Top Member should be defined
Validation rule	Rule 16: All accounts descendants of 1_PL should start with 1

RULE TYPE	RULE NAME
Validation rule	Rule 17: All accounts descendants of 2_ASSETS should start with 2
Validation rule	Rule 18: All accounts descendants of 3_LIABILITIES should start with 3
Manual validation rule	Rule 19: Base accounts should also exist under 1_PL_ML
Manual validation rule	Rule 19: Base accounts in Helios data set should exist under 1_PL
Manual validation rule	Rule 20: Account type of descendants of 1_PL should be REVENUE
Validation rule	Rule 20: Account type of descendants of 1_PL should be REVENUE
Validation rule	Rule 20: The value of field "Account type" of 1_PL and its descendants member should be REVENUE
Manual validation rule	Rule 21: Account type of descendants of 2_ASSETS should be ASSET
Validation rule	Rule 21: Account type of descendants of 2_ASSETS should be ASSET
Validation rule	Rule 21: The value of field "Account type" of 2_ASSETS and its descendants member should be ASSET
Manual validation rule	Rule 22: Account type of descendants of 3_LIABILITIES should be LIABILITY
Validation rule	Rule 22: Account type of descendants of 3_LIABILITIES should be LIABILITY
Validation rule	Rule 22: The value of field "Account type" of 3_LIABILITIES and its descendants member should be LIABILITY

RULE TYPE	RULE NAME
Validation rule	Rule 24: Field "Enable Custom X Aggregation" of all accounts descendants of 1_PL or 2_BS should be YES
Validation rule	Rule 24: Field "Enable Custom X Aggregation" of all accounts descendants of 1_PL or 2_BS should be YES
Validation rule	Rule 25: Custom1 Top member must be defined
Validation rule	Rule 25: Custom1 Top member must be defined
Manual validation rule	Rule 26: Account ends with T but not start with 'TE' must have children.
Validation rule	Rule 27: Custom3 Top member must be defined
Validation rule	Rule 27: Custom3 Top member must be defined
Automated rule	Rule 28: Field Enable Data Audit of account must be set as FALSE
Manual validation rule	Rule 29: Account description must be defined
Automated rule	Rule 30: Field Security Class of entity should be set as "E-" + entity name
Manual validation rule	Rule 31: Base members should be present as descendant of BASS, TotAct, HSF
Automated rule	Rule 32: Field Security Class of all custom1 which is descendant of parent with 3 character, should be set as "C1-" + name of 3 character parent name
Automated rule	Rule 33: Automatically set "Exchange rate type" to None

RULE TYPE	RULE NAME
Validation rule	Rule 34: If account is intercompany account then the plug account must be defined.
Validation rule	Rule 35: Entity default currency must be defined

Rules for financial schema

NO	D.E.C.	RULE TYPE	NAME	DESCRIPTION
1	Account	Manual validation rule	IsCalculated field must be defined in Account table	Requires the Account table's IsCalculated field to have a value.
2	Custom1	Manual validation rule	IsCalculated field must be defined in Custom1 table	Requires the Custom1 table's IsCalculated field to have a value.
3	Custom2	Manual validation rule	IsCalculated field must be defined in Custom2 table	Requires the Custom2 table's IsCalculated field to have a value.
4	Custom3	Manual validation rule	IsCalculated field must be defined in Custom3 table	Requires the Custom3 table's IsCalculated field to have a value.
5	Custom4	Manual validation rule	IsCalculated must be defined in Custom4 table	Requires the Custom4 table's IsCalculated field to have a value.

NO	D.E.C.	RULE TYPE	NAME	DESCRIPTION
6	Custom1	Manual validation rule	SwitchSignForFlow	Requires the field must be defined in Custom1 table
7	Custom2	Manual validation rule	SwitchSignForFlow	Requires the Custom1 table's SwitchSignForFlow field to have a value.
8	Custom3	Manual validation rule	SwitchSignForFlow	Requires the Custom2 table's SwitchSignForFlow field to have a value.
9	Custom4	Manual validation rule	SwitchSignForFlow	Requires the Custom3 table's SwitchSignForFlow field to have a value.
10	Custom1	Manual validation rule	SwitchTypeForFlow	Requires the Custom4 table's SwitchSignForFlow field to have a value.
11	Custom2	Manual validation rule	SwitchTypeForFlow	Requires the Custom1 table's SwitchTypeForFlow field to have a value.
12	Custom3	Manual validation rule	SwitchTypeForFlow	Requires the Custom2 table's SwitchTypeForFlow field to have a value.
			SwitchTypeForFlow	Requires the Custom3 table's SwitchTypeForFlow field to have a value.

NO	D.E.C.	RULE TYPE	NAME	DESCRIPTION
13	Custom4	Manual validation rule	SwitchTypeForFlow	Requires the field must be defined in Custom4 table
14	Custom1 hierarchy	Automated rule	Set default value for AggregationWeight field in Custom1 hierarchy table	Requires the Custom4 table's SwitchTypeForFlow field to have a value.
15	Custom2 hierarchy	Automated rule	Set default value for AggregationWeight field in Custom2 hierarchy table	Sets the default value for the AggregationWeight field in the Custom1 hierarchy table.
16	Custom3 hierarchy	Automated rule	Set default value for AggregationWeight field in Custom3 hierarchy table	Sets the default value for the AggregationWeight field in the Custom2 hierarchy table.
17	Custom4 hierarchy	Automated rule	Set default value for AggregationWeight field in Custom4 hierarchy table	Sets the default value for the AggregationWeight field in the Custom3 hierarchy table.

Known limitations

Error management

When using the add-on export and import procedures on a data set containing a significant number of integrity errors (more than ~100), the add-on cannot fix these errors automatically. It is recommended to fix them in EBX® before using the Hyperion import and export procedures.

Migration procedures

- Actions on duplicated data sets when restoring are not supported.
- There is not an option to decide whether to restore, or backup data set mappings.
- Tracking restore traceability is not supported.

Release Notes

- [Version 4.6.0](#)
- [All release notes](#)

Version 4.6.0

- [New features](#)
- [Changes in Functionality](#)
- [Changes to third-party libraries](#)
- [Closed issues](#)
- [Known issues](#)

New features

This release was updated to ensure compatibility with TIBCO EBX® version 5.10.0 and the Add-ons Bundle version 4.6.0

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

The following third-party libraries were updated:

- The Apache Commons IO library was updated to version 2.20.0.

- The Apache Commons Compress library was updated to version 1.28.0.

Closed issues

This release contains no closed issues.

Known issues

This release contains no known issues.

All release notes

- [Version 4.6.0](#)
 - [New features](#)
 - [Changes in Functionality](#)
 - [Changes to third-party libraries](#)
 - [Closed issues](#)
 - [Known issues](#)
- [Version 1.13.16](#)
 - [New features](#)
 - [Changes in Functionality](#)
 - [Changes to third-party libraries](#)
 - [Closed issues](#)
 - [Known issues](#)
- [Version 1.13.15](#)
 - [New features](#)
 - [Changes in Functionality](#)
 - [Changes to third-party libraries](#)
 - [Closed issues](#)
 - [Known issues](#)
- [Version 1.13.14](#)
 - [New features](#)

- Changes in Functionality
- Changes to third-party libraries
- Closed issues
- Known issues
- Version 1.13.13
 - New features
 - Changes in Functionality
 - Changes to third-party libraries
 - Closed issues
 - Known issues
- Version 1.13.12
 - New features
 - Changes in Functionality
 - Changes to third-party libraries
 - Closed issues
 - Known issues
- Version 1.13.11
 - New features
 - Changes in Functionality
 - Changes to third-party libraries
 - Closed issues
 - Known issues
- Version 1.13.10
 - New features
 - Changes in Functionality
 - Changes to third-party libraries
 - Closed issues
 - Known issues

- Version 1.13.9
 - New features
 - Changes in Functionality
 - Changes to third-party libraries
 - Closed issues
 - Known issues
- Version 1.13.8
 - New features
 - Changes in Functionality
 - Changes to third-party libraries
 - Closed issues
 - Known issues
- Version 1.13.7
 - New features
 - Changes in Functionality
 - Changes to third-party libraries
 - Closed issues
 - Known issues
- Version 1.13.6
 - New features
 - Changes in Functionality
 - Changes to third-party libraries
 - Closed issues
 - Known issues
- Release Notes 1.13.5
 - Product updates
- Release Notes 1.13.4

- Product updates
- Release Notes 1.13.3
 - Product updates
- Release Note 1.13.2
 - New features
 - Bug fixes
- Release Note 1.13.1
 - Featured update
- Release Note 1.13.0
 - Featured update
 - Bug fixes
- Release Note 1.12.1
 - Bug fixes
- Release Note 1.12.0
 - New features
 - Bug fixes
- Release Note 1.11.1
 - Bug fixes
- Release Note 1.11.0
 - New features
 - Bug fixes
- Release Note 1.10.3
 - Bug fixes
- Release Note 1.10.2
 - Bug fixes

- Warnings
- Release Note 1.10.1
 - Bug fixes
- Release Note 1.10.0
 - New features
 - Bug fixes
- Release Note 1.9.0
 - New features
 - Bug fixes
- Release Note 1.8.1
 - New features
 - Bug fixes
- Release Note 1.8.0
 - New features
 - Bug fixes
- Release Note 1.7.3
 - New features
 - Bug fixes
- Release Note 1.7.2
 - New features
 - Bug fixes
- Release Note 1.7.1
 - New features
- Release Note 1.7.0
 - New features
 - Bug fixes

- Release Note 1.6.3
 - UI improvements
 - Bug fixes
- Release Note 1.6.2
 - New features
 - UI improvements
 - Bug fixes
- Release Note 1.6.1
 - New features
- Release Note 1.6.0
 - New ADS (EPMA) data format
 - Data mapping
 - 'TIBCO EBX™ Add-on for Oracle Hyperion EPM' data space
 - Migration service
 - Miscellaneous
- Release Note 1.5.9
 - Bug fixes
- Release Note 1.5.8
 - Bug fixes
- Release Note 1.5.7
 - Bug fixes
- Release Note 1.5.6
 - Bug fixes
- Release Note 1.5.5
 - Bug fixes
- Release Note 1.5.4

- Bug fixes
- Release Note 1.5.3
 - Bug fixes
- Release Note 1.5.2
 - New predefined rules on schema
 - Bug fixes
 - Limitation
- Release Note 1.5.1
 - New update
- Release Note 1.5.0 fix 002
 - Bug fixes
- Release Note 1.5.0 fix 001
 - Update predefined rules
 - Bug fixes
- Release Note 1.5.0
 - Custom dimensions management
 - Export configuration
 - The configuration properties are now available from the export UI
 - Integration with the TIBCO EBX™ Match and Cleanse Add-on
 - Integration with the TIBCO EBX™ Rules Portfolio Add-on
 - Remove the sample data set for EBX® Add-on for Oracle Hyperion EPM extension
- Release Note 1.4.0
 - User interface improvement
 - New business rules
 - New properties applied to the configuration of the CSV export operation
- Release Note 1.3.0

- Data set extension
- Default data set mapping
- Data authoring
- Import and export traceability
- TIBCO EBX™ Rules Portfolio Add-on
- Release Note 1.2.1
 - New updates
 - Known limitations
 - Technical limitations
 - Known bugs
 - Bugs fixes
- Release Note 1.2.0
 - New updates
 - Known limitations
 - Technical limitations
 - Known bugs
 - Bugs fixes
- Release Note 1.1.2
 - New updates
 - Known limitations
 - Technical limitations
 - Known bugs
- Release Note 1.1.1
 - New updates
 - Known limitations
 - Technical limitations
 - Known bugs
- Release Note 1.1.0

- New updates
- Known limitations
- Technical limitations
- Known bugs
- Release Note 1.0.1
 - New updates
 - Bug fixes
- Release Note 1.0.0

Version 4.6.0

New features

This release was updated to ensure compatibility with TIBCO EBX® version 5.10.0 and the Add-ons Bundle version 4.6.0.

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

The following third-party libraries were updated:

- The Apache Commons IO library was updated to version 2.20.0.
- The Apache Commons Compress library was updated to version 1.28.0.

Closed issues

This release contains no closed issues.

Known issues

This release contains no known issues.

Version 1.13.16

New features

The `Servlet` version was updated to 3.1.

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

This release contains no changes to third-party libraries.

Closed issues

This release contains no closed issues.

Known issues

This release contains no known issues.

Version 1.13.15

New features

This release contains no new features.

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

The following third-party libraries were updated:

- The Apache Commons Compress library was updated to version 1.26.0.
- The Apache Commons IO library was updated to version 2.12.0.

Closed issues

This release contains no closed issues.

Known issues

This release contains no known issues.

Version 1.13.14

New features

The add-on was updated to ensure compatibility with JDK 17.

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

This release contains no changes to third-party libraries.

Closed issues

This release contains no closed issues.

Known issues

This release contains no known issues.

Version 1.13.13

Released: July 2023

New features

This release contains no new features.

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

The jQuery library was updated to version 3.6.4.

Closed issues

This release contains no closed issues.

Known issues

This release contains no known issues.

Version 1.13.12

Released: April 2023

New features

This release was updated to ensure compatibility with the TIBCO EBX® Add-ons Bundle version 4.5.17.

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

This release contains no changes to third-party libraries.

Closed issues

This release contains no closed issues.

Known issues

This release contains no known issues.

Version 1.13.11

Released: January 2023

New features

Instructions were added to the documentation that describe how to prevent library conflicts when deploying on WebLogic. See the *Preventing library conflicts on WebLogic* heading in the `index.html` file that is located in the following folder in the 4.5.16 Add-on Bundle: *documentation* > *hmfh_`<xxx>`*.

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

This release contains no changes to third-party libraries.

Closed issues

This release contains no closed issues.

Known issues

This release contains no known issues.

Version 1.13.10

Released: October 2022

New features

This release contains no new features.

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

This release contains no changes to third-party libraries.

Closed issues

A vulnerability needs to be fixed.

Known issues

This release contains no known issues.

Version 1.13.9

Released: August 2022

New features

This release contains no new features.

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

The Apache POI library was updated to version 5.2.2.

Closed issues

This release contains no closed issues.

Known issues

This release contains no known issues.

Version 1.13.8

Released: March 2022

New features

This release contains no new features.

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

The jQuery library was updated to version 3.6.0.

Closed issues

This release contains no closed issues.

Known issues

This release contains no known issues.

Version 1.13.7

Released: December 2021

New features

This release contains no new features.

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

This release contains no third-party library updates.

Closed issues

This release contains the following closed issues:

- Validate the **Alias** setting inputs.

Known issues

This release contains no known issues.

Version 1.13.6

Released: August 2021

New features

This release contains no new features.

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

This release contains the following third-party updates:

- Apache Commons Compress to version 1.21.

Closed issues

This release contains no closed issues.

Known issues

This release contains no known issues.

Release Notes 1.13.5

Released: January 2021

Product updates

Imported XML files can no longer declare a Document Type Definition (DTD) element. The add-on will throw an error if this element is included.

Release Notes 1.13.4

Release Date: September 18, 2020

Product updates

This release contains the following updates:

- Support has been updated to include the OpenJDK8 and OpenJDK11 libraries.
- The jQuery library was updated to v3.5.0 and Apache POI to 4.1.2.

Release Notes 1.13.3

Release Date: June 23, 2020

Product updates

The jQuery library was updated to version 3.4.0 and the log4j library version 1.12.7 was removed.

Release Note 1.13.2

Release Date: December 10, 2019

New features

The **Export** service now exports Boolean field values correctly when a field value is TRUE or FALSE.

Bug fixes

- An unexpected value is found in the exported file if the value of a String field is set to 'TRUE' or 'FALSE'.

Release Note 1.13.1

Release Date: June 20, 2019

Featured update

The add-on has been updated to ensure compatibility with the TIBCO EBX® 5.9.4 release.

Release Note 1.13.0

Release Date: October 26, 2018

Featured update

The EBX® Add-on for Oracle Hyperion EPM has undergone significant updates to ensure compatibility with the EBX® 5.9.0 GA release.

Bug fixes

- A redundant line appears on top of the extension tab.

Release Note 1.12.1

Release Date: April 6, 2018

Bug fixes

- The validation rule on constraint does not work.

Release Note 1.12.0

Release Date: January 31, 2018

New features

This release contains the following new features:

- The 'Extract' service is now available to extract data on a financial data set based on the stored configuration data.
- A filter component allows you to filter the extract result.

Bug fixes

- NullPointerException occurs when generating extensions for some field types.

Release Note 1.11.1

Release Date: September 6, 2017

Bug fixes

- Zooming out the screen below 80% from the Comparing dimension page breaks the form.
- Running Compare through workflow on the regular Comparison service mode displays an unexpected Comparison screen.
- The generated mapping result is wrong if Mapping data is deleted when running the service.
- Running 'Copies to use TIBCO EBX™ Match and Cleanse Add-on' on the Matching data space whose label contains cross-scripting displays an unexpected pop-up.
- Child data set mapping whose label contains cross-scripting displays incorrectly when running the Generates mapping data service.
- Warning messages are misplaced while running the 'Generates mapping data' service with missing data configuration.
- Invalid application is shown incorrectly when users run the 'Generates mapping data for extra custom' service.
- A blank pop up appears when users create child data set mapping from a non-existent parent data set mapping.
- Error message is misplaced when users create child data set mapping from a non-existent parent data set mapping.
- Detail documentation is not aligned with EBX®'s.
- In the detail documentation, the options of the 'Is extension table' field should start with capital letters.
- An unexpected error pop up displays when running the 'Generate extension mapping' service on the Custom4 dimension.
- There is no information given when hovering over on the 'Inherit', 'Override' and 'Wizard' buttons.
- Technical value is displayed in some fields of the 'Extension mapping generation result' screen.
- The first letters of some screen headers are not capitalized.
- An unclear error message is displayed when users delete an 'Application' mapping during 'Export' progress.
- A record label whose description contains cross-scripting is displayed incorrectly in the Description, and Hierarchy table.

- Some error messages should be more exact and user-friendly.
- Some labels and tooltips are not aligned with the user guide.
- No error message is displayed when users restore a data set whose name has already been used.
- The error message displayed when users restore a non-existent data is not user-friendly.
- The wrong model is returned when users import an extension model from module.
- A blank page displays when users run the 'Generate mapping data set' service with a non-existent Mapping data set.
- Not all features of tables in the mapping data set are available when users access it by clicking the 'Preview' button.
- Some behavior is incorrect when users access a record in the Currency table.
- Some tooltips are not displayed in a user-friendly manner in the mapping data set.
- Some tooltips are not displayed in a user-friendly manner in the Hyperion configuration data set.
- Some bullets are not displayed in a user-friendly manner in the user guide.
- Some spelling mistakes are detected in the user guide.
- Unexpected error pop up is displayed when users run the Compare service without an Application in Mapping data set.

Release Note 1.11.0

Release Date: June 14, 2017

New features

- Table tooltips have been fixed that were not available, or displayed incorrectly under the 'Generate mapping data' service.
- The suggested values of the 'Items per page' fields are now use Integer when setting a workflow's 'Compare service'.
- During export an error message is raised on the 'Choose dimension screen' when the 'Header's label of description' property contains a delimiter.
- A new button was added to each record on the 'Mapping table for Dimension' table to set default mapping data.

Bug fixes

- An error message is displayed when running the 'Purge' service if the Report data set contains data for an import or export result.
- The 'Extension' tab should be hidden in the 'History' table when the data set extension is not defined.
- Perspective display does not match current settings under 'Administration'.
- An unexpected error message displays under the Filter by date fields when running the 'Purge report' service.
- On the Compare result page, data cannot be exported using 'Export to excel' when the data set label contains cross-scripting.
- The exported data is not correct when exporting a 'null' record.
- The Comparison result cannot be accessed through a perspective despite proper configuration.
- Data is not exported to an Excel file when exporting from a child recursive node.
- A warning message is displayed in the kernel log when starting server.
- Some tooltips are not user-friendly in the 'Basis' table under 'Hyperion configuration'.
- When returning from next page all 'data set(s) selected for the copy' are checked while 'Select all' is not checked.
- A blank page displays when users run 'Generates mapping data' with a Mapping data set that no longer exists.
- Running 'Execute rules' under financial data set causes a NullPointerException in ebx-addon-hmfh.log file.
- The Comparison configuration cannot be accessed through Workflow.

Release Note 1.10.3

Release Date: May 19, 2017

Bug fixes

- An exception occurs when deploying on an application server that strictly validates web.xml.

Release Note 1.10.2

Release Date: April 18, 2017

Bug fixes

- Unauthorized access to resources is possible.

Warnings

- Customers are strongly advised to upgrade to the latest version which patches the security issue.

Release Note 1.10.1

Release Date: March 31, 2017

Bug fixes

- [Documentation](#) There are grammatical errors in the names of compare classes.
- DimensionsComparison.getComparisonService does not work when being called through a record on a workflow.

Release Note 1.10.0

Release Date: March 20, 2017

New features

- A new API is available to get the user interface for the Compare service.

Bug fixes

- Error messages for the Description must start with 'Non_Prod' for both ancestors of 'Non_Prod_types' and itself.
- There are some grammatical errors on Import/Export service tool tips when the language is set to French.
- An error occurs when backing up an EBX® Add-on for Oracle Hyperion EPM data space that has a custom mapping.
- The Boolean field value is always false on Excel files even when the display format is set to true.
- The Generate extension mapping service should be hidden in the History table.
- Un-mapped MDM field(s) still loads after launching the Generate mapping data service for the second time.

- An unexpected error occurs after exporting when inputting a value smaller than '-9' into the Generate levels dynamically field.
- An error occurs when running Generate mapping data with no hierarchy table selected in the Mapping tables.
- Files cannot be imported or exported when a record name contains the special character '|'.

Release Note 1.9.0

Release Date: January 23, 2017

New features

- Mapping data can be generated for custom dimensions.
- Mapping data can be generated for extension fields.
- It is now possible to generate new mapping data for dimensions and applications.
- Indexes for field 'descriptions' (table 'Description') and 'order' (table 'Hierarchy') were added.
- The 'DimensionClass' can now be mapped.
- A default configuration for 'Update dimension and field order' on the import service is now available.
- Predefined rules data has been brought into alignment with the current version of the TIBCO EBX™ Rules Portfolio Add-on.
- New configuration options such as: check special character 'yes/no' and filter special character at 'EPMA' export time are available.
- It is possible to expand/collapse one level for all nodes on the 'Compare' service.
- A new option is available to specify whether to export 'ALL' members of Custom dimensions.
- The mapping data set in the 'Choose data set mapping' is viewable screen without specific permission.
- The levels of 'Essbase' rainbow file can now be handled dynamically.
- It is now possible to export from multiple nodes on the hierarchy view.
- You can now switch the display of boolean fields in exported Excel file between 'T/F' and 'Y/N'.
- The 'Export for this node' service is available from 'EBX® Add-on for Oracle Hyperion EPM' services menu option in the Action menu and when viewing an opened record.

Bug fixes

- Incorrect behavior when executing 'Send to end' under the 'Services' menu in the node which does not have a parent.
- Broken form occurs when a user clicks on the 'Import/Export' services if the label of data set mapping contains the '<' character.
- Broken form occurs on the export result screen if the label of the financial data set contains the '<' character.
- The 'Extension' field is not exported in a node which is a Default parent of other nodes when exporting 'HFM' with 'CSV' type.
- The temporary files of EBX® Add-on for Oracle Hyperion EPM import/export should be automatically deleted.
- The system does not show a message to inform users when running the export service after deactivating data set mapping.
- The execute button does not work after creating a new data set with the 'Unique name' that already exists.

Release Note 1.8.1

Release Date: December 16, 2016

New features

- The 'Comparison mode' of 'childOrder' is set to 'default'.

Bug fixes

- The node order is incorrect when importing into a child data set that has overwritten nodes.

Release Note 1.8.0

Release Date: October 12, 2016

New features

- It is now possible to set the field order for exported ADS files.
- It is now possible to export from a node in the hierarchy view.
- You can now configure Compare to display only the results related to a specified node.
- It is now possible to export comparison results.

- Pagination is now available for Comparison results.
- A scrollbar was added to allow you to scroll to deep level nodes.
- Each node in the Comparison result now has a tooltip to show its description and full path.

Bug fixes

- The 'EBX® Add-on for Oracle Hyperion EPM Export - Preference for export' ADS file page is still loaded when selecting and then deselecting export ADS type.
- An exception occurs in the log file when exporting an Excel file with the single quote character.
- A message should be displayed when inputting wrong data or nothing for the Application and Dimension fields in Data workflow.
- It is impossible to import the file which was previously exported using the extension for the 'Custom5' table.
- The Error log doesn't describe clearly when the TIBCO EBX™ Rules Portfolio Add-on is not yet installed.
- A blank page displays when exporting the newly created application.
- A redundant Import or Export configuration screen displays after coming back from the Choose dimension screen.
- Default mapping is faulty because of the duplication mapping for the 'Agg2gregation' field of the Custom1 dimension.

Release Note 1.7.3

Release Date: July 8, 2016

New features

- Integration with the TIBCO EBX™ Information Governance Add-on allows labels to be modified.
- Compare can be configured to display only the Comparison result on the workflow user task.
- The 'Show relationship record' service is available to show hierarchy records in a hierarchy view.
- The path of the 'isPrimary' field into 'HmfhDefaultPath' is provided.
- It is now possible to configure validation regular expressions for Names.
- A 'ConsolidationAccountType' can now be configured.
- The member formula ASO now supports up to 2000 characters.

Bug fixes

- An error message is displayed when importing the file which was previously exported using the extension for the 'Setting' table.
- The extension tab in the extra custom hierarchy table is missing.
- The 'isPrimary' field in the hierarchy table in the extra custom dimension is missing.
- Exception occurs when you export an EPMA application for the Custom dimension which contains the 'ALL' member.

Release Note 1.7.2

Release Date: June 10, 2016

New features

- An 'Extension' tab now displays on Hierarchy and Setting tables, which displays record extension data.
- The Version and Currency tables now contain an 'Extension' tab that displays extension data.
- A trigger will automatically fill the primary key for an extension record on creation.

Bug fixes

- The 'Custom2HierarchyExtension' table contained an incorrect reference when importing 'sampleExtensionTemplate.xsd'.

Release Note 1.7.1

Release Date: May 19, 2016

New features

- Display Compare service on workflow - user task.
- Display Export service on a snapshot.
- Display 'Shared information' as a tab.

Release Note 1.7.0

Release Date: April 13, 2016

New features

Compare service

It is now possible to compare two versions of the same dimension using the 'Compare' service. This service is located at the data set level and allows you to select a dimension from a source and target data set for comparison. The data sets can be from previously taken snapshots, or current data spaces.

Bug fixes

- The creation of predefined rules leads to redundant rule execution that causes an error.
- A blank page displays after exporting when a record in the 'Mapping MDM table' no longer exists.
- A fatal error occurs in a mapping data set when a 'Mapping table' record is deleted.
- EBX® Add-on for Oracle Hyperion EPM services are still available even when a data set is hidden.

Release Note 1.6.3

Release Date: January 18, 2016

UI improvements

- Aligned warning messages in the 'Extension' tab.

Bug fixes

- The 'Member Formula' field's French label in the 'Account' table is not correct.
- The 'Send to end' service on the Custom5/Member table and the 'Switch active' service on Custom5/Hierarchy table of the 'Extra Custom' data set are missing.
- No error message displays when the 'From' and 'To' fields are blank in the 'Purge report' service.
- If an email's subject is blank, the 'Export/Import' service displays a blank page when running in 'Run in background' mode.

Release Note 1.6.2

Release Date: November 3, 2015

New features

- The following three rules that use the TIBCO EBX™ Rules Portfolio Add-on scripting language have been added:
 - Rule 04: When account name ends with C, 'IsCalculated' is set to 'Yes' - Use Script.
 - Rule 33: Automatically set 'Exchange rate type' to None - Use Script.
 - Rule 34: When account is an 'intercompany' account, the plug account must be defined - Use Script.

UI improvements

- Updated the tooltip button position in the Import and Export screens.
- Removed redundant results on the 'Purge results' screen when only one result type is selected.

Bug fixes

- An 'Alert' dialog displays when the 'Description' field contains an HTML <script> tag.
- The 'Export preference' value is not saved when updating the 'Mapping field' table's 'Section' field value for the first time.

Release Note 1.6.1

Release Date: September 16, 2015

New features

- Align predefined rules to TIBCO EBX™ Rules Portfolio Add-on 1.3.0

Release Note 1.6.0

Release Date: August 24, 2015

New ADS (EPMA) data format

- The 'Dimension' and 'Mapping field' tables have a new property called 'Flat ADS name' that manages the ADS (EPMA) data format.
- New possible values are available for the ADS (EPMA) data format in the 'Mapping field' table's 'Field type' property such as: Scale, Symbol, etc.
- Integration of the ADS (EPMA) data format lead to the following new data values in the predefined data mapping:

- 'EPMA' application, 'MDM table' and 'MDM field' for new tables and fields handled by the ADS format.
- 'Dimension', 'Application dimension', and data mapping through the 'Mapping table' and 'Mapping field' tables.
- A new data group called 'ADS properties' has been added to the dimension setting tables.
- The following new dimensions have been created and enriched: Version, Version description, Version hierarchy, Version setting, Currency hierarchy, Currency setting, Alias hierarchy and Alias setting.

Data mapping

- The 'MDM field' table has a new property that allows you to declare a default value for boolean fields.
- The validation report has been enriched to detect warnings and errors in the data mapping configuration ('Validate' service on the data set containing the data mapping).
- Association views have been added to facilitate navigation between the different tables used for data mapping configuration.

'TIBCO EBX™ Add-on for Oracle Hyperion EPM' data space

- The 'Application' table is used to get the list of applications that are available for data mapping.
- The 'Preference' table is used to save data mapping preferences when exporting financial data in the ADS (EPMA) data format.
- The 'Dimension association' table saves dimension information during the ADS (EPMA) import process in order to reuse it during export.

Migration service

- The 'Migrate mapping data set' service, located in the 'EBX® Add-on for Oracle Hyperion EPM' data space, updates data mapping values with new EPMA statements, and enriches the validation report service with new controls
- The 'ebx-addon-hmfh-mapping' archive has been enriched with the ADS (EPMA) statements. It also contains other modifications applied on existing data mappings to eliminate potential warnings and errors raised by the validation report service.

Miscellaneous

- You can backup and restore multiples data spaces.

- The data model used as an extension sample has changed from 'ebx-addon-hmfh-extension.xsd' to 'sampleExtensionTemplate.xsd'.
- You can adapt the import process to manage missing values.

Release Note 1.5.9

Release Date: July 31, 2015

Bug fixes

- Update archive ebx-addon-hmfh-matching-configuration for compliance with TIBCO EBX™ Match and Cleanse Add-on 1.7.0

Release Note 1.5.8

Release Date: July 21, 2015

Bug fixes

- One or more empty cells in the exported Excel file is not BLANK type

Release Note 1.5.7

Release Date: April 2, 2015

Bug fixes

- A white page is displayed when clicking on view history of tables in financial data set

Release Note 1.5.6

Release Date: March 25, 2015

Bug fixes

- When the EBX® Add-on for Oracle Hyperion EPM data model is not extended, the tab Extension is hidden

Release Note 1.5.5

Release Date: March 17, 2015

Bug fixes

- When the EBX® Add-on for Oracle Hyperion EPM data model is extended, the UI is now displayed with a tab (rather than a simple link)

Release Note 1.5.4

Release Date: February 12, 2015

Bug fixes

- Rule "Set default value for field 'Aggregation weight'" does not work in case the record is created by an automated rule.

Release Note 1.5.3

Release Date: January 26, 2015

Bug fixes

- Set default value for hidden fields in duplication.

Release Note 1.5.2

Release Date: October 10, 2014

New predefined rules on schema

- Field IsCalculate must be defined on Account.
- Field IsCalculate must be defined on Custom1.
- Field SwitchSignForFlow must be defined on Custom1.
- Field SwitchTypeForFlow must be defined on Custom1.
- Set default value for AggregationWeight on Custom1 hierarchy.
- Field IsCalculate must be defined on Custom2.
- Field SwitchSignForFlow must be defined on Custom2.
- Field SwitchTypeForFlow must be defined on Custom2.
- Set default value for AggregationWeight on Custom2 hierarchy.
- Field IsCalculate must be defined on Custom3.
- Field SwitchSignForFlow must be defined on Custom3.
- Field SwitchTypeForFlow must be defined on Custom3.

- Set default value for AggregationWeight on Custom3 hierarchy.
- Field IsCalculate must be defined on Custom4.
- Field SwitchSignForFlow must be defined on Custom4.
- Field SwitchTypeForFlow must be defined on Custom4.
- Set default value for AggregationWeight on Custom4 hierarchy.

Bug fixes

- Description is not inherited after click to inherit button.

Limitation

- Must be submit after clicking to inherit button to get the inherited Description.

Release Note 1.5.1

Release Date: September 12, 2014

New update

- Ability to add integer number from 0 to 100 for field 'percent consol' in Consolidation method.

Release Note 1.5.0 fix 002

Release Date: August 20, 2014

Bug fixes

- Missing aggregation weight of children of #root on custom dimension when export excel file.

Release Note 1.5.0 fix 001

Release Date: August 7, 2014

Update predefined rules

- Rule 1: Update rule to carry description.
- Rule 3: Create new rule to work with child data set.
- Rule 4: Update simple expressions.
- Rule 6: Update rule in case "create", "manual validation".
- Rule 10: Update rule not to be case sensitive.

- Rule 11/12: Update simple expressions.
- Rule 19: Update simple expressions.
- Rule 20, 21, 22: Update rule to work with the grand child.
- Rule 24, 25: Update simple expressions.
- Rule 26: Update simple expressions.
- Rule 28: Add manual rule check with warning for this rule.

Bug fixes

- Blank page on export when TIBCO EBX™ Add-on for Oracle Hyperion EPM has validation error.
- Always select default data set mapping on Export.

Release Note 1.5.0

Release Date: June 24, 2014

Custom dimensions management

- It is now possible to create any number of new custom dimensions. In the previous version, the maximum number of custom dimensions is limited to four

Export configuration

- The configuration for the 'flat CSV file' is extended with the property 'Is export custom dimension order' (CSV item '!CUSTOM_ORDER=Custom1;Custom2;...').

The configuration properties are now available from the export UI

- All configuration properties are now available from the export UI directly. By default, their values are retrieved from the configuration declared in the Administration tab.

Integration with the TIBCO EBX™ Match and Cleanse Add-on

- It is now possible to use the TIBCO EBX™ Match and Cleanse Add-on features on financial data. A new configuration property 'Use TIBCO EBX™ Match and Cleanse Add-on' allows you to declare if the TIBCO EBX™ Match and Cleanse Add-on is available or not.
- The TIBCO EBX™ Match and Cleanse Add-on is used to de-duplicate and clean up financial data.

Integration with the TIBCO EBX™ Rules Portfolio Add-on

- The TIBCO EBX™ Rules Portfolio Add-on is now integrated with EBX® Add-on for Oracle Hyperion EPM. The properties 'Business rules portfolio activated' and 'Permission rules portfolio activated' allow you to decide if the Rules portfolio is used or not.
- The TIBCO EBX™ Rules Portfolio Add-on is used to enforce a transparent management of the business rules and permission rules applied to financial data (rules portfolio, rules configuration, traceability of the rules execution).

Remove the sample data set for EBX® Add-on for Oracle Hyperion EPM extension

- Since 1.5.0, the sample data set for EBX® Add-on for Oracle Hyperion EPM extension: data set 'EBX® Add-on for Oracle Hyperion EPM Extension' is removed. It is necessary to delete the data set 'EBX® Add-on for Oracle Hyperion EPM Extension' to avoid error when import/export archive.

Release Note 1.4.0

Release Date: March 6, 2014

User interface improvement

- Progress bars are now available during the execution of these services: Import, Export, Backup, Restore and Migrate data set mapping.

New business rules

- The following business rules have been added into the rules portfolio for EBX® Add-on for Oracle Hyperion EPM:
 - For accounts, if Data Type is set to any value except Currency, then Exchange Rate Type must be None
 - If account is intercompany account (IsICP = 'Y') then the plug account must be defined
 - Entity default currency must be defined

New properties applied to the configuration of the CSV export operation

- Three new properties are available for configuring the export of a flat CSV file: 'Header's label of default parent', 'Header's label of description', 'Is export hierarchy header'

Release Note 1.3.0

Release Date: January 6, 2014

Data set extension

- When a data set extension is used to extend the prebuilt financial data model, the import and export procedures manage the extended tables and fields.

Default data set mapping

- A default data set mapping is now configured at the level of every financial data set.

Data authoring

- The modification of the description of a Dimension is now possible in the UI of the dimension, previously it was required to open the table 'Description'.

Import and export traceability

- The execution of the import and export procedures are logged in a dedicated data set. It allows a full and detailed traceability of the operations. A purge service allows to clean up the report on demand.

TIBCO EBX™ Rules Portfolio Add-on

The TIBCO EBX™ Rules Portfolio Add-on is now able to manage the EBX® permission rules applied on the services, data sets, tables, records and fields. Two new types of rules are available:

- 'Access permission rule' to evaluate the access permission on a data set, table, field or record.
- 'Action permission rule' to evaluate the permission for a service.

These new types of Data Element Concepts are available to allow the configuration of the permission rules:

- Service
- Data set
- Field

These new predefined business rules are available to get 'True' or 'False' results directly. By combining this prebuilt assertion rules with the 'Simple expression' configurations, it becomes possible to create new assertions rules based on the fields of a table:

- Assertion true(Manual validation)

- Assertion true(Manual validation)
- Assertion true(Table set)
- Assertion false(Manual validation)
- Assertion false(Validation)
- Assertion false(Table set)

These new predefined permission rules are available:

- Hidden service
- Disable service
- Enable service
- Hidden the data set
- Permission read-only for the data set
- Permission read-write for the data set
- Hidden table
- Permission read-only for the table
- Permission read-write for the table
- Hidden field
- Permission read-only for the field
- Permission read-write for the field
- Hidden record
- Permission read-only for the record
- Permission read-write for the record

New services:

- 'D.E.C. creation': this service allows to create automatically the D.E.C.(s) based on a selected data space and data set.
- 'Create demo rule(s)': this service creates at set of sample rule(s).
- 'TIBCO EBX™ Rules Portfolio Add-on migration': this service allows to update a former rules configuration to this new version of the add-on.

Release Note 1.2.1

Release Date: November 4, 2013

New updates

- Set reference tables as Read only
- Add record to select 'All data spaces' and 'All data sets' on table Data set
- Allows to set 'Is active' on Rule execution
- Select multi-datasets to execute a Rule on 'Rule execution'
- Add constraint on table 'Rule execution' to confirm Rule or Rules set is defined
- Defined validity date and expiry date on Rule execution
- Auto fill 'Name' and 'Error message' when create new rule if it is empty
- Filter D.E.C. attachment as table by checking value of field 'Object class' in group 'Implementation' of a Rule

Known limitations

- Field 'Label' in HFM/Essbase import file must be unique
- When executing Rule 2, after creating a record from APMCONTRIB or APMCONTRIBDKK node, system shows error message

Technical limitations

- Only XML and Excel imports are supported
- Only XML, Excel, and CSV exports are supported
- Only Hyperion 'Financial Management', 'Essbase', and 'Planning' are supported
- Changing the XML format of 'Financial Management' is unsupported
- Import/export services for Hyperion Financial data do not work with extension data sets
- The migration procedure does not work with extension data sets. The data in extension data sets must be manually backed up and restored
- The financial parent and child data sets use the same extension data set

Known bugs

- When a record is deleted, the hierarchy record is not deleted.

Bugs fixes

- When creating a new D.E.C for a specific data set, all rules are not applied for that data set.

Release Note 1.2.0

Release Date: October 3, 2013

New updates

- RC Business rules portfolio management for EBX® Add-on for Oracle Hyperion EPM.
- New Business rule type "Table set Rule". Add a Service on Table called "Execute rules" and service on DataSet "Execute rules" to run all the WF Rules
- On TIBCO EBX™ Add-on for Oracle Hyperion EPM / Basis, add field 'Export order for hierarchy' to select the hierarchy order when export

Known limitations

- Field 'Label' in HFM/Essbase import file must be unique
- When executing Rule 2, after creating a record from APMMCONTRIB or APMMCONTRIBDKK node, system shows error message

Technical limitations

- Only XML and Excel imports are supported
- Only XML, Excel, and CSV exports are supported
- Only Hyperion 'Financial Management', 'Essbase', and 'Planning' are supported
- Changing the XML format of 'Financial Management' is unsupported
- Import/export services for Hyperion Financial data do not work with extension data sets
- The migration procedure does not work with extension data sets. The data in extension data sets must be manually backed up and restored
- The financial parent and child data sets use the same extension data set

Known bugs

- When a record is deleted, the hierarchy record is not deleted.
- When creating a new D.E.C for a specific data set, all rules are not applied for that data set.

Bugs fixes

- When export HFM CSV, there are some cases the field Default parent doesn't contain value.
- When export Planning CSV, if field value contains separator character, the protection by quote is not correct.

Release Note 1.1.2

Release Date: June 6, 2013

New updates

- Beta version of API for EBX® Add-on for Oracle Hyperion EPM.
- An option has been added in the export interface for whether or not to export `#root` from the hierarchy.
- It is now possible to rename the column names of 'Name', 'Parent', and 'Child' for HFM CSV flat files.
- EBX® Add-on for Oracle Hyperion EPM tables can now be converted to reusable types.
- A default language can now be defined when manually creating the description for a dimension.
- A default selection when using the export service can now be configured.
- The data set 'EBX® Add-on for Oracle Hyperion EPM Extension' has been added.

Known limitations

- Field 'Label' in HFM/Essbase import file must be unique

Technical limitations

- Only XML and Excel imports are supported
- Only XML, Excel, and CSV exports are supported
- Only Hyperion 'Financial Management', 'Essbase', and 'Planning' are supported
- Changing the XML format of 'Financial Management' is unsupported
- Import/export services for Hyperion Financial data do not work with extension data sets
- The migration procedure does not work with extension data sets. The data in extension data sets must be manually backed up and restored
- The financial parent and child data sets use the same extension data set

Known bugs

- When a record is deleted, the hierarchy record is not deleted.

Release Note 1.1.1

Release Date: May 20, 2013

New updates

- Beta version of API for EBX® Add-on for Oracle Hyperion EPM.
- An option has been added in the export interface for whether or not to export `#root` from the hierarchy.
- It is now possible to rename the column names of 'Name', 'Parent', and 'Child' for HFM CSV flat files.
- Hyperion tables can now be converted to reusable types.
- A default language can now be defined when manually creating the description for a dimension.
- A default selection when using the export service can now be configured.
- The data set 'EBX® Add-on for Oracle Hyperion EPM Extension' has been added.

Known limitations

- Field 'Label' in HFM/Essbase import file must be unique

Technical limitations

- Only XML and Excel imports are supported
- Only XML, Excel, and CSV exports are supported
- Only Hyperion 'Financial Management', 'Essbase', and 'Planning' are supported
- Changing the XML format of 'Financial Management' is unsupported
- Import/export services for Hyperion Financial data do not work with extension data sets
- The migration procedure does not work with extension data sets. The data in extension data sets must be manually backed up and restored
- The financial parent and child data sets use the same extension data set

Known bugs

- When a record is deleted, the hierarchy record is not deleted.

Release Note 1.1.0

Release Date: April 2, 2013

New updates

- Now only hierarchy nodes that have 'Active' set to 'Yes' are exported. Previously, this value was not taken into account.

- Import and export for 'Planning' have been added.
- Add-on has been separated into a business data set and a mapping data set, and a Hyperion configuration data set has been added.
- Support for exporting localized descriptions is now included.
- Descriptions are now displayed according to the locale of EBX®.
- Export and import operations are now performed in the background. An email containing the results is sent once the process has completed.
- It is now possible to backup and restore financial data.
- Import and export of HFM spreadsheets is now supported.
- Support has been added to create a child data set for data set mapping.
- TIBCO EBX™ Add-on for Oracle Hyperion EPM - Logging is now supported.
- Export of HFM CSV is now supported.

Known limitations

- Field 'Label' in HFM/Essbase import file must be unique

Technical limitations

- Only XML and Excel import are supported
- Only XML, Excel, and CSV export are supported
- Only Hyperion 'Financial Management', 'Essbase', and 'Planning' are supported
- Changing 'Financial Management' XML format is unsupported

Known bugs

- When a record is deleted, the hierarchy record is not deleted.

Release Note 1.0.1

Release Date: November 26, 2012

New updates

Allow description changes when editing a member table

When editing a member, such as an account or entity, the user is now able to edit its description in the same form. Previously, the description was inherited, and in order to change it, the user had to access it from the description table.

Bug fixes

- Issue when deleting a node from the hierarchy.

When the user deletes a record in a hierarchy, the record's description remains in the description table.

- Ordering node not set by default.

In the table 'Entity', the field 'orderBy' is not set by default. It should be set to 'childOrder'.

- Export service does not work properly when application not mapped.

During an export, when there is no mapping data for the application in 'Administrative Data', the 'Choose dimension(s)' button is still active. If this button is clicked, the page reloads no message displayed to the user.

- New function: backup/restore financial data

Release Note 1.0.0

Release Date: November 9, 2012

Java API Reference

- [TIBCO EBX® Add-ons 4.6.0 Java API](#)