



Configuring MITS Discover to Display Data from Two Related Hypercubes

Introduction

MITIS Discover flash screens have historically been limited to displaying data from only one Hypercube at a time. A new feature has been introduced in version 7.0 that provides a method by which data from two different (but related) Hypercubes can be displayed in a single flash screen.

This document is designed to guide you in how to use this feature. We will walk you through the necessary steps from start to finish using a Sales Hypercube and a Bookings Hypercube as an example throughout.

Requirements

Before you begin...

You will need:

- MITIS Discover 7.0.0 or greater
- Two related MITIS Discover Hypercubes within the same MITIS Discover operational account
- At least 1 identifier that is common to both MITIS Discover Hypercubes (the identifier abbreviations do not have to be the same)

NOTE: The identifiers you wish to link between Hypercubes must also have common drill-down "paths". For example, if Hypercube 1 has an identifier, "customer", accessed under the identifier "branch", Hypercube 2 must have a similar drill-down path, where the customer identifier is accessible under a branch identifier.

For more information on identifiers (including their abbreviations) see *Assigning Data Element Roles* in the "MitsMaker User Guide".

The Example Scenario

To better illustrate the use of this feature, we will be using the following example scenario throughout all of the steps: The Vice President of Sales would like a flash screen that shows him not only the current sales so far this year, but also the sales that are currently booked. This information exists on two different Hypercubes: One is named SALES, and the other is named BOOKINGS.

To solve this problem, we will need to do 3 things:

1. Create a reference to match the identifier abbreviations between the Hypercubes by adding an item to the `MITIS.APPS.IDENTS.XREF` file
2. Configure MITIS Discover columns to pull data from one Hypercube into flash screens created from the other. If you have MitsMaker, you will create a new column template. Otherwise, you will use the Column Maintenance Wizard within MitsWeb to create a new column.
3. Add the new column to the desired flash screen

Matching Identifiers

In order to use columns from an external Hypercube, MITS Discover needs to know the relationship that exists between the identifiers in the separate Hypercubes. These relationships can be specified by creating an item in the file named MITS.APPS.IDENTS.XREF located in the MITS Discover Operational Account (often named MITS.SYSTEM).

NOTE: If MITS.APPS.IDENTS.XREF does not exist in the MITS Discover Operational Account, create the file before continuing.

Each item in the file associates one Hypercube with one other Hypercube and will include two attributes that form a parallel array.

- **Item ID:** The two Hypercube names delimited by an underscore, in this format: Hypercube1_Hypercube2
- **Attribute 1:** List of identifier abbreviations (separated by value marks) in Hypercube1 that correspond to identifiers in Hypercube2. This is a parallel array to Attribute 2.
- **Attribute 2:** List of identifier abbreviations (separated by value marks) in Hypercube2 that correspond to identifiers in Hypercube1. This is a parallel array to Attribute 1.

Example:

```

ID: SALES_BOOKINGS
001: CyPyW
002: CNyPNyWH

```

WARNING: Don't include all identifier abbreviations from both Hypercubes unless ALL of them match up. Only the identifier abbreviations that match up should be included.

The table below shows how we used a chart to determine which identifiers have matching relationships.

Identifier Abbreviations from SALES	Identifier Name	Identifier Abbreviations from BOOKINGS
C	Customer	CN
P	Product	PN
PL	Product Line	-----
W	Warehouse	WH
R	Region	-----

Create the Column Template

Before viewing information from a second Hypercube in a flash screen, columns have to be created to retrieve the information. This is done by creating a column template that uses the ACCUM.EXTERNAL function in the expression field of the column template. For this example, the column template will be created in Hypercube1: SALES.

The screenshot shows a dialog box titled "BOOKED.ORDERS.{EON} Properties". It contains several input fields: "Description" with the value "Booked Orders from BOOKINGS ({WHEN} {SPAN})", "Treatments" with "RO,", "Width" with "9", "Heading" with "Booked Orders.from BOOKINGS_{EON}_", "Group" with "Bookings by {CYCLE}", and "Expression" with "ACCUM.EXTERNAL(BOOKINGS,BOOKED.PRICE.EXT,{EON})". The "Expression" field is circled in red. At the bottom, there is a checkbox labeled "Calculate this template from other columns?" which is checked, and a "Close" button.

Similar in form and function to the ACCUM function call, the ACCUM.EXTERNAL function call differs in that it requires an additional argument in order to know which Hypercube to build the columns from.

The syntax for the ACCUM.EXTERNAL function is:

```
ACCUM.EXTERNAL(<appName>, <accumName>, {EON})
```

This function is passed three arguments:

1. **<appName>** - The name of Hypercube2 from the cross-reference (BOOKINGS in the example).
2. **<accumName>** - The name of the accumulator you want to use from Hypercube2. This is the actual name of the Data Pool item in Hypercube2 (BOOKED.PRICE.EXT in the example).
3. **EON** - The time span to use. Use the {EON} placeholder to create a column for each known eon in Hypercube1.

NOTE: A specific eon can also be used for single column creation, using the Column Maintenance Wizard within the MitsWeb interface.

Example:

```
ACCUM.EXTERNAL(BOOKINGS,BOOKED.PRICE.EXT, {EON})
```

This column expression instructs MITS Discover to use the BOOKED.PRICE.EXT accumulator from the BOOKINGS Hypercube to create a column for each eon in the SALES Hypercube.

NOTE: When modifying column templates, a create-app and a build wrap-up will be necessary to complete the changes. See the MitsMaker User Guide for more information.

Best Practices

This section lists recommendations that should be followed while configuring this feature.

Column Headings and descriptions should refer back to the associated Hypercube

Because these columns are generated from a different Hypercube, their headings and descriptions should reflect which Hypercube they are associated with. For example, consider a column that contains sales data that has been "booked" but not sold and is generated from the BOOKINGS Hypercube. It could be named "Booked Orders from BOOKINGS" followed by the appropriate eon description or placeholder.

Be aware of user security restrictions on both Hypercubes

MITS Discover security works across both Hypercubes. So, if a user only has access to one Hypercube, or is restricted from viewing specific information from the second Hypercube, the restricted information will not be displayed. Make sure that the users who need to view flash screens with external columns do not have restrictions that would prevent them from viewing the data you have intended them to see.

Prevent information mismatch due to build scheduling problems

If the Hypercubes are updated on different schedules, there may be a mismatch of information when only one Hypercube has been updated. Configure build schedules and comment, where appropriate (using the "Additional Column Information" feature), about potential gaps in the accuracy of the displayed information.

Make sure that the eons match

The two Hypercubes should have identical eons. For example, SALES and BOOKINGS are both transactional Hypercubes that each contain columns based on YEAR eons. If SALES was based on a fiscal year and BOOKINGS was based on a calendar year, the information might be mismatched if the fiscal year differs from the calendar year or if the fiscal periods don't match up to calendar months. Another example would be if one of them were to be a snapshot Hypercube (which only provides a single reference point, like CURRENT for an Inventory Hypercube), the data could be confusing to the users because of this eon mismatch.

Use a chart to help match identifier abbreviations

A chart can help in matching identifier abbreviations between two Hypercubes. The chart should be filled out following these three steps:

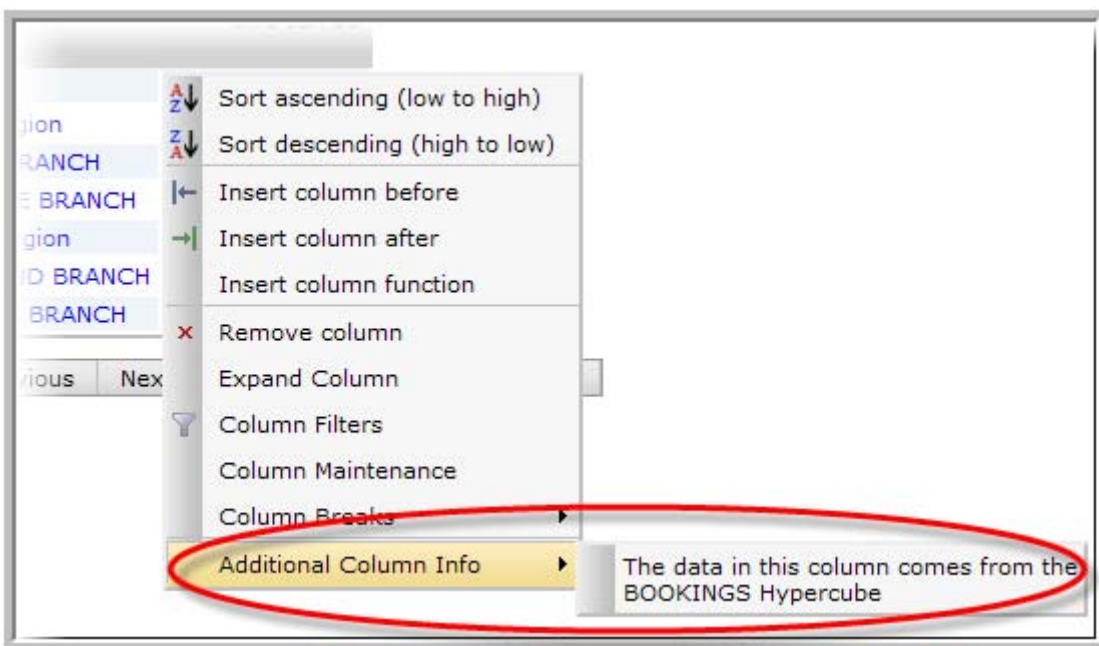
1. Fill out Column 1 with all of the identifier abbreviations from Hypercube1. This information can be found in the MitsMaker Data Elements screen. (It can also be found in the IDENTIS item of the MITS.CONFIG_<appName> file, located within the MITS Discover Operational account.)
2. Next, use Column 2 to write in the name of the identifier that corresponds to the abbreviation from Column 1.
3. Put a line through the boxes in Column 3 for the identifiers you will not be using, then fill out the remaining boxes with the appropriate identifier abbreviations from Hypercube2.

Repeat the steps, reversing the roles of Hypercube1 and Hypercube2 to verify that you have all of the matching identifiers between the two Hypercubes, then use these charts to create the linking items in MITS.APPS.IDENTIS.XREF.

NOTE: A printable example of chart you can use for matching identifier abbreviations is provided at the end of this document.

Provide additional column information, when possible

Use the "Additional Column Information" feature when creating column templates (or individual columns) to provide clarification and further explanation about how the column was generated and why. This information is made available to anybody who views the column by looking in the column menu.



Perform a Create-app and a build wrap-up after creating a new column template

You will need to perform a create-app and a build wrap-up to generate the columns from a newly-created column template before the columns will be made available for use in MITS Discover.

Be aware of the differences between Hypercube drill-down values

Even when Hypercubes have the same drill-down "paths", the drill-down "results", or "values", may differ. At this time, only the comparable values are linked. For example, a Sales Hypercube has a drill-down path that ends with Customer and returns six customer IDs (Customers A, C, D, E, F, and G).

Sales			****
SALES	Customer		SALES YEAR 2008 thru Oct 22
SALES:			507,772
Customer A	+		65,423
Customer C	+		52,835
Customer D	+		65,256
Customer E	+		114,842
Customer F	+		135,944
Customer G	+		73,472
Grand total	=		507,772

[Return to top of Page](#)

An Open Orders Hypercube has a similar drill-down path that also ends with Customer, but only returns five customer IDs (Customers A, B, C, D, and E).

Open Orders			****
OPEN.ORDERS	Customer		SALES YEAR 2008 thru Oct 22
OPEN.ORDERS:			493,288
Customer A	+		70,648
Customer B	+		97,535
Customer C	+		65,237
Customer D	+		71,955
Customer E	+		187,913
Grand total	=		493,288

[Return to top of Page](#)

If a column is added to the first flash screen that reads its values from the Open Orders Hypercube, the added column will only display information for the four customers shared between the two Hypercubes. Additional values (such as those from Customer B, which only appears in the Open Orders Hypercube) are not appended to the drill-down list.

Sales			
SALES Customer		**** SALES YEAR 2008 thru Oct 22	**** SALES FROM OPEN ORDERS YEAR 2008 thru Oct 22
SALES:		507,772	493,288
Customer A	+	65,423	70,648
Customer C	+	52,835	65,237
Customer D	+	65,256	71,955
Customer E	+	114,842	187,913
Customer F	+	135,944	0
Customer G	+	73,472	0
Grand total	=	507,772	395,753

[⏪](#)
[⏩](#)
[Return to top of Page](#)

Displaying an External Column

External columns are added to a flash screen in the same manner as regular columns: through the modify menu or the column menu. If best practices are followed, it should be easy to tell which columns come from external Hypercubes by their names.

Sales YTD

SALES	SALES
Region	YEAR 2008
Warehouse	thru Jun 06
SALES:	1,449,638
Eastern Region	520,936
BOISE BRANCH	154,800
SPOKANE BRANCH	
Western Region	
PORTLAND BRANCH	
SEATTLE BRANCH	

Available Columns

- Add
- Bookings by month
- Bookings by quarter
- Bookings by year
- Booked Orders from BOOKINGS (year-to-date YEAR 2008 thru Jun 06)
- Booked Orders from BOOKINGS (back 1 year YEAR 2007)
- Booked Order years YEAR 2006

Sales YTD w/ Bookings

SALES	SALES	Booked Orders
Region	YEAR 2008	from BOOKINGS
Warehouse	thru Jun 06	YEAR 2008
		thru Jun 06
SALES:	1,449,638	234,517
Eastern Region	520,936	100,859
BOISE BRANCH	154,800	14,635
SPOKANE BRANCH	366,136	86,224
Western Region	928,701	133,658
PORTLAND BRANCH	485,069	118,850
SEATTLE BRANCH	443,633	14,808

