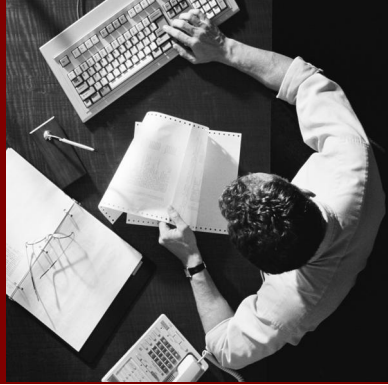


PUBLIC



SAP Library

Data Manager Guide

SAP Business Planning and Consolidation 7.0 SP03

version for the Microsoft platform

Target Audience

- System Administrators
- Technology Consultants

Document version: 2.0 – January 30, 2009



© Copyright 2009 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, Informix, i5/OS, POWER, POWER5, OpenPower and PowerPC are trademarks or registered trademarks of IBM Corporation.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.
Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP AG
Dietmar-Hopp-Allee 16
69190 Walldorf
Germany
T +49/18 05/34 34 24
F +49/18 05/34 34 20
www.sap.com

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

SAP Library document classification: PUBLIC

Disclaimer

Some components of this product are based on Java™. Any code change in these components may cause unpredictable and severe malfunctions and is therefore expressly prohibited, as is any decompilation of these components.

Any Java™ Source Code delivered with this product is only to be used by SAP's Support Services and may not be modified or altered in any way.

Documentation in the SAP Service Marketplace

Documentation is available on SAP Service Marketplace at

<http://service.sap.com/instguidescpm-bpc>

Table of Contents

Welcome to Data Manager	5
About Data Manager.....	5
Understanding the Data Manager menu	6
Understanding the Data Manager toolbar	7
Starting Data Manager.....	8
Clearing saved prompt values	9
Using the member selector	9
Manage Data Options	10
Data Manager packages	11
About packages	11
Standard data management packages	11
Administrative data management packages	12
Using sample DTS packages	13
Running BPC logic with packages.....	15
Organizing packages	16
Assigning packages to teams	17
Modifying packages	17
Testing packages	18
Running and scheduling Data Manager packages	19
Viewing package schedule status	20
Scheduling dimension member processing.....	21
Custom DTS tasks	22
About BPC custom tasks.....	22
Using custom tasks in custom packages.....	23
DTS Terminology	23
Locating Data Transformation Services	23
Using the EvDTSDumpLoad task.....	23
Using the Ev4DTSTaskAdmin task	29
Ev4DTSSendMail.....	32
Ev4DTSTrigger	34
Ev4DTSAvailable.....	35
EvDTSFTP.....	36
K2DTSRunLogic.....	36
EvDTSMModifyPkg	39
Registering custom tasks.....	45
Defining data transformations	47



Using transformation files.....	47
Using conversion files	56
Data transfer and preview.....	57
Transferring Data Files.....	57
Previewing data	58
Index	58

Welcome to Data Manager

Data Manager is a Business Planning and Consolidation (BPC) module that helps you manage the data moving into and out of BPC, as well as copy or move data within and across applications. In addition to allowing you to move and copy data, Data Manager supports mapping and complex transformations of data.

About Data Manager

Data Manager (DM) is based on Microsoft Data Transformation Services (DTS) and SQL Server Integration Services (SSIS). You can use DTS to manage your SQL Server 2000 packages, and SSIS to manage your SQL Server 2005 packages. A 'package' is a set of specific tasks. These tasks are based on information you provide either by entering information in Data Manager package prompts or by modifying the package directly through SQL Server.

You can run DTS for SQL Server 2000 packages (.DTS files) and SQL Server 2005 (.DTSX files). However, if you need to edit packages, you must use SSIS. Please contact your SAP representative for more information about editing existing files in SSIS.

Note: Data Manager is supported for Excel 2000 and later BPC clients.

BPC is delivered with a set of standard packages to meet your many data management needs. See Data manager packages.

Transforming data

Data Manager performs data transformations and/or mapping using transformation and conversion files. For example, when performing a data import, the source data may not be in the correct format. The dimension names may not match the dimension names in BPC; in this case, you would map, in the conversion file, the external dimensions to the internal dimensions. When the DM import package is run it reads the transformation file (which has a pointer to a conversion file) and correctly maps the data. This is just one example of what you can do with data transformations. See Defining data transformations.

Modifying packages

You might have to modify DM packages to work within your company's requirements. The packages that come with BPC are designed to be dynamic so that you do not need to modify the packages in order for them to work with your applications and dimensions. But you might want to modify packages to behave differently or to perform additional tasks. You can make modifications to the packages supplied with BPC or use those packages as templates to build packages as per your requirements. See Modifying packages.

Monitor package status

The Data Manager Status task lets you view the status of packages that are currently running and packages that have been completed. See Reviewing package status.

Understanding the Data Manager menu

The Data Manager menu is named eData, and displays when you open BPC for Excel.



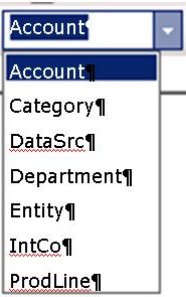
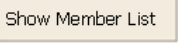
Menu item	Description
Run Package	Opens the Run Package dialog box, where you can run a package for a site and application. See Running and scheduling Data Manager packages
Package Schedule Status	Opens the Package schedule status dialog box, where you can view and edit scheduled packages. See Viewing package schedule status
View Status	Opens the View Status dialog box, where you can see the status of currently running or previously run packages. See Reviewing package status
Organize Package List	Opens the Organize Package List dialog box, where you can add packages and assign them to sites. See Organizing packages
Manage Site User Package Access	Allows you to assign packages to be available to site administrators. See Assigning packages to sites
New Transformation File	Creates a new transformation file Excel workbook. See Transformation files
Manage Transformation File	Sub menus allow you to edit, save, copy, and validate transformation files. See Transformation files
Validate and Process Transformation File	Checks the structure and syntax of transformation files and reports any potential problems. See Validating transformation files
Copy Transformation File	Creates new transformation files based on existing ones that you can then modify for your needs. See Copying transformation files
New Conversion File	Creates new conversion files. See Conversion files
Manage Conversion File	Allows you to edit, save, copy, and validate conversion files. See Conversion files
Validate and Process Conversion File	Checks the structure and syntax of conversion files and reports any potential problems. See Validating conversion files
Copy Conversion File	Creates new conversion files based on existing ones that you can then modify for your needs. See Copying conversion files
Add Conversion Sheet	Adds a sheet to an existing conversion file. You can have multiple sheets in one conversion file in order to use different

Menu item	Description
	<p>sheets depending on the context of the transformation.</p> <p>See Creating and using conversion file worksheets</p>
Clear Saved Prompt Values	<p>Clears saved values for all Data Manager dialog boxes. Values include references to folders and lists of members.</p> <p>See Clearing saved prompt values</p>
Data Manager Help	Opens this help.
About Data Manager	Displays version information about Data Manager.

Understanding the Data Manager toolbar

The following table lists the icons on the Data Manager toolbar and their function.

Icon	Task	Description
	Run Package	<p>Opens the Run Package dialog box, where you can run a package for a site and application.</p> <p>See Running and scheduling Data Manager packages</p>
	View Status	<p>Opens the View Status dialog box, where you can see the status of currently running or previously run packages.</p> <p>See Reviewing package status</p>
	New Transformation File	<p>Creates a new Transformation File Excel workbook.</p> <p>See Transformation files</p>
	Manage Transformation File	<p>Sub menus allow you to edit, save, copy, and validate transformation files.</p> <p>See Transformation files</p>
	Validate and Process Transformation File	<p>Checks the structure and syntax of transformation files and reports any potential problems.</p> <p>See Validating transformation files</p>
	Copy Transformation File	<p>Creates new transformation files based on existing ones that you can then modify for your needs.</p> <p>See Copying transformation files</p>
	New Conversion File	<p>Creates new conversion files.</p> <p>See Conversion files</p>
	Manage Conversion File	<p>Sub menus allow you to edit, save, copy, and validate conversion files.</p> <p>See Conversion files</p>
	Validate and Process Conversion File	<p>Checks the structure and syntax of conversion files and reports any potential problems.</p> <p>See Validating conversion files</p>

Icon	Task	Description
	Copy Conversion File	Creates new conversion files based on existing one that you can then modify for your needs. See Copying conversion files
	Add Conversion Sheet	Adds a sheet to an existing conversion file. You can have multiple sheets in one conversion file in order to use different sheets depending on the context of the transformation. See Creating and using conversion file worksheets
	< Selected Dimension >	Works with the Show Member List button (see below) to show the Data Manager member selector for the selected dimension.
	Show Member List push button	Clicking this button shows the Data Manager member selector. See Using the member selector

Starting Data Manager

You start Data Manager from BPC for Excel.

To start Data Manager

1. Click the BPC icon on your desktop.
2. From the BPC launch page, select BPC for Excel.
3. From the Getting Started - BPC for Excel action pane, select Manage Data.

Clearing saved prompt values

Data Manager saves values you type into various dialog boxes so that you do not have to repeat the information each time. You can clear these saved prompt values so that the system starts over with no values automatically filled in inside dialog boxes.

To clear saved prompt values

1. Start BPC for Excel, and select eData > Clear Saved Prompt Values.
2. At the dialog box, select Yes to clear the saved prompt values, or No to leave the saved prompt values in place.

Using the member selector

The Data Manager member selector gives you a hierarchical view of a dimension's members so you can choose one or more of them, depending on the context.

You can use the member selector anywhere you see a choose members button in the Data Manager interface, such as when you run a package. You can also use the member selector when you select a dimension and click the Show Member List button on the Data Manager toolbar. See Understanding the Data Manager toolbar.

The Member Selector dialog box that displays after clicking the choose members button allows you to browse and select members. There are two general steps to using the member selector:

- Set up dimension member selection
- Select dimension members

Tip: Additionally, you can save and load filters that you set up. See the Saving and Loading filters section, below.

The following procedure steps through the entire member selection process.

To select dimension members

1. Start BPC for Excel, and select Manage Data from the Getting Started - BPC for Excel action pane.
2. From the Manage Data Options action pane, select Run a data management package, double-click a package to open it, and then click on the choose members button next to the dimension whose members you want to select.
3. You can narrow the list of members on the left by choose to filter the list based on properties. If you would like to filter the member list, do the following:
 1. Select the Use Filter check box.
 2. Choose a value for each property that you want to filter by clicking a down arrow and selecting from the list.
 3. Click the Refresh button to filter the member list on the left.
4. You can select one or more members from the list on the left.
 - a. To select one member, click that member. Or, you can view the properties of the selected member by right-clicking the member and choosing Properties from the menu or by clicking the Properties button at the top of the screen.
 - b. To select more than one member, you can use CTRL-click to select non-contiguous members, or you can use SHIFT-click to select a range of members.
5. You can show member descriptions in the member list by selecting the View member with description checkbox.
6. You can copy the selected members. You might want to do this to paste the list into an Excel spreadsheet. To copy selected members, right-click one of the selected members, and then choose Copy from the menu.
7. When you are satisfied with your selection, click the Take selected button to return the selected members to the package dialog box.

Note: You can also choose to ignore the members you have selected by clicking the Take all in list button. This option returns all the members in the list to the package. Note that if you used a filter to create the list, only the members that match the filter are returned.

Saving and loading filters

When you create a filter by filling out the filter section of the member selector dialog box, you can save it and then reuse it later.

You can use the Manage Filter button to save the filter, and then you can choose that filter from the User filter list to load the filter.

To save and load member selector filters

1. Access the member selector. For example, run a package and then click the choose members button.
2. Choose the filter criteria from the filter lists.
3. Click the Manage Filter button, type the name of the filter, and then click the Save button.
4. To load the filter at a later time, choose it from the User Filter list at the top.
5. To delete a filter, choose it from the User Filter list, click the Manager Filter button, and then click the Remove button.

Manage Data Options

The following table describes the items on the Manage Data Options action pane.

Item	Description
Upload data file	See Transferring data files
Download data file	See Transferring data files
Preview data file	See Previewing data
Run a data management package	See Running and scheduling Data Manager packages
View package status	See Viewing package schedule status
View schedule status	See Viewing package schedule status
Maintain conversions	See Using conversions
Maintain transformation	See Using transformation files
Maintain data management	See Data Management Maintenance Options

Data Manager packages

Packages allow you to move data into, out of, and between your BPC databases. BPC works with SQL's Data Transformation Services (DTS) to allow you to set up and run DTS packages to work with your data.

You should be familiar with SQL DTS if you want to create your own packages. You can perform transformations of your data when you import or export data to and from your BPC database.

About packages

Packages allow you to move data into, out of, and between your BPC databases. There are two broad types of packages you can use to move data depending on your needs:

- Standard Packages

Standard packages perform their function while the system is live - meaning that users can be online. For example, you can use a standard package to load monthly data.

We recommend that you use standard packages whenever possible, because they work within the BPC user ID security framework and there is little risk of data integrity issues.

Administrators with the appropriate task security can run standard packages.

See [Standard data management packages](#)

- Administrative Packages

Administrative packages are designed to perform operational tasks on the database. Examples include back-end data manipulation where you are loading bulk data for the first time, exporting an entire database, and running default logic formulas on your data.

Administrative packages take the system offline and can only be run by administrators with the appropriate task security.

See [Administrative data management packages](#)

You can also assign transformation files to packages to format your inbound and outbound data. See [Defining data transformations](#).

The Data Management, Data Modeling, and Financial Processes groups all contain Standard packages. The System Administration group contains Administrative packages. See [Standard data management packages](#) and [Administrative data management packages](#).

To see a list of your Data Manager packages, select **Manage Data > Run Package** or **Manage Data > Organize Package List** from the **Getting Started - BPC Excel** action pane.

Standard data management packages

The standard packages that are provided by default with the sample application set, Apshell are divided in two groups:

- Data Management
- Financial Processes

Note: For information on the System Administration package group, see [Administrative data management packages](#). For information on adding additional packages, see [Using sample DTS packages](#).

Data Management

You can use the packages provided in the Data Management folder to perform general data management tasks, such as importing and exporting data. If a file has an associated transformation file, you can assign the file to the package as is, or modify it to suit your needs. See [Transformation files](#).

Package name	Description
--------------	-------------

Package name	Description
Clear	<p>This package clears the data in the Fact and FAC2 table by importing zero values. The optimize process removes the zero values, which clears all values from the table. Running this package also clears comments from the comments table.</p> <p>You can use the import.xls transformation file with this package.</p> <p>You must select at least one dimension in order to run this package. If you select [All], the package will not run.</p>
Copy	<p>This package copies data between dimensions within BPC application.</p> <p>See Running a Copy package</p>
Export	<p>This package exports data values from the BPC cube (including calculated values) into an ASCII file. You are prompted to enter a name for the export file and for a data selection to export.</p> <p>You can use the export.xls transformation file with this package.</p>
Import	<p>This package imports one or more ASCII files into the current application. You are prompted for one or more import file and associated conversion files. The signs of the amounts are reversed based on the Account type.</p> <p>You can use the import.xls transformation file with this package.</p>
Move	<p>This package lets you move any selection of data within an application. You are prompted for source and destination data selections for all dimensions. The destination selection's fact table records are cleared before the source's records are moved. The source's fact table records are cleared after they are moved to the destination.</p>
Financial processes	
Package	Description
FX Restatement	<p>This package is used for currency translation. See Admin Help for more information on setting up currency translation.</p>
IC Eliminations	<p>This package is used to Perform Inter-Company eliminations. Please see Administrator help for more information on Inter-Company eliminations.</p>

Administrative data management packages

The administrative data management packages provided with BPC allow you to import and export data in bulk, copy data, and perform other administrative tasks. If a package has an associated transformation file, you can assign the file to the package as is, or modify it to suit your needs. See Transformation files.

Package Name	Description
Append into Fact Table	<p>This package imports data into the database by appending the data to the fact table, processes the data into the BPC cube, and then runs default logic against the imported data set.</p> <p>You can use the import.xls transformation file with this package.</p>
Import into Fact Table with Clear	<p>This package clears the fact table for all Entity/Category/Time member combinations found in the data source file and imports the data into the database. Optionally, it also processes the data into the BPC cube, and/or runs default logic against the imported data set.</p> <p>You can use the import.xls transformation file with this package.</p>

Package Name	Description
Export from Fact Table	This package exports input-level data (but includes translated currency values) from the application fact table into an ASCII file. You are prompted to enter a name for the export file, a selection of data to export, and a conversion file to use. You can use the export.xls transformation file with this package.
Default Formulas	This package executes default formulas stored in your default.xls file.

Using sample DTS packages

In addition to the standard and administrative packages provided with Data Manager, BPC provides several additional sample DTS packages that you can add to Data Manager. That is, they are not part of the default configuration, but you can add them to Data Manager using the Organize Packages task. See [Organizing packages](#).

You can use many of the sample packages as provided, or you can modify them to suit your needs. The following tables describe the sample packages. They can be found on the BPC File server in the DataManager\PackageFiles\Examples folder of the associated application set and application.

Samples requiring modification

The following table describes the sample DTS packages that require initial modification. See [Modifying packages](#).

Package name	Description
ImportUsingFTP	This package allows you to import an ASCII file to download from an FTP site into the current application. You can use SQL Enterprise Manager to modify the server name within the package. After you modify the server name, you can run this sample package at any time.
LoadRootCause	This package loads values from a text file into the RootCauseEvent SQL table. Before running the package, you specify the name of the text file that contains the root cause data, the FTP location (if applicable), and some connection properties.

Samples that do not require modification

The following table describes the BPC sample packages that you can use as provided, or modify, as desired. See [Modifying packages](#).

Package name	Description
AdminTask_MakeDim	Creates dimension members using a SQL table and schedules the processing of dimension members. For example, if a new member ID is added, this process updates the fac2 table with the updated list. See Scheduling dimension member processing
AdminTask_Optimize	Optimizes applications with lite, incremental, or full option.
AdminTask_Process	Processes the application with incremental or full option.
AdminTask_Security	Allows you to run and/or schedule application set security.
AdminTask_Validate	Validates the logic file.
Append	This package appends an ASCII file into the current application.

Package name	Description
AuditBackup	Backs up audit data.
AuditClear	Clears audit data.
AuditPurge	Purges audit data.
AuditRestore	Restores audit data.
ClearComments	Clears comments from the comments table.
Clear from Fact Table	Removes the data from the Fact and FAC2 tables. The optimize process resynchronizes the Fact and FAC2 tables with the cube. Running this package also clears comments from the comments table.
Clear the Journals Tables	Clears Journal tables and creates an output file.
Copy Application	Copies the application database.
Copy from Fact Table	Copies the fact table.
Export the Journal tables	Export Journals Table to an output file.
Import Access into Fact Table	Imports the MS Access database into the fact table. You can use the import.xls transformation file with this package.
Import Access	Imports the MS Access database into the fac2 table. You can use the import.xls transformation file with this package.
Import Category by Time into Fact Table	Imports Category by Time data into the fact table. You can use the Category_By_Time.xls transformation file with this package.
Import Category by Time	Imports Category by Time data into the fac2 table. You can use the Category_By_Time.xls transformation file with this package.
Import Dimension	Available for backward compatibility. See AdminTask_MakeDim.
Import Excel into Fact Table	Imports contents of an Excel file into the fact table. You can use the import.xls transformation file with this package.
Import Excel	Imports contents of an Excel file into the fac2 table.
Import SQL into Fact Table	Imports SQL database into the fact table.
Import SQL	Imports a SQL database into the fac2 table. You can use the import.xls transformation file with this package.
ImportAndSendmail	Import an ASCII file and send the result to specific user.
Move from Fact Table	Moves data from the fact table.

Package name	Description
Restore the Journals Tables	Restores Journals tables from an output file.
Scenario Modeling	Allows you to use scenario modeling for forecasting purposes. This package calls a logic file, which is designed for your specific needs.

To add sample DTS packages to Data Manager

1. From Data Manager, select eData > Organize Package List.
2. Click the Add Package icon.
3. Complete the Add Package dialog box, then click Add. See Organizing packages.
4. Click Save, then click OK.

Running BPC logic with packages

You can set up BPC logic to work together with your Data Manager packages. Most packages automatically run default logic (default.xls) for your application when importing data. You can also specify to run a different set of logic or no logic, as required.

You can also set up a step inside a package that runs logic. The K2DTSRunlogic custom DTS task allows you to set up parameters and run any logic file or logic subroutines as needed. See K2DTSRunlogic.

Organizing packages

You can organize the packages that are available in Data Manager. Organizing packages involves adding packages to Data Manager, editing, moving, copying, and removing packages in Data Manager. The organize package tasks are grouped into a dialog box called Organizing Packages. You can perform the following package organization tasks:

- Add packages to Data Manager or edit Data Manager packages. See Adding or editing packages
- Move or copy packages. See Moving or copying packages
- Remove packages. See Removing packages

Adding or editing packages

BPC provides several sample packages that you can add to Data Manager. You can also edit any of the packages provided in Data Manager.

The following table describes the fields you must complete when adding or editing packages. If any of these fields are incorrect or left blank, the package generates an error when the package is executed.

Field	Description
PackageType	SQL, FILE or Repository. See SQL server books online for more information.
File	If the FILE package type is selected, a File text box is displayed so that you can specify the file that contains the package. You can browse for files using the browse button.
PackageName	The name of the package. Use the View Package List button to see a list of available packages. If you select the File package type, then you must select a file containing packages in order to use the View Package List button.
Group	The folder group to which the package will be added. You can define a new group by typing a new group name here.
Description	A description of the package.

Field	Description
User Group	You can control who has access to run the package: A user or an administrator.

To add or edit packages

1. Start BPC for Excel, and select Manage Data from the Getting Started - BPC for Excel action pane.
2. From the Manage Data Options action pane, select Organize Package List. The Package Maintenance tasks screen is displayed.
3. Click the Add or Edit icon.
4. Complete the fields from the table above. If you want to create a new group, type the new group's name in the Group field.
5. When adding a new package, if you click the browse icon next to the File text box, the Add File dialog box is displayed.

Note: You can find example packages in the Examples folder. See Using sample DTS packages. You can upload package files to the MyFiles or Examples group using the eData > Data Upload task. See Transferring Data Files for more information. The System Files group is read-only and contains packages that are maintained by SAP. You cannot upload files to this group.

6. If adding a package, click the Add button, or, if editing a package, click the Save button.
7. Click the package to select it, and then click the Save Package List pushbutton.

Note: You must save the package before you can run it. Adding the package only adds it to the folder list. Saving it makes it available for use.

Moving or copying packages

You can move or copy packages between sites and groups. You might want to do this so that site administrators can work with a modified version of a company package.

To move or copy packages

1. Start BPC for Excel, and select Manage Data from the Getting Started - BPC for Excel action pane.
2. From the Manage Data Options action pane, select Organize Package List.
3. Click the Move or Copy icon.
4. Select the Destination Site from the drop down list.
5. Select the Destination Group from the drop down list.
6. Click the Move or Copy button.
7. Click the package to select it, and then click the Save Package List button.

Removing packages

You can remove packages if you no longer want them available to be run through Data Manager.

To remove a package from the list

1. Start BPC for Excel, and select Manage Data from the Getting Started - BPC for Excel action pane.
2. Select Organize Package List.
3. Highlight the package you want to remove.
4. Click the Remove icon.
5. Click the Save Package List button.

Assigning packages to teams

You can assign packages to be available to BPC teams. You use the Manage User Package Access task to do this.

To assign packages to teams

1. Start BPC for Excel, and select eData > Manage Team User Package Access.
2. Chose a team from the icons on the left.
3. From the package list, select the check box next to each package that you want to make available to the users in the selected team.
4. If you want to give all teams access to the packages, select the All Packages check box at the top of the screen.

Modifying packages

You can modify any of the provided packages to suit your business needs. BPC provides a mechanism for doing this with the EvDTSMModifyPgk custom task. The EvDTSMModifyPgk task allows you to set up pre-defined answers to package prompts so that you can automate running packages. See EvDTSMModifyPgk.

You can also modify packages through the SQL Enterprise Manager interface if you have the appropriate security privileges.

To modify a sample package

1. From the applicable server, start SQL Enterprise Manager.
2. Expand the nodes until you get to Data Transformation Services folder, then right-click it and select Open Package. See Locating Data Transformation Services.
3. Browse to the DataManager\PackageFiles\Examples folder of the applicable application set and application.
4. Select the sample package you want to modify, then from the Select Package dialog box, select the package again, and click OK.
5. Double-click on a task to view its properties, and modify as desired. See About custom DTS tasks.

Testing packages

We recommend that you test all packages before making them available to users. You use the Debug script in the Data Manager (DM) task EvDTSMModifyPgk for testing packages. After successfully running the package in the Debug script you copy it into the Dynamic script for use in BPC.

To test a package

1. In SQL Server Enterprise Manager, expand the BPC server then expand the Data Transformation Services folder.
2. Select Local Packages.
3. In the Local Packages pane, double-click the package you want to modify.
4. Double-click EvDTSMModifyPgk (Assign Parameters in BPC sample packages) task.
5. With the Dynamic script button selected, copy the script by right clicking, then selecting Select All, then right-clicking again and selecting Copy within the script body.
6. Select Debug script button and paste the script into the Debug script by right-clicking and selecting Paste.
7. Make any desired modifications to the script as necessary. See EvDTSMModifyPgk.
8. When you are ready to test the script click Ok.
9. Select Package > Execute.

10. If the package is successful and accomplishes the desired outcome, copy the modified script back into the Dynamic script. If the package was not successful you can use the Debug command (see below) to troubleshoot the script.
11. Select Package > Save.

The Debug command

When inserted in a script the debug command creates a log file (EvDTSMModifyPkg.log) which lists the all the properties of a package along with their current values. Using this file you can check to see that all properties are being assigned the correct values.

Debug command syntax:

```
Debug(value)
```

where *value* can be one of the following:

```
On – Turn on the Debug feature
```

```
APPEND – Turn on the debug feature. Debug messages are appended to the  
debug file.
```

The debug file, named EvDTSMModifyPkg.log, is created in the tempfiles directory of the current BPC application.

To use the debug command

1. Insert the debug command in the Debug script of the EvDTSMModifyPkg task.
2. Execute (Run) the package.
3. Review the EvDTSMModifyPkg.log file for errors.

Running and scheduling Data Manager packages

You can run and/or schedule to run a Data Manager (DM) package. Access to DM packages is controlled by task security set by an administrator.

The Run Package dialog box shows the contents of the Data Management folder. You can also see a customized view of the list in the Selected List dialog box. See Creating and using selected package lists, below.

To run and/or schedule packages

1. Start BPC for Excel, and select Manage Data from the Getting Started - BPC for Excel action pane.
2. From the Manage Data Options action pane, select Run a data management package.
3. From the Run Package dialog box, select a team from the left, if desired, select a folder under the Package Group folders, then select a package from the list on the right.
4. Click the Run button at the bottom of the dialog box.
5. The screens that follow differ depending on the package chosen. Each package allows you to set options for running the package, scheduling the package, or both. Follow the instructions on the screen.
6. After supplying the required information the package is sent to the server and executed. You can view the status of the package. See Reviewing package status.

Creating and using selected package lists

Administrators have access to all packages for the AppSet. You may want to customize the list of packages you run often so you do not have to find the packages each time. You do this using the Selected List button on the Run Package dialog box.

You can also use the Selected List dialog box for the following: Use the data file transfer button to upload packages to the server (see Transferring data files) or use the view package status button to view package status (see Reviewing package status).

To create a selected package list

1. Start BPC for Excel, and select Manage Data from the Getting Started - BPC for Excel action pane.
2. From the Manage Data Options action pane, select Run a data management package, then click the Selected List button.
3. Click the Organize this screen button at the bottom of the dialog box. The Organize Package List screen is displayed.
4. Move the packages you want to show in the list to the Selected List area.
5. If you want your short list to be displayed when you choose Run a data management package, select the show the selected list screen check box, then click OK.
6. To run a package, click a package and then click the Run button.
7. If you only wanted to set up the list, click the Cancel button.

Running a Copy package

This package copies data within BPC application. You are prompted for source and destination data selections for all dimensions in the application. The destination data selection's fact table records are cleared before the source records are copied.

If the source and destination packages have different dimensionality, you must map the dimensions using a transformation file. See Mapping dimensions between applications.

To run a Copy package

1. Start BPC for Excel, and select Manage Data from the Getting Started - BPC for Excel action pane.
2. From the Manage Data Options action pane, select Run a data management package.
3. Select Copy, then Run.
4. Select one of the following:
 - Merge to copy all specified records, leaving all remaining records in the destination intact (does not add to existing data)
 - Replace & clear to clear data in the destination area using the selected dimension list (by default, each entity/category/time combination), then copy the data from the source to the destination.

For example, these are the database records before an import:

Category	Entity	Time	Account	Amount
Actual	US	2008.NOV	Sales	1000
Actual	EMEA	2008.NOV	Sales	1000
Actual	US	2008.NOV	Cash	200
Actual	EMEA	2008.NOV	Cash	200

Import file records:

Actual	US	2008.NOV	Sales	2000
Actual	EMEA	2008.NOV	Cash	400

Results of import with clear:

Actual	US	2008.NOV	Sales	2000
Actual	EMEA	2008.NOV	Cash	400

Results of import with merge:

Actual	US	2008.NOV	Sales	2000
Actual	EMEA	2008.NOV	Sales	1000
Actual	US	2008.NOV	Cash	200
Actual	EMEA	2008.NOV	Cash	400

5. Select whether to run default logic for stored values after importing. If you select No, the package does a straight copy of the data. If you select Yes, the copy package copies the data, then runs the default advanced formulas that apply to that data.

6. Select whether to check work status settings when importing data. If you select Yes, work status settings will be copied with the data. If you select No, work status settings are not copied with the data.

Viewing package schedule status

When you run packages, you have the option of scheduling them to run at predetermined times. The schedule status is different from the package status because package status shows you the results of running packages, while the schedule status shows you when various packages are scheduled to run.

You can check and change package schedules for all scheduled packages using the Package Scheduler Status window.

To view package schedule status

1. Select eData > Package Schedule Status.
2. View the list of scheduled packages.
3. To edit a schedule, select the package, then click the Edit schedule button.
4. To view a history of when the package ran against the schedule, click the View history button.
5. To edit the parameters for the package, click the Edit parameters button.
6. To delete the schedule, click the Delete button.

Scheduling dimension member processing

Before scheduling the processing of dimension members, you must first use BPC Administration to validate a dimension. Then use Data Manager to schedule the processing of the members of that dimension.

To schedule the processing of dimension members

1. Open Data Manager and select eData > Organize Package List. The Organize package list dialog is displayed.
2. From the toolbar, click the Add Package icon. The Add Package dialog is displayed.
3. Select File in the Package Type field. Click the lookup button beside the File field to display the list of packages.
4. Select the Examples folder and select AdminTask_MakeDim.dts.

Note: If you have an existing application set, copy this package from the most recent version of ApShell to get the appropriate updates.

5. Click Open. You are returned to the Add Package dialog. Click Add. You are returned to the Organize package list dialog. Click Save.
6. Select eData > Run Package. Select AdminTask_MakeDim.dts from the package list and click Run. The Run Package dialog is displayed.
7. Be sure the desired dimension name in the Input Dimension name field. (Leave the Input Source table name field blank.) Click Next.
8. Select Yes in the Schedule the processing of dimension members field. Click Finish.
9. Complete the desired schedule information and click OK.
10. Use the custom DTS task EvDTSMModifyPkg to schedule the dimension member processing as follows:

```
TASK(Admin_Makedim,BSCHEDULEPROCESS,%ScheduleProcess%)
```

...where %ScheduleProcess% is 1 (Yes). See EvDTSMModifyPkg

Custom DTS tasks

BPC provides custom DTS tasks for you to use, or you can create your own. These custom tasks allow you to manage your data and perform unattended Administration tasks. You use the tasks by registering the task, then adding them to your packages.

About BPC custom tasks

The following table describes the provided BPC custom DTS tasks:

DTS Task	Description
EvDTSDumpLoad	The EvDTSDumpLoad task has many useful data dump and data retrieval functions. See Using the EvDTSDumpLoad task
Ev4DTSAdmin	Enables you to run one of a number of eAdmin tasks through DTS. See Using the Ev4DTSAdmin task
Ev4DTSSendMail	Sends an email message before or after another task has processed. See Ev4DTSSendMail
Ev4DTSTrigger	The trigger task allows you to wait for a trigger condition before processing the next task. See Ev4DTSTrigger
Ev4DTSAvailable	This task allows you to control whether the system is set to available (online) or not available (not online) status. See Ev4DTSAvailable
EvDTSFTP	Use the FTP task to transfer data files from an FTP site to the BPC server. See EvDTSFTP
K2DTSRunlogic	K2DTSRunlogic is a DTS version of the BPC Logic module that can be used from within the Data Manager module. This allows you to use the full power of the Logic module in a batch setting. See K2DTSRunlogic
EvDTSMModifyPkg	EvDTSMModifyPkg is a task that can be inserted into any Data Manager package to modify the package itself, when executed. When this task runs, it modifies the properties of objects contained in the package (like tasks and connections), according to some user-defined instructions specified in the task itself. See EvDTSMModifyPkg
EvDTSConvert	This task takes an input data and transformation file (which may or may not use conversion files) to create an output file. For example, an input file could be a file exported from a similar product, and takes the transformation and conversion files to convert to the BPC format.
K2DTSMakeDim	Creates dimension members using a SQL table and schedules the processing of dimension members. For example, if a new member ID is added, this process updates the fac2 table with the updated list.

Using custom tasks in custom packages

Custom tasks allow you to manage your data and perform unattended Administration tasks. You use the tasks by registering the task, then adding them to your packages.

To use custom tasks in custom packages

1. From SQL Server Enterprise Manager, open the Data Transformation Services folder.
2. Create a new package, or select an existing package to modify.
3. Select Task > Register Custom Task.
4. Click the Browse button next to the Task Location field to search for the location of the desired dll file. (By default, the .dll files are stored in BPC/Websrvr/bin).
5. Enter a task description, select an appropriate icon, then click OK.
6. Drag the icon to the designer window, modify its properties, as desired, then select Package > Save.

DTS Terminology

The following table lists some common DTS terms and their descriptions. For more complete information, see the DTS Help (DTSSQL.CHM) on your SQL server.

Term	Description
DTS Package	An organized collection of connections, DTS tasks, DTS transformations, and workflow constraints. See About packages
DTS Task	A discrete set of functionality, executed as a single step in a package. Each task defines a work item to be performed as part of the data movement and data transformation process, or as a job to be executed.
DTS Transformations	One or more functions or operations applied against a piece of data before the data arrives at the destination. See Creating and editing transformation files

Locating Data Transformation Services

Data Transformation Services is available from SQL Enterprise manager on the BPC server, not from the Data Manager interface.

To locate Data Transformation Services

1. Open SQL Enterprise Manager on the BPC server.
2. Open the SQL Server Group and Server.
3. Click on the Data Transformation Services folder.
4. Open or create a package to which you want to add BPC tasks.

Using the EvDTSDumpLoad task

The EvDTSDumpLoad task has many useful data dump and data retrieval functions. As stated in the About Data Manager topic, you can export data directly from the SQL fact table or the Analysis Services cube, import ready-to-load ASCII files directly into the fact table, and perform processing tasks.

EvDTSDumpLoad can be added to any DTS package. You can complete the necessary fields directly in the EvDumpLoad Task or by script using EvModifyPkg.

You can perform data management tasks and journal management tasks with EvDTSDumpLoad. Data management tasks involve processing the database, importing or exporting data, and clearing data. Journal management tasks involve exporting, restoring, or clearing journal data.

EvDTSModifyPkg task list

You can use the following task commands in the EvDTSModifyPkg task:

```
TASK(TASKNAME,NAME,DTSTask_EvDTSDumpLoad.clsEvDTSDumpLoad_1)
TASK(TASKNAME,DESCRIPTION,TASKNAME)
TASK(TASKNAME,APPSET,apshell)
TASK(TASKNAME,APP,finance)
TASK(TASKNAME,USER,tonyd)
TASK(TASKNAME,DATATRANSFERMODE,1)
TASK(TASKNAME,PROCESSMODE, 2)
TASK(TASKNAME,PROCESSCUBE,1)
TASK(TASKNAME,PROCESSCUBE_FAC2,)
TASK(TASKNAME,FILE,C:\...\Webfolders\ApShell\Finance\DataManager\DataFiles\Examples\Import.txt)
TASK(TASKNAME,SQL,)
TASK(TASKNAME,MDX,)
TASK(TASKNAME,MAXMEMBERS,)
TASK(TASKNAME,SELECTION,)
TASK(TASKNAME,CLEARDATA,0)
TASK(TASKNAME,TIMEIDFORMAT,0)
TASK(TASKNAME,SELECTIONFROMTASK,"CATEGORY,TIME,ENTITY")
TASK(TASKNAME,DIMLIST,"CATEGORY,TIME,ACCOUNT,ENTITY,DATASRC,INTCO,RPTCURRENCY")
TASK(TASKNAME,RUNTHELOGIC,0)
TASK(TASKNAME,GETMEASURE,)
TASK(TASKNAME,MACHINENAME,)
TASK(TASKNAME,VALIDATETMPFILE,)
TASK(TASKNAME,DMMCOPY,)
```

Data management tasks

EvDTSDumpLoad tasks allow you to perform application set processing, import or export data, and clear data from the database.

Processing the application (using EvDTSDumpload)

You can process the application in conjunction with data import, export, or clear, or on its own. Select the None option in the EvDTSDumpload task to process the application.

To process the application set

1. Open a package or create a new package in Data Transformation Services on the BPC server. See Data Manager packages.
2. If you have not already done so, register the Ev4DTSDumpload task (Ev4DTSDumpLoad.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task, and then select the None option under Data Management.
5. Fill in the application set, application, and the user ID.

Note: The user ID must be BPC administrator ID in order to validate logic files. For more information about validating logic, see Administration Help.

6. Select the Process Cube option, then select one of the following options:
 - Full Process — Rebuilds the entire application (cube). Select this option if the application structure is changed.
 - FAC2 — Process only short-term data storage (the FAC2 table). Data Manager imports data to the short-term data storage by default, so there may be no need to process the entire application.
 - Refresh Data — Clears data from the application and repopulates with existing data.
7. Click OK.

Tip: You can also use TASK commands in the EvDTSMModifyPkg task to automatically complete the information for you.

Importing into SQL

You can import data into SQL using the Import into SQL option.

To import into SQL

1. Open a package or create a new package in Data Transformation Services on the BPC server. See Data Manager packages.
2. If you have not already done so, register the Ev4DTSDumpload task (Ev4DTSDumpLoad.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task, and then select the Import into SQL option under Data Management.
5. Fill in the application set, application, and the user ID.

Note: The user ID must be BPC administrator ID in order to validate logic files. For more information about validating logic, see Administration Help.

6. Type the path and name of a import file, or select one using the browse button.
7. By default, the import process appends the data in the import file to the data already in the database. Select the Clear existing data checkbox to change this behavior.
8. If you selected the Clear existing data checkbox, in the dimension list, select the dimensions on which to perform the data clear.
9. Select the Process Cube option, then select one of the following options:
 - Full Process — Rebuilds the entire application (cube). Select this option if the application structure is changed.
 - FAC2 — Process only short-term data storage (the FAC2 table). Data Manager imports data to the short-term data storage by default, so there may be no need to process the entire application.
 - Refresh Data — Clears data from the application and repopulates with existing data.
10. Click OK.

Tip: You can also use TASK commands in the EvDTSMModifyPkg task to automatically complete the information for you.

Importing into Cube

The Import into Cube option allows you to import an ascii text file that is in the correct format directly into the Analysis Services database.

To import into a Cube

1. Open a package or create a new package in Data Transformation Services on the BPC server. See Data Manager packages.
2. If you have not already done so, register the Ev4DTSDumpload task (Ev4DTSDumpLoad.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task, and then select the Import into cube option under Data Management.
5. Enter the application set, application, and the user ID.

Note: The user ID must be an administrator ID in order to validate logic files. For more information about validating logic, see Administration Help.

6. Type the path and name of a import file, or select one using the browse button.
7. By default, the import process appends the data in the import file to the data already in the database. Select the Clear existing data checkbox to change this behavior.
8. If you selected the Clear existing data check box, in the dimension list, select the dimensions on which to perform the data clear.
9. Select ProcessCube Fac2 if you want to process the short-term data storage database.

10. Select Run default logic if you want the logic contained in default.lgc to be run after importing the data.
11. Click OK.

Tip: You can also use TASK commands in the EvDTSMModifyPkg task to automatically complete the information for you.

Exporting from SQL

The Export from SQL option allows you to export an ASCII text file that contains data from your BPC database.

To export from SQL

1. Open a package or create a new package in Data Transformation Services on the BPC server. See Data Manager packages.
2. If you have not already done so, register the Ev4DTSDumpload task (Ev4DTSDumpLoad.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task, and then select the Export from SQL option under Data Management.
5. Fill in the application set, application, and the user ID.

Note: The user ID must be BPC administrator ID in order to validate logic files. For more information about validating logic, see Logic help.

6. Type the path and name of an export file, or select one using the browse button.
7. Select Export time in timeID format if you want the time members to use the TIMEID rather than time member names.
8. You can also choose to enter a selection file, which contains a list of dimensions and member to export, or a SQL statement that returns the member list needed for export.
9. Click OK.

Tip: You can also use TASK commands in the EvDTSMModifyPkg task to automatically complete the information for you.

Exporting from Cube

The Export from Cube option allows you to export an ascii text file that contains data from your BPC database.

To export from a cube

1. Open a package or create a new package in Data Transformation Services on the BPC server. See Data Manager packages.
2. If you have not already done so, register the Ev4DTSDumpload task (Ev4DTSDumpLoad.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task, and then select the Export from Cube option under Data Management.
5. Fill in the application set, application, and the user ID.

Tip: The user ID must be BPC administrator ID in order to validate logic files. For more information about validating logic, see Admin Help.

6. Type the path and name of an export file, or select one using the browse button.
7. Select Export time in timeID format if you want the time members to use the TIMEID rather than time member names.
8. Select a time measure from the Measure list.
9. You can also choose to enter a selection file, which contains a list of dimensions and member to export, or a SQL statement that returns the member list needed for export.
10. Click OK.

Tip: You can also use TASK commands in the EvDTSMModifyPkg task to automatically complete the information for you.

Clearing data from SQL

Select the Clear data from SQL option to clear data from the SQL database.

To clear data from SQL

1. Open a package or create a new package in Data Transformation Services on the BPC server. See Data Manager packages.
2. If you have not already done so, register the Ev4DTSDumpload task (Ev4DTSDumpLoad.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task, and then select the Clear data from SQL option under Data Management.
5. Fill in the application set, application, and the user ID.

Note: The user ID must be BPC administrator ID in order to validate logic files. For more information about validating logic, see Logic help.

6. Type a member selection in order to limit the clear action to the specified members.
7. Select the Process Cube option, then select one of the following options:
 - Full Process — Rebuilds the entire application (cube). Select this option if the application structure is changed.
 - FAC2 — Process only short-term data storage (the FAC2 table). Data Manager imports data to the short-term data storage by default, so there may be no need to process the entire application.
 - Refresh Data — Clears data from the application and repopulates with existing data.
8. Click OK.

Tip: You can also use TASK commands in the EvDTSMModifyPkg task to automatically complete the information for you.

Journal Management tasks

The EvDTSDumpLoad Journal management tasks allow you to export, restore, or clear journal data.

Exporting journal data

You can export journal data from BPC using the Export Journal task.

To export journal data

1. Open a package or create a new package in Data Transformation Services on the BPC server.
2. If you have not already done so, register the Ev4DTSDumpload task (Ev4DTSDumpLoad.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task, and then select the Export option under Journal Management.
5. Fill in the application set, application, and the user ID.

Note: The user ID must be BPC administrator ID in order to validate logic files. For more information about validating logic, see Logic help.

6. Type a path and file name to which the journal data will be exported.
7. Click OK.

Restoring journal data

The restore journal data task allows you to restore journal entry information from a text file.

To restore journal data

1. Open a package or create a new package in Data Transformation Services on the BPC server.

2. If you have not already done so, register the Ev4DTSDumpload task (Ev4DTSDumpLoad.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task, and then select the Restore option under Journal Management.
5. Fill in the application set, application, and the user ID.

Note: The user ID must be an OutlookSof administrator ID in order to validate logic files. For more information about validating logic, see Logic help.

6. Type a path and file name from which the journal data is restored.
7. Click OK.

Clearing journal data

The clear journal data task allows you to clear journal entry information from a selected data region.

To clear journal data

1. Open a package or create a new package in Data Transformation Services on the BPC server.
2. If you have not already done so, register the Ev4DTSDumpload task (Ev4DTSDumpLoad.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task, and then select the Clear option under Journal Management.
5. Fill in the application set, application, and the user ID.

Note: The user ID must be BPC administrator ID in order to validate logic files. For more information about validating logic, see Logic help.

6. Type a list of dimension members that the clear applies to or click the browse button to located a text file that contains a list of members to clear.
7. Click OK.

Using the Ev4DTSAdmin task

This task enables you to run one of a number of eAdmin tasks through DTS, including:

- Validate logic
- Validate security
- Optimize application
- Process application
- Create dimension

You can add this task to your DTS package by registering Ev4DTSAdmin.dll, located in the BPC websrvr\bin folder, with the Data Transformation services interface. See Registering custom tasks.

EvDTSMModifyPkg task list

You can use the following task commands in the EvDTSMModifyPkg task:

```
TASK(TASKNAME,NAME,DTSTask_Ev4DTSAdmin.clsDTSAdmin_1)
TASK(TASKNAME,DESCRIPTION,TASKNAME)
TASK(TASKNAME,APPSET,)
TASK(TASKNAME,APP,)
TASK(TASKNAME,USERID,)
TASK(TASKNAME,PROCESSMODE,4)
TASK(TASKNAME,OPTIMIZEOPTION,0)
TASK(TASKNAME,PROCESSOPTION,)
TASK(TASKNAME,COMPRESSDATABASE,0)
```

```
TASK (TASKNAME, APPLICATIONLIST, )  
TASK (TASKNAME, LOGICFILE, )  
TASK (TASKNAME, DIMENSIONNAME, )  
TASK (TASKNAME, SOURCETABLE, )  
TASK (TASKNAME, BAPPLICATIONPROCESS, 0)  
TASK (TASKNAME, BTAKESYSTEMAVAILABLE, 0)
```

Validating logic

You can validate logic using the Validate logic option.

Note: This procedure requires some knowledge of SQL Data Transformation Services. Please read the SQL documentation for more complete information on constructing packages.

To validate logic

1. Open a package or create a new package in Data Transformation Services on the BPC server.
2. If you have not already done so, register the Ev4DTSAdmin task (Ev4DTSAdmin.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task, and then select the Validate logic option.
5. Fill in the application set, application, and the user ID.

Note: The user ID must be BPC administrator ID in order to validate logic files. For more information about validating logic, see the Logic help.

6. Select the logic file (*.lgf) to validate, and then click OK.

Tip: You can also use TASK commands in the EvDTSMModifyPkg task to automatically complete the information for you.

Validating security

You can validate security using Validate security option.

Note: This procedure requires some knowledge of SQL Data Transformation Services. Please read the SQL documentation for more complete information on constructing packages.

To use the validate security function

1. Open a package or create a new package in Data Transformation Services on the BPC server.
2. If you have not already done so, register the Ev4DTSAdmin task (Ev4DTSAdmin.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task, and then select the Validate security option.
5. Fill in the application set, application, and the user ID, and then click the OK button.

Note: The user ID must be BPC administrator ID in order to validate security. For more information about validating security, see the Admin help.

Tip: You can also use TASK commands in the EvDTSMModifyPkg task to automatically complete the information for you.

Optimizing the application

Selecting the Optimize application option allows you to automate application optimization using a DTS package. You may need to optimize an application in preparation for, or after running, another task.

Note: This procedure requires some knowledge of SQL Data Transformation Services. Please read the SQL documentation for more complete information on constructing packages.

To use the optimize application function

1. Open a package or create a new package in Data Transformation Services on the BPC server.
2. If you have not already done so, register the Ev4DTSAdmin task (Ev4DTSAdmin.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task, and then select the Optimize application option.
5. Fill in the application set, application, and the user ID.

Note: The user ID must be BPC administrator ID.

6. Select an application from the list.
7. Select Lite, Incremental, or Full optimization and whether you want to compress the database, and then click the OK button.

Note: For more information about the optimization options, see the Admin help.

Tip: You can also use TASK commands in the EvDTSModifyPkg task to automatically complete the information for you.

Processing the application (using Ev4DTSAdmin)

Selecting the Process application option allows you to automate application processing using a DTS package. You may need to process an application in preparation for or after running another task.

Note: This procedure requires some knowledge of SQL Data Transformation Services. Please read the SQL documentation for more complete information on constructing packages.

To process the application

1. Open a package or create a new package in Data Transformation Services on the BPC server.
2. If you have not already done so, register the Ev4DTSAdmin task (Ev4DTSAdmin.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task, and then select the Process application option.
5. Fill in the application set, application, and the user ID.

Note: The user ID must be BPC administrator ID.

6. Select an application from the list.
7. Select Incremental or Full process, and then click the OK button.

Note: For more information about the optimization options, see the Admin help.

Tip: You can also use TASK commands in the EvDTSModifyPkg task to automatically complete the information for you.

Creating dimensions

Selecting the Create Dimension option allows you to automate creating dimensions using a DM package. This task requires advanced knowledge of the BPC member tables.

During normal operation, BPC creates a dimension table during the validation of the member sheet in Excel. This table is the source for processing a dimension into the application.

Using this task, you can set up your source system to produce a replicated member table rather than via Excel. That member table can then be processed into BPC application by this task.

Note: This procedure requires some knowledge of SQL Data Transformation Services. Please read the SQL documentation for more complete information on constructing packages.

To use the Create dimension function

1. Open a package or create a new package in Data Transformation Services on the BPC server.

2. If you have not already done so, register the Ev4DTSAdmin task (Ev4DTSAdmin.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task, and then select the Create dimension option.
5. Fill in the application set, application, and the user ID.

Note: The user ID must be BPC administrator ID.

6. Type the new dimension name and the SQL source table.
7. Select whether to perform a full process after creating the dimension and whether to make the system available after creating the dimension, and then click the OK button.

Note: For more information about the creating dimensions, see the Admin help.

Tip: You can also use TASK commands in the EvDTSMModifyPkg task to automatically complete the information for you.

Ev4DTSSendMail

You can use the Send mail task to send an email message before or after another task has processed. For example, if a task fails, the send mail task can send a fail message to BPC user. Or you can send an email after completion of a certain task. You can send mail to any BPC user and the mail can include one or more attachments.

You can add this task to your DTS package by registering Ev4DTSSendMail.dll located in the BPCwebsrvr\bin folder, with the Data Transformation services interface. See Registering custom tasks.

Note: You must set up the email system parameters in order to enable the system to send email. See Admin help.

EvDTSMModifyPkg task list

You can use the following task commands in the EvDTSMModifyPkg task:

```
TASK (TASKNAME, NAME, DTSTask_Ev4DTSSendMail.clsDTSSendMail_1)
TASK (TASKNAME, DESCRIPTION, TASKNAME)
TASK (TASKNAME, MAILTO, )
TASK (TASKNAME, MAILCC, )
TASK (TASKNAME, MAILSUBJECT, )
TASK (TASKNAME, MAILMESSAGE, )
TASK (TASKNAME, MAILATTACHMENTS, )
TASK (TASKNAME, MAILAPPSET, )
TASK (TASKNAME, MAILUSER, )
TASK (TASKNAME, ADDRRESULT, 0)
```

To use the send mail task

1. Open a package or create a new package in Data Transformation Services on the BPC server.
2. If you have not already done so, register the Ev4DTSSendMail task (Ev4DTSSendMail.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task and fill in the application set, application, and the user ID.

Note: The user ID must be BPC administrator ID.

5. Fill in the TO: and CC: fields. You can type BPC user IDs separated by semi-colons (;) or click the browse button to select from a list of users for either of these fields.

6. Type a subject and a message, and optionally add attachments. You can use the Plus (+) button to add one or more attachments.
7. Click OK to save the task.

Tip: You can also use TASK commands in the EvDTSMModifyPkg task to automatically complete the information for you.

Ev4DTSTrigger

The trigger task allows you to wait for a trigger condition before processing the next task. Triggers include the existence of a specified file or the setting of a flag in a database. You can set up to three trigger conditions that all have to be met in order to exit the trigger successfully. You can also set a retry interval if you want this task to wait for a trigger condition.

You can add this task to your DTS package by registering Ev4DTSTrigger.dll located in the BPC\websrvr\bin folder, with the Data Transformation services interface. See Registering custom tasks.

EvDTSMModifyPkg task list

You can use the following task commands in the EvDTSMModifyPkg task:

```
TASK(TASKNAME,NAME,DTSTask_Ev4DTSTrigger.clsDtsTrigger_1)
TASK(TASKNAME,DESCRIPTION,TASKNAME)
TASK(TASKNAME,APPSET,)
TASK(TASKNAME,USERID,)
TASK(TASKNAME,RETRYCOUNT,)
TASK(TASKNAME,INTERVALTIME,)
TASK(TASKNAME,CONDITION1_TYPE,)
TASK(TASKNAME,CONDITION1_VALUE,)
TASK(TASKNAME,CONDITION2_TYPE,)
TASK(TASKNAME,CONDITION2_VALUE,)
TASK(TASKNAME,CONDITION3_TYPE,)
TASK(TASKNAME,CONDITION3_VALUE,)
TASK(TASKNAME,CONDITION4_TYPE,)
TASK(TASKNAME,CONDITION4_VALUE,)
TASK(TASKNAME,CONDITION5_TYPE,)
TASK(TASKNAME,CONDITION5_VALUE,)
TASK(TASKNAME,CONDITION6_TYPE,)
TASK(TASKNAME,CONDITION6_VALUE,)
TASK(TASKNAME,CONDITION7_TYPE,)
TASK(TASKNAME,CONDITION7_VALUE,)
TASK(TASKNAME,CONDITION8_TYPE,)
TASK(TASKNAME,CONDITION8_VALUE,)
TASK(TASKNAME,CONDITION9_TYPE,)
TASK(TASKNAME,CONDITION9_VALUE,)
TASK(TASKNAME,CONDITION10_TYPE,)
TASK(TASKNAME,CONDITION10_VALUE,)
```

To use the trigger task

1. Open a package or create a new package in Data Transformation Services on the BPC server.
2. If you have not already done so, register the Ev4DTSTrigger task (Ev4DTSTrigger.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.

4. Double-click the task and fill in the application set, application, and the user ID.

Note: The user ID must be BPC administrator ID.

5. Select a retry count. If you select a retry count greater than zero, select the interval time in minutes to perform the retry.
6. Select File name search or Table search. File name search allows you to search for a specified file's existence. Table search allows you to query a database table for a specified condition.
7. Click OK to save the task.

Tip: You can also use TASK commands in the EvDTSMModifyPkg task to automatically complete the information for you.

Ev4DTSAvailable

This System available task allows you to control whether the system is set to available (online) or not available (not online) status. You can set the system to not available, then run some tasks, then set it back to available status using this task.

You can add this task to your DTS package by registering Ev4DTSAvailable.dll located in the BPC\websrvr\bin folder, with the Data Transformation services interface. See Registering custom tasks.

EvDTSMModifyPkg task list

You can use the following task commands in the EvDTSMModifyPkg task:

```
TASK (TASKNAME,NAME,DTSTask_EvDTSAvailable.clsEvDTSAvailable_1)
TASK (TASKNAME,DESCRIPTION,TASKNAME)
TASK (TASKNAME,APPSET, )
TASK (TASKNAME,USERID, )
TASK (TASKNAME,TAKESYSTEMAVAILABLE,0)
```

To use the System available task

1. Open a package or create a new package in Data Transformation Services on the BPC server.
2. If you have not already done so, register the Ev4DTSAvailable task (Ev4DTSAvailable.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task and fill in the application set and the user ID.

Note: The user ID must be BPC administrator ID.

5. Select Available to users or Unavailable to users to control where non-administrator users have access to the system.
6. Click OK to save the task.

Tip: You can also use TASK commands in the EvDTSMModifyPkg task to automatically complete the information for you.

EvDTSFTP

You can use the FTP task to transfer data files from an FTP site to the BPC server.

You can add this task to your DTS package by registering EvDTSFTP.dll located in the BPC\websrvr\bin folder, with the Data Transformation services interface. See Registering custom tasks.

EvDTSMModifyPkg task list

You can use the following task commands in the EvDTSMModifyPkg task:

```
TASK (TASKNAME, NAME, DTSTask_EvDTSFTP.clsEvDTSFTP_1)
TASK (TASKNAME, DESCRIPTION, TASKNAME)
TASK (TASKNAME, SOURCESITE, )
TASK (TASKNAME, SOURCEUSERNAME, )
TASK (TASKNAME, SOURCEPASSWORD, )
TASK (TASKNAME, PORTNUMBER, 0)
TASK (TASKNAME, SOURCEFILENAME, )
TASK (TASKNAME, DESTSITE, )
TASK (TASKNAME, OVERWRITABLE, )
TASK (TASKNAME, NUMRETRIESONSOURCE, 0)
```

To use the FTP task

1. Open a package or create a new package in Data Transformation Services on the BPC server.
2. If you have not already done so, register the Ev4DTSAvailable task (Ev4DTSAvailable.dll) with Data Transformation Services. See Registering custom tasks.
3. Select the task and add it to the package.
4. Double-click the task and fill in the application set and the user ID.

Note: The user ID must be BPC administrator ID.

5. Type the FTP site (tcp/ip address or url of the computer), user name and password to access the ftp site, and the port number (default is port 21, the FTP protocol default).
6. Type the number of times the task should retry to connect to the ftp site.
7. Type or select a destination directory using the browse button.
8. Click the Files tab at the top of the task interface, then select one or more files to transfer. Click the move to destination button to select the files to be downloaded.

Note: You can double-click a directory on the left side to select from a list of files in that directory. Also, you can click the move all to destination button to select all displayed files for download.

9. Click OK to save the task.

Tip: You can also use TASK commands in the EvDTSMModifyPkg task to automatically complete the information for you.

K2DTSRunlogic

K2DTSRunlogic is a DTS version of the BPC Logic module that can be used from within the Data Manager module. This allows you to use the full power of the Logic module in a batch setting. The functionality of this package is similar to the functionality of the Logic Debugger in the Admin module. The main difference is that this package is set to affect the data in the database by default, while the Logic Debugger is set to run in simulation mode by default.

K2DTSRunlogic is especially useful when importing data directly into long term storage (the fact table), which normally bypasses default formulas. See Running and scheduling DTS packages.

You can use K2DTSRunlogic for the following tasks:

- Calculations that can only be executed in batch mode after the appropriate data has been entered into the Application database (the cube). For example, you run an elimination procedure after all intercompany declarations have been entered by all companies. Or, for another example, currency conversion must be re-executed after some exchange rates have been modified.
- For performance reasons. For your situation it might be better to wait until all data is verified correct before running logic on it.

Note: You can use EvDTSModifyPkg tasks to manipulate the values of the K2DTSRunLogic task. Alternative EvDTSModifyPkg tasks are given in the description of the K2DTSRunLogic interface, below. For more information about EvDTSModifyPkg, see Modifying packages.

EvDTSModifyPkg task List

You can use the following task commands in the EvDTSModifyPkg task:

```
TASK(TASKNAME,NAME,DTSTask_K2DTSRunLogic.cdTSRunLogic_1)
TASK(TASKNAME,DESCRIPTION,TASKNAME)
TASK(TASKNAME,USER,)
TASK(TASKNAME,APPSET,)
TASK(TASKNAME,APP,)
TASK(TASKNAME,RUNMODE,0)
TASK(TASKNAME,SELECTION,)
TASK(TASKNAME,DATAFILE,)
TASK(TASKNAME,LOGICMODE,0)
TASK(TASKNAME,LOGICBY,)
TASK(TASKNAME,SCOPEBY,)
TASK(TASKNAME,FORMULASCRIP,T,)
TASK(TASKNAME,LOGICFILE,)
TASK(TASKNAME,DEBUGFILE,)
TASK(TASKNAME,QUERYSCOPE,0)
TASK(TASKNAME,QUERYTYPE,0)
TASK(TASKNAME,DELIMITER,"")
TASK(TASKNAME,SIMULATION,0)
TASK(TASKNAME,CALCULATEDIFFERENCE,1)
TASK(TASKNAME,MAXMEMBERS,)
TASK(TASKNAME,RUNTHELOGIC,)
```

Simulation mode

When the Simulation Mode check box in the K2DTSRunLogic interface is selected, the logic is executed, but the resulting values are not posted to the Application database (the cube). Use this mode to debug your logic formulas without impacting the stored values.

Debug file

Use the Debug file (bottom of the interface) to create a file that contains an audit of the logic execution, and that shows the details of all the generated queries. If no path is entered, the file defaults to the TempFiles directory for the current user. This is especially useful for Simulation Mode.

Note: If the file cannot be created, for example, due to read-only permission on the given path or the file is in use by another user, the logic runs normally, but no debug file is created.

EvDTSMModifyPkg tasks associated with debug file location

```
TASK(TASKNAME,DEBUGFILE,C:\BPC
\Webfolders\ApShell\AdminApp\Finance\debug.log): debug file path
```

Memory management

You can define memory settings to control the size and format of generated queries. Memory settings include Small, Medium, and Large.

The size of the query is only relevant when the data file mode is selected, and should be kept to Large for faster execution. Smaller sizes could be selected only if the large queries take too much memory. The selection of the size of the query should have no impact on the results of the logic calculations.

The field max members allows you to restrict the number of members to process in a query. If the number is exceeded, the query is broken in multiple smaller queries. The syntax is:

```
{dimension} = {number of members}[,{dimension} = {number of members}]
```

If this field is left blank, the default value is ENTITY=1.

This field corresponds to the XDIM_MAXMEMBERS instruction available in the logic script.

EvDTSMModifyPkg tasks associated with Memory Management

TASK(TASKNAME,SCOPEBY,)

Using this instruction can be extremely useful to expand the scope of a logic execution or when data are entered through BPC for Office or through an import file.

If, for example, the formulas to execute should span across all products, irrespective of what products have been modified, it is possible to expand the scope of the execution to all products by redefining the scope_by with a list of dimensions that does not include the product. In this case the instruction could be:

```
*SCOPE_BY = CATEGORY, TIME, ENTITY
TASK(TASKNAME,QUERYSCOPE,0)
*QUERY_SIZE = 0 | 1 | 2
```

where:

2 is the (default) largest size of queries

1 is an intermediate size, and

0 is the smallest practical size

TASK(TASKNAME,MAXMEMBERS,Entity) : Define Maximum member in a Query

The field 'max members' allows the user to restrict the number of members to process in a query.

If the number is exceeded, the query is broken in multiple smaller queries. The syntax is:

```
{dimension} = {number of members}[,{dimension} = {number of members}]
```

If this field is left blank, the default value will be ENTITY=1.

TASK(TASKNAME,RUNTHELOGIC,value) : Value can be 0,1

0: Do not run this task

1: Run this task

Through this property the user can decide at run-time whether a logic should be executed or not, for example, after the completion of an import task.

Example:

In the script of K2ModifyPkg, the following instructions would let the user decide to run a RunLogic task, after importing a file of input data.

```
PROMPT(CHECKBOX,%RUN%,"Run the default logic after import")
```

```
TASK(K2RUNLOGIC,RUNTHELOGIC,%RUN%)
```

```
TASK(TASKNAME,CALCULATEDIFFERENCE,1)
```

This option can be selected to speed up the posting of the calculated values into the database. When a calculated value is sent to the BPC data-write engine, BPC in reality needs to write in the database just the difference between what is already stored and the new value. The calculation of the difference takes time. If such calculation is performed directly by the Logic Module at the time of executing the logic, it is possible that the subsequent posting time will be reduced.

This option defaults to "yes."

MDX query type

MDX query type settings can be used to control the size and the format of the generated queries. The type of query can also have a beneficial (or negative) impact on performance, with the exception of the NonEmptyCrossJoin mode, that should only be used in special cases. NonEmptyCrossJoin is an advanced parameter that can dramatically slow down performance.

EvDTSTaskModifyPkg tasks associated with MDX query type

`TASK(TASKNAME,QUERYTYPE,0)` : Format of MDX queries

While the Logic Module by default generates a multi-axis query, the format of the MDX query can be controlled by the logic using the instruction:

```
*QUERY_TYPE=0 | 1 | 2
```

where:

0 is the default multi-axis type

1 is a row/column type, with multiple crossjoins in rows

2 is a row/column type, with one nonemptycrossjoin in rows

EvDTSTaskModifyPkg

EvDTSTaskModifyPkg is a custom task that can be inserted into any Data Manager package to modify the package itself, when executed. When this task runs, it modifies the properties of objects contained in the package (like tasks and connections), according to some user-defined instructions specified in the task itself. In simple terms, EvDTSTaskModifyPkg provides you with a single interface from which you can specify a modification for any property within the package. This task is often used to replace user-provided information with script-provided information to help with automating the task.

Note: This section explains how to use EvDTSTaskModifyPkg to modify the behavior of packages at run time. For help on building packages please see SQL Server Help. In the sample packages included with BPC, the EvDTSTaskModifyPkg task is called Assign Parameters. However, the underlying file is still named evDTSTaskModifyPkg. Some knowledge of Microsoft DTS packages is recommended for working with the EvDTSTaskModifyPkg task.

EvDTSTaskModifyPkg user interface

The Dynamic Script is displayed in the script area. The dynamic script contains some instructions for modifying the package.

The Display Pkg button lets you view the objects in packages. You can view Tasks, Connections, OLEDBConn, Steps Globals or ALL for any package. To do this you enter the package name and location of the package, then select an object, then click GO.

You can review the list of objects in a package to determine what, if anything you need to modify in the package. If you find a task, for example, that you want to modify you can simply copy the line from the Display Package Script window and paste it into the Debug and/or Dynamic script window above. Then modify its value.

Note: The information displayed here is from the last time the package was saved. If you made changes to another object within the package since the last save it will not be displayed here.

EvDTSTaskModifyPkg syntax

The syntax for package objects (task, connection, step, global) is:

```
OBJECT ( TASKNAME , PROPERTY , VALUE )
```

For example:

```
TASK ( EVIMPORT , FILENAME , IMPORT1 .TXT )
```

Dynamically pass text to logic

You can use the EvDTSModifyPkg task to dynamically pass a text string to logic in Data Manager. For example, a user who wishes to dynamically pass a text string representing a year (which is a portion of the *XDIM_MEMBERSET instruction) could use the following steps:

1. Using the EvDTSModifyPkg task, prompt for the year, i.e., PROMPT(TEXT,%TEXT%,"select a year")
2. Pass the returned %TEXT% to the FormulaScript of the RunLogic task as follows:
TASK(RUNLOGIC,FORMULASCRIPT,"*FUNCTION MYYEAR=%TEXT%")
3. In the Data Manager logic, use the dynamically created function as follows: *XDIM_MEMBERSET TIME=MYYEAR.INPUT.

The logic name in the RunLogic task must be specified with the .LGF extension to enforce its validation at run time.

The Global() instruction

You can use the Global instruction to create a global variable that can be used in any part of the DTS package.

The syntax for the Global instruction is:

```
Global ( "variablename" , "value" )
```

where "variable name" is the name of the variable and "value" is the text string the global variable represents.

The Info() instruction

The INFO instruction is not an executable instruction, in the sense that it does not 'do' anything specific to the package. Its purpose, as stated earlier, is only to define and assign a value to a script variable.

The syntax for the INFO instruction is:

```
INFO("variablename", "value")
```

where "variable name" is the name of the variable and "value" is the value of the variable.

Instead of assigning a value directly to a task, the value (or part of it) can be replaced with the name of a variable defined in an INFO() instruction. At run-time, the task will replace the name of all INFO variables found in all executable instructions with the current value of those variables, before executing the executable instructions.

The following example illustrates how to define a variable with an INFO() instruction, and how to use it in another instruction like TASK().

```
INFO(%FILE%, IMPORT.TXT)
TASK ( "EVIMPORT", "FILENAME", %FILE% )
```

You can also "nest" one level of info variables inside other info variables as the example below depicts.

```
INFO(%PATH%, "C:\MyDir\")
INFO(%FILE%, %PATH%IMPORT.TXT)
TASK ( "EVIMPORT", "FILENAME", %FILE% )
```

The name assigned to the variable is entirely up to your discretion, however, we strongly recommended you use expressions enclosed between some easily identifiable characters, for example the percent character (%VARIABLE%), to make the script more readable and less subject to errors when the substitutions are performed. Note also that all system generated INFO() variables comply with this practice. See the Predefined Constants list, above for a list of system generated INFO() variables.

The GETINFO() instruction

A special case of INFO() instruction is the GETINFO() instruction. This instruction allows you to dynamically generate additional INFO() instructions using some specific parameters.

The syntax of the GETINFO() instruction is:

```
GETINFO([getinfoType], [parameters,...] )
```

Where:

[GETINFOType] is a reserved keyword indicating what info is going to be generated

[parameters,...] is a comma delimited list of required or optional parameters, specific to the selected GETINFO type.

The currently supported GETINFO types are:

- SELECTION_FROM_FILE
- FACTSELECTION_FROM_FILE
- DELETESTATEMENT_FROM_FILE

SELECTION_FROM_FILE and FACTSELECTION_FROM_FILE

You use the following parameters with the SELECTION_FROM_FILE and FACTSELECTION_FROM_FILE types:

- variablename — the INFO variable where to store the result (the default variable name is %SELECTION%)
- file name — the name of the file to scan (required)
- dimensions — the dimensions to scan in the file (the default dimensions are "category,time and entity")

These instructions scan a specified (with the file name parameter) ASCII file and build a SQL query based on all different combinations of certain dimensions found in the file.

Example:

```
GETINFO(SELECTION_FROM_FILE,%SELECTION%,MYFILE.TXT,"CATEGORY,TIME")
```

Content of MYFILE.TXT:

```
CATEGORY, TIME, ENTITY, ACCOUNT, AMOUNT
ACTUAL,2000.JAN,ITALY,SALES,1234
ACTUAL,2000.FEB,ITALY,SALES,2345
ACTUAL,2000.JAN,FRANCE,SALES,3456
ACTUAL,2000.FEB,FRANCE,SALES,4567
BUDGET,2000.JAN,ITALY,SALES,5678
```

The result of the above instruction will be the generation of the following INFO()

```
INFO(%SELECTION%,"( ([CATEGORY]='ACTUAL' and [TIME]='2000.JAN') or _
([CATEGORY]='ACTUAL' and [TIME]='2000.FEB') or _
([CATEGORY]='BUDGET' and [TIME]='2000.JAN')")")
```

Tip: This instruction is useful to dynamically decide what data to clear before importing a user-selected file of data. The following example of script comes from a real-life data import package. Note how the generated %SELECTION% variable is used 'get in some subsequent TASK() instruction.

```
'the data range to clear from the converted file
GETINFO(FACTSELECTION_FROM_FILE, %SELECTION%, %TEMPPATH%%TEMPFILE%)
'Modify Clear fact and wb tables tasks
TASK(DELETEDESTFACT,SQLSTATEMENT,DELETE FROM TBLFACT%APP% where
%SELECTION%)
TASK(DELETEDESTWB,SQLSTATEMENT,DELETE FROM TBLFACTWB%APP% where
%SELECTION%)
```


Tip: The only difference between `SELECTION_FROM_FILE` and `FACTSELECTION_FROM_FILE` is that, when the dimensions to scan are taken by default, the name of the time dimension is left as is (by `SELECTION_FROM_FILE`) or forced to be "TIMEID" (by `FACTSELECTION_FROM_FILE`).

DELETESTATEMENT_FROM_FILE

The syntax for the `DELETESTATEMENT_FROM_FILE` GETINFO type is:

GETINFO(DELETESTATEMENT_FROM_FILE, VariableName, FileName, TableName, DimensionList, Delimiter, TimeOrTIMEID)

VariableName: the INFO variable where to store the result (the default variable name is %STATEMENT%)

FileName: the name of the file to scan (required)

TableName: the name of the table (required)

DimensionList: the dimensions to scan in the file (the default dimensions are "category,time,entity")

Delimiter: Delimiter in the file (default is comma (,))

TimeOrTimeID: 1 for time and 0 for timeID (default is 0)

This instruction scans a specified text file and builds a SQL delete query based on all different combinations of certain dimensions found in the file.

For example:

Content of MYFILE.TXT:

```
CATEGORY , TIME , ENTITY , ACCOUNT , AMOUNT
ACTUAL , 2000 . JAN , ITALY , SALES , 1234
ACTUAL , 2000 . FEB , ITALY , SALES , 2345
ACTUAL , 2000 . JAN , FRANCE , SALES , 3456
ACTUAL , 2000 . FEB , FRANCE , SALES , 4567
BUDGET , 2000 . JAN , ITALY , SALES , 5678
```

Instruction:

```
GETINFO(DELETESTATEMENT_FROM_FILE,%STATEMENT%,MYFILE.TXT,tblFactFinance,"CATEGORY
,TIME")
```

The result of the above instruction will be the generation of the following INFO() statement:

```
INFO(%STATEMENT%, Delete from [tblFactFinance] where ([CATEGORY]='ACTUAL' and
[TIMEID]='20000100') or _
```

```
([CATEGORY]='ACTUAL' and [TIMEID]='20000200') or _
```

```
([CATEGORY]='BUDGET' and [TIMEID]='20000100'))
```

If the TimeOrTimeID options is set to "1" like this:

```
GETINFO(DELETESTATEMENT_FROM_FILE,%STATEMENT%,MYFILE.TXT,tblFactFinance,"CATEGORY
,TIME",,1)
```

Then the following INFO() statement is generated:

```
INFO(%STATEMENT%, Delete from [tblFactFinance] where ([CATEGORY]='ACTUAL' and
[TIME]='2000.JAN') or _
```

```
([CATEGORY]='ACTUAL' and [TIME]='2000.FEB') or _
```

```
([CATEGORY]='BUDGET' and [TIME]='2000.JAN'))
```

The Prompt() command

This command generates customized dialog boxes, and, by answering the requests of these dialogs, the user tells BPC things like: what file to load, what selections of data to export, in which application to copy the selected data, etc.

About the Prompt() command

This command generates customized dialog boxes, and, by answering the requests of these dialogs, the user tells BPC things like: what file to load, what selections of data to export, in which application to copy the selected data, etc.

When multiple PROMPT() commands are entered in the same script, all prompts are combined in one single dialog box and presented to the user as such. This rule applies to all PROMPT() commands asking for the selection of a file, a delimiter or a text, but not to the COPYMOVE and SELECT prompts. The COPYMOVE and SELECT prompts are presented to the user individually, with as many dialog boxes as prompts are found in the script.

All file selection prompts return the name of the selected file with the complete path.

If the user leaves blank any selection field of a SELECT prompt (or BOTH source and destination fields for a given dimension in a COPYMOVE prompt), the system will assume ALL members for this dimension (and no range will be inserted for this dimension in the returned SQL statement).

The syntax for the PROMPT() command is:

```
PROMPT([prompt type], [variable name], [second variable name],
[parameters,...])
```

The currently supported types of PROMPT() commands are:

- INFILE
- OUTFILE
- SELECTION
- TRANSFORMATION
- DELIMITER
- TEXT
- COPYMOVE
- SELECT
- SELECTINPUT
- COPYMOVEINPUT

Predefined paths

BPC supports predefined paths in the INFILE, TRANSFORMATION, and OUTFILE prompts.

The predefined Paths are:

```
%TEMPPATH%
%APPPATH%
%DATAPATH%
%SELECTIONPATH%
%TRANSFORMATIONPATH%
%LOGICPATH%
```

For example:

```
PROMPT(INFILE, %FILE%, "select a file", "*.txt,*.txt", %TEMPPATH%)
PROMPT(INFILE, %FILE1%, "test APPPATH", "*.txt,*.txt",%APPPATH%)
PROMPT(INFILE, %FILE2%, "test DATAPATH", "*.txt,*.txt", %DATAPATH%)
PROMPT(INFILE, %FILE3%, "test SELECTIONPATH", "*.txt,*.txt",
%SELECTIONPATH%)
PROMPT(INFILE, %FILE4%, "test TRANSFORMATIONPATH", "*.txt,*.txt",
%TRANSFORMATIONPATH%)
```

The INFILE Prompt() command

Used to prompt the user for the name of the export file.

PROMPT(OUTFILE, [variable], [label], [mask], [path])

Where	means	Default value
OUTFILE	Select an existing DATA file	n/a
[variable]	The name of the returned variable	%FILE%
[label]	The text to display to the user	Please select a file
[mask]	The types of files to select	Data files (*.txt), *.txt, All files(*.*), *.* *
[path]	The search path	The data files directory

You can use Predefined paths with this prompt type. See Predefined paths.

The SELECTION Prompt() command

Used to prompt the user for a delimiter to be used in the import or export file.

PROMPT(DELIMITER, [variable], [label])

Where	means	Default value
DELIMITER	Select a delimiter	n/a
[variable]	The name of the returned variable	%DELIMITER%
[label]	The text to display to the user	Please select a delimiter

The TEXT Prompt() command

Used to prompt the user for a source and destination data range.

PROMPT(COPYMOVE, [variable], [second variable], [label], [dimensions])

Where	means	Default value
COPYMOVE	Select two data ranges	n/a
[variable]	The name of the returned variable	%SELECTION%
[second variable]	The name of the second returned variable	%TOSELECTION%
[label]	The text to display to the user	Enter your selection
[dimensions]	The dimensions to select from	Category, Time, Entity

The SELECT() Prompt command

Used to allow the user to select only non-calculated members, for export, for example.

PROMPT(SELECTINPUT, [variable], [second variable], [label], [dimensions])

Where	means	Default value
SELECTINPUT	Select one non-calculated data range	n/a
[variable]	The name of the returned variable	%SELECTION%
[label]	The text to display to the user	Enter your selection
[dimensions]	The dimensions to select from	Category, Time, Entity

The TASKCOPYCOLUMNS command

The TASKCOPYCOLUMNS command is an executable command, just like a TASK. It is only used in EvDTSMModifyPkg. This command allows you to dynamically copy column settings of a given transformation.

Syntax

```
TASKCOPYCOLUMNS("TaskDescription","[source columns]","[destination columns]")
```

where:

TaskDescription — is the description property of the task that performs the column copy

[source columns] — is the comma delimited range of columns of the source connection

[destination columns] — is the comma delimited range of columns of the destination connection

Note: The source columns find their corresponding columns in the destination columns on a positional basis. The ranges of the source and destination columns must have the same number of items. Double quotes are required to mark the end of the source columns list and the beginning of the destination columns list. This command is only found in EvModify task. It does not appear when displaying the task properties of a package.

Registering custom tasks

When you create a new Data Transformation Service (DTS) package you have to register BPC's custom DTS tasks to the package in order to use them in that package.

The task is added to the Task list on the left of the package window.

To register custom tasks

1. In SQL Server Data Transformation Services, create or open the package to which you want to add the task.

2. On the Task menu select Register Custom Task.
3. Complete the following fields:
 - a. Task description — this should correspond with the name of the task.
 - b. Location — [x]:/BPC/Websrvr/Bin/[task name].dll
4. Click OK.

Defining data transformations

You can define data transformations so that you can map external data to internal BPC data structures. Data Manager helps you set up data import and export into and out of your BPC applications.

Data transformation is done by setting up at least two Excel workbook files:

- Transformation file — Allows application administrators to set up the rules for reading data from an external source and putting it in the proper form for your BPC database.
- Conversion file — Allows application or site administrators to map member names from external to internal dimension structures. You can set up multiple sheets in a conversion file so that many transformations can access the same conversion workbook. You can have one conversion file per dimension.

After you set up these files, you run packages to use these files to define your data transformations. See [Running packages](#).

Using transformation files

Transformation files define how to change the data from an external source so that it works with your BPC data structure.

Note: Only administrators can edit and create transformation files.

Transformation options

Transformation files are Microsoft Excel files that contain one worksheet, named *Instructions*. The transformation file worksheet has three sections: *Options, *Mapping, and *Conversion.

This topic discusses the *OPTIONS section.

*OPTIONS section

This section of the Transformation file contains definitions for various options that you can set for your transformation.

The options are:

- AMOUNTDECIMALPOINT
- CONVERTAMOUNTWDIM
- CREDITPOSITIVE
- DELIMITER
- EXPORTFORMAT
- FORMAT
- HEADER
- MAXREJECTCOUNT
- NULLAMOUNTVALUE
- OUTPUTDELIMITER
- OUTPUTHEADER
- ROUNDAMOUNT
- SKIP
- SKIPIF
- SPECIFICMAPPING
- VALIDATERECORDS

AMOUNTDECIMALPOINT=<text_character>

Default: Period

Allows you to specify a non-period decimal point for countries that use a different character, such as a comma. Note: the character specified in this option must differ from the character specified for the DELIMITER.

CONVERTAMOUNTWDIM=<dim_name>

Default: Account

Specifies which dimension to look at for value calculations.

You must specify a dimension conversion sheet using the Amount *Conversion option. See Mapping dimensions to conversion files for more information.

If there is no formula in the Formula column of the Amount conversion sheet this parameter has no affect at all. For example, the Formula column has the following formula: Value*1.10. All accounts will be increased by 10% during the conversion.

CREDITPOSITIVE= YES | NO

Default: YES

If No, all amounts referring to an ACCOUNT type (LEQ, INC) will have their signs reversed.

DELIMITER <text_character> | SPACE | TAB

Default: , (comma)

If the FORMAT option is set to DELIMITED, this option defines the single character that is the delimiter between columns. use the keywords SPACE or TAB if space- or tab-delimited.

ExportFormat = CategoryByTime | MultiWithHeader | MultiWithNoHeader

Default: There is no default. This option must be defined.

The following is a sample input file:

```
CATEGORY,TIME,ACCOUNT,ENTITY,DATASRC,INTCO,RPTCURRENCY,AMOUNT
ACTUAL,2008.JAN,ACCPAY,SALESJAPAN,INPUT,NON_INTERCO,LC,-807.990551
ACTUAL,2008.JAN,CASH,SALESJAPAN,INPUT,NON_INTERCO,LC,989.2362356
ACTUAL,2008.JAN,DIRECTLABOR,SALESJAPAN,INPUT,NON_INTERCO,LC,772.5536823
ACTUAL,2008.JAN,PPE,SALESJAPAN,INPUT,NON_INTERCO,LC,276.5432596
ACTUAL,2008.JAN,INVENTORY,SALESJAPAN,INPUT,NON_INTERCO,LC,333.8124752
ACTUAL,2008.JAN,INVENTORY,SALESJAPAN,INPUT,NON_INTERCO,LC,333.8124752
ACTUAL,2008.JAN,OTHEREXP,SALESJAPAN,INPUT,NON_INTERCO,LC,283.995986
ACTUAL,2008.JAN,SALARIES.CAT2,SALESJAPAN,INPUT,NON_INTERCO,LC,141.007185
ACTUAL,2008.JAN,HEADCOUNT,SALESJAPAN,INPUT,NON_INTERCO,LC,-341.2652016
ACTUAL,2008.JAN,ACCUMDEPR,SALESJAPAN,INPUT,NON_INTERCO,LC,212.2526169
ACTUAL,2008.JAN,PPE,SALESJAPAN,INPUT,NON_INTERCO,LC,311.9484186
ACTUAL,2008.JAN,RAWMATERIALS,SALESJAPAN,INPUT,NON_INTERCO,LC,903.4849405
ACTUAL,2008.JAN,STPORTIONLTDEBT,SALESJAPAN,INPUT,NON_INTERCO,LC,-
104.1463614
ACTUAL,2008.JAN,ACCRUEDEXP,SALESJAPAN,INPUT,NON_INTERCO,LC,-40.25912285
ACTUAL,2008.JAN,APIC,SALESJAPAN,INPUT,NON_INTERCO,LC,-499.6262789
ACTUAL,2008.JAN,APIC,SALESJAPAN,INPUT,NON_INTERCO,LC,-441.7678118
ACTUAL,2008.JAN,PPE,SALESJAPAN,INPUT,NON_INTERCO,LC,311.9484186
ACTUAL,2008.JAN,RAWMATERIALS,SALESJAPAN,INPUT,NON_INTERCO,LC,903.4849405
ACTUAL,2008.JAN,STPORTIONLTDEBT,SALESJAPAN,INPUT,NON_INTERCO,LC,-
104.1463614
ACTUAL,2008.JAN,ACCRUEDEXP,SALESJAPAN,INPUT,NON_INTERCO,LC,-40.25912285
ACTUAL,2008.JAN,APIC,SALESJAPAN,INPUT,NON_INTERCO,LC,-499.6262789
ACTUAL,2008.JAN,APIC,SALESJAPAN,INPUT,NON_INTERCO,LC,-441.7678118
```

ACTUAL,2008.JAN,SUPPLIES,SALESJAPAN,INPUT,NON_INTERCO,LC,960.5683088

```

ACTUAL
1
1
SALESJAPAN,ACCPAY,-807.990551
SALESJAPAN,ACCRUEDEXP,-40.2591228
SALESJAPAN,ACCUMDEPR,212.2526169
SALESJAPAN,APIC,-941.3940907
SALESJAPAN,CASH,989.2362356
SALESJAPAN,DIRECTLABOR,772.5536823
SALESJAPAN,HEADCOUNT,-341.2652016
SALESJAPAN,INVENTORY,333.8124752
SALESJAPAN,OTHERCURRENTLIAB,-650.3449678
SALESJAPAN,OTHEREXP,283.995986
SALESJAPAN,PPE,588.4916782
SALESJAPAN,RAWMATERIALS,903.4849405
SALESJAPAN,SALARIES.CAT2,141.007185
SALESJAPAN,STPORTIONLTDEBT,-920.2105999
SALESJAPAN,SUPPLIES,960.5683088

```

CATEGORY, ENTITY, ACCOUNT, DATASRC, INTCO, RPTCURRENCY, 2008 . JAN, 2008 . FEB, 2008 . MAR, 2008 . APR, 2008 . MAY, 2008 . JUN, 2008 . JUL, 2008 . AUG, 2008 . SEP, 2008 . OCT, 2008 . NOV, 2008 . DEC
ACTUAL, SALESJAPAN, ACCPAY, INPUT, NON_INTERCO, LC, - 807.990551, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
ACTUAL, SALESJAPAN, ACCUMDEPR, INPUT, NON_INTERCO, LC, 0, 0, 0, 0, 0, 0, 0, 0, 212.2526 169, 0, 0, 0
ACTUAL, SALESJAPAN, CASH, INPUT, NON_INTERCO, LC, 0, 989.2362356, 0, 0, 0, 0, 0, 0, 0, 0 , 0, 0
ACTUAL, SALESJAPAN, DIRECTLABOR, INPUT, NON_INTERCO, LC, 0, 0, 772.5536823, 0, 0, 0, 0, 0 0, 0, 0, 0, 0, 0
ACTUAL, SALESJAPAN, HEADCOUNT, INPUT, NON_INTERCO, LC, 0, 0, 0, 0, 0, 0, 0, 0, - 341.2652016, 0, 0, 0, 0
ACTUAL, SALESJAPAN, INVENTORY, INPUT, NON_INTERCO, LC, 0, 0, 0, 0, 333.8124752, 0, 0, 0 0, 0, 0, 0, 0
ACTUAL, SALESJAPAN, OTHEREXP, INPUT, NON_INTERCO, LC, 0, 0, 0, 0, 0, 283.995986, 0, 0, 0 0, 0, 0, 0
ACTUAL, SALESJAPAN, PPE, INPUT, NON_INTERCO, LC, 0, 0, 0, 276.5432596, 0, 0, 0, 0, 0, 31 1.9484186, 0.0


```

ACTUAL,SALESJAPAN,RAWMATERIALS,INPUT,NON_INTERCO,LC,0,0,0,0,0,0,0,0,0,0,9
03.4849405,0

ACTUAL,SALESJAPAN,SALARIES.CAT2,INPUT,NON_INTERCO,LC,0,0,0,0,0,0,141.0071
85,0,0,0,0,0

ACTUAL,SALESJAPAN,SALARIES.CAT2,INPUT,NON_INTERCO,LC,0,0,0,0,0,0,141.0071
85,0,0,0,0,0

ACTUAL,SALESJAPAN,STPORTIONLTDEBT,INPUT,NON_INTERCO,LC,0,0,0,0,0,0,0,0,0,
0,0,-104.1463614

```

Using the MultiWithNoHeader option generates the same output file as MultiWithHeader without the header titles of Category, Entity, Account, and so on.

FORMAT= DELIMITED | FIXED | CATEGORYBYTIME

Default: There is no default. This option must be defined.

The format of the data in the input or output file. You must enter a format type. Delimited means that there is a special character between each column of data, defined by the Delimiter option (below). Fixed means that the data is in a fixed field format. Use the FCOL mapping option, defined below. CategoryByTime means that the data is listed by categories that represent time periods.

HEADER= YES | NO

Default: YES

If YES, then your input file contains one header row that defines the fields. If you do have a header row, you can refer to a field by name in the MAPPING section.

MAXREJECTCOUNT= <empty_string> | -1 | positive number

Default: <empty_string>

If validating records, the number of rejected records at which to stop processing. A value of -1 implies that DM should keep processing no matter how many rejected records exist. The default value is 500, which can be represented by an empty string.

NULLAMOUNTVALUE= <text_character>

Default: a space character

Allows you to assign a value to null records.

OUTPUTDELIMITER= <text_character> | SPACE | TAB

Default: , (comma)

Same as DELIMITER, but used to define the delimiter when using the transformation to export data.

OUTPUTHEADER= <text_string>

Default: empty string

Allows you to use a custom header to export data. Is ignored if the transformation file is used to import data. You can use {CRLF} to create more than one row for the header. For example:

MyCompany Data{CRLF}Category,Time,Account, Entity,Datasrc,Intco,RPTCurrency,Amount

Note: The above example should be placed all in the same cell in the transformation file.

ROUNDAMOUNT= <integer>

Default: no rounding occurs

Specifies the amount of decimal places to round values during the transformation.

SKIP= <integer>

Default: 0 (zero)

Number of lines to skip at the top of a data file. If your data file has a header, set this value so that those lines are skipped during transformation.

SKIPIF= <text string> | NULL

Default: <empty_string>

Skip a line in the data file if it begins with the specified string. Setting this option to an empty string means that no lines are skipped in the body of the data file. Header lines can still be skipped using the SKIP command, above. You can skip null database records by setting SKIPIF=NULL.

SPECIFICMAPPING= No | Yes

Default: No

Allows you to turn off the automatic dimension mapping feature. When set to YES, Data Manager does not automatically map dimension names. This is especially useful for data export.

VALIDATERECORDS= No | Yes

Default: No

If YES, validate the mapping and that members exist, and map data in the proper order for BPC. For example, the Amount field is always moved to the end of the records even though it might be written in the first line of the mapping section.

If NO, will not validate. In this case, if the mapping section is out of order or just wrong, fields will be skipped.

Mapping options

This section of the transformation file defines how data is mapped to your BP database.

*MAPPING section

The following options are available:

- *COL(A)
- *COL(A,B: C)
- *FCOL(A:B)
- *MVAL(A:B)
- *NEWCOL(A)
- *Str(string)
- *Pad(A,B,C,D)
- *If(Condition1 then Action1; Condition2 then Action2; Default Action)

Note: By default, Data Manager associates fields to header names when you have a header row in your data file. (To change the default, see SPECIFICMAPPING.) This makes it easier to define mappings in this section of the transformation file. For example, if you have a header row and it defines a column named "AccountVal," you can map the Account dimension to the Account field from the data file this way:
Account = AccountVal

*COL(A)

A = column index in the data file

Defines a dimension for a field in the data file when the data file does not have a header row. See the note below about automatic field names when you do have a header row.

Example:

```
Account = *COL( 2 )
```

*COL(A,B: C)

A = column index in the data file

B: C = start/end position within the column

Defines a dimension for a subset of a field in the data file.

Example:

```
Account= *COL( 3 , 1 : 4 )
```

*FCOL(A:B)

A:B = start/end position within the data row

When using Fixed format data files, this option allows you to define the start and end character columns for each field. In the example below, the columns 4 through 7 represent the Account dimension.

Example:

```
Account=*FCOL(4:7)
```

*MVAL(A:B)

A:B = start/end column of time period

Use this command if your data file has multiple period columns. For example, if you have columns that are like this:

Account,Category,DATASRC,entity,IntCo,RptCURRENCY,JAN,FEB,MAR,APR,MAY,JUN,JUL,AUG,SEP,OCT,NOV,DEC

You would use Time = *MVAL(7:18), since columns 7 through 18 contain data based on time periods. Using this command requires that you use a Time.xls conversion sheet. See the sample Time.xls conversion file, located in the ConvesionFiles folder, for more information.

Note: If your data does not have a header row, the time periods will be named P1, P2, P3,...,PN by default.

Example:

```
Time = *MVAL(7:18)
```

*NEWCOL(A)

A = dimension member

Creates a new field with the given value.

Example:

```
Account=*NEWCOL(Revenue)
```

*Str(string)

Adds a text string to the members of a column. Useful if you need to map data file fields to fields in your database that have the same names, except they have extra characters appended or in front of the field name.

Example:

```
Entity = *Str(NE) + *COL(1)
```

*Pad(A,B,C,D)

A = Field name

B = Total length

C = Padding direction

D = Padding character

Use to format exported data into fixed-width records. Use in conjunction with the "None" delimiter. The Padding character is optional and is a space by default.

Example:

```
Account=*Pad(Account,10,L)
Account=*Pad(Account,10,L,0)
```

*If(Condition1 then Action1;Condition2 then Action2;Default Action)

Condition1 - If this evaluates to "True," map using Action1

Condition2 - If Condition1 is "False" and Condition2 is "True," map using Action2

Default Action - If both Condition1 and Condition2 are "False", map using this Default Action

Condition1 and Condition 2 can contain multiple items that are added together (using the plus (+) sign), as in this example:

```
Product=*if (Product+ID+Entity = *str(148552) then *str(MHarn);ID(1:1) =
*str(C) then *str(XX) ; *str(YY))
```

If Product + ID + Entity value equals "148552" then Change Product value to "MHarn".

If the first value of ID field equals "C" then Change Product value to "XX" or change product value to "YY"

Note: The *STR() function must be used when evaluating numeric constants in an *IF statement.

Example 1

Given this data:

```
PRODUCT, ID, ENTITY
14, 85, 52
AB, CD, EF
GH, IJ, KL
```

And the mapping formula as shown below:

```
Product=*IF (product+ID+Entity=*Str(148552) then *str(MHarn);ID(1:1) =
*str(C) then *str(XX) ; *str(YY))
```

The result is as follows:

```
PRODUCT, ID, ENTITY
MHarn, 85, 52
AB, XX, EF
YY, IJ, KL
```

Example 2

Given this data in the following example:

```
Entity, SEntity
U1000000, US01
Z2000A01, CA03
K3430000, JP04
```

And with the mapping formula shown below:

```
Entity=*IF(*col(1,1:1)=U then SEntity;*col(1,1:1)=Z then *col(1,3:6);
*STR(ERR))
```

You see the following result:

```
Entity, SEntity
US01, US01
A01, CA03
ERR, JP04
```

Mapping dimensions to conversion files

Conversion files define how to map the data contained in your input files.

*CONVERSION Section

This section of the transformation file defines which conversion sheet(s) to use with which dimensions, and has the following syntax:

Dimension Name = [COMPANY]WorkbookName[!SheetName]

Items in brackets are optional.

DimensionName is either the dimension name to correlate with the conversion file or the keyword Amount.

Note: Use the Amount option to specify the conversion file to be used if you specify the ConverAmountWDim option in the *Options section. This allows you to use a formula to scale the input or output values.

Workbookname is the name of the conversion file.

SheetName = is the name of the worksheet to use within the conversion file. If not specified, Data Manager assumes the sheet name is: Conversion

If [COMPANY] is defined then DM will try to obtain the conversion file from the Main company Data Manager folder, otherwise Data Manager looks in the appropriate Site folder.

Examples:

Where AccountConv.xls is the name of the Account dimension conversion file.

Account = AccountConv.xls

Amount = AccountConv.xls (Use the Amount keyword with the ConvertAmountWDim transformation option)

Account = AccountConv.xls!newaccount

Account = [COMPANY]AccountConv.xls!newaccount

Mapping dimensions between applications

When you want to copy dimension files between applications, but the dimension names are different, you can create a transformation file to map the dimensions.

For example, if the dimensions in the source application are:

Category, Account, Entity, Time, Rptcurrency, Intco, and Datasrc

and the dimensions in the target application are:

Cat, Acc, Ent, Tim, Currency, Int, and Dat

You can create a transformation file like the following:

```
*OPTIONS
FORMAT = DELIMITED
HEADER = YES
DELIMITER =
SKIP = 0
SKIPF =
CREDITNEGATIVE=NO
CONVERTAMOUNTWDIM=
MAXREJECTCOUNT=
VALIDATERECORDS=YES

*MAPPING
cat=category
Acc=Account
Ent = Entity
Tim=time
Currency=Rptcurrency
int=intco
dat=Datasrc

*CONVERSION
```

Creating and editing transformation files

You can create and edit transformation files to define how you want data to be transformed during import or export from BPC.

To create or edit transformation files

1. Do one of the following:
 - To create a new Transformation file, select eData > New Transformation File.
 - To edit an existing Transformation file, select eData > Manage Transformation Files, choose a site in the left column of the dialog box, choose a folder, and then double-click an existing Transformation file name.
2. Fill out or define the options, mapping, and conversion file sections.
3. Save the Transformation file.

Copying transformation files

You can save a copy of an open transformation file.

Note: This section only applies to application administrators.

To copy transformation files

1. Start BPC for Excel, and select Manage Data from the Getting Started - BPC for Excel action pane.
2. From the Manage Data Options action pane, select Maintain transformations.
3. Select Manage existing transformation, select a file and click Open.
4. Select Copy Transformation File.
5. Select a file and click the Save button.

Using conversion files

Conversion files define the mapping by dimensions from external member names to internal member names. They also provide the ability to perform arithmetic and rounding on data as it is converted.

You create one conversion file per dimension in a transformation. Each conversion file can contain one or more sheets for different types of data transformations. As a best practice, conversion files should be named the same as the dimension for which they are being used.

Note: Data Manager supports VB script in conversion files. While VBA can be used successfully in some instances, we cannot guarantee it will work with BPC for Excel.

Special options in conversion files

You can use special options when you define your conversion files. This section explains how to use the *skip keyword and how to use wildcard characters.

*skip

If you want to ignore certain external data, you can place the keyword *skip in the Internal column. For example:

External	Internal
ACCPAY	*SKIP

Wildcards

You can use the asterisk (*) and question mark (?) wildcards in the External and/or Internal columns. asterisk (*) stands for ANY character, while question mark (?) stand for any SINGLE character.

For example, if you want to reference all members, use the asterisk (*). This would be useful to apply a formula to all members:

External	Internal	Formula
*	*	value*1.10

Validating conversion files

After you define a conversion file, you can validate it.

To validate a conversion file

1. Create or edit a Conversion file. See Creating and using conversion files.
2. Select eData > Validate & Process Conversion File.

Copying conversion files

You can save a copy of an open Conversion file.

To copy Conversion files

1. Start BPC for Excel, and select Manage Data from the Getting Started - BPC for Excel action pane.
2. From the Manage Data Options action pane, select Maintain Conversions.
3. Select Manage Existing conversion definition, select a file and click Open.
4. Select Copy Conversion.
5. Select a conversion file and click the Save button.

Data transfer and preview

Data Manager allows you to transfer data files to and from the server. You can also use the data preview feature to preview database, excel, or text data files. This allows you to see the format of your data when deciding on how to set up a transformation.

Transferring Data Files

In order to import a file, the file must reside in the Data Manager Data Files folder on the server. The Upload function allows you to move files from your local machine to the Data Files folder on the server. The Download function allows you to move files from the Data Files folder on the server to your local hard drive.

Uploading files

You can upload files to the Data Manager folders on the server. This is especially useful for transferring data files to the server.

To upload files

1. From BPC for Excel, select Manage Data from the Getting Started action pane.
2. Enter the path to the file you want to upload in the Source File text box or click the browse files button to find the file on your hard drive.
3. After entering the file name, you can click the preview button to preview the file.
4. Enter the path to the destination folder in the Destination File text box or click the browse files button to choose a location within the Data Files folder.
5. Click the Transfer button.

Downloading files

You can download files from the Data Manager folders on the server to your local hard drive. This is useful if you export data from BPC and need to use it to feed into another system.

To download files

1. From BPC for Excel, select Manage Data from the Getting Started action pane.
2. Enter the path to the file you want to download in the Source File text box or click the browse files button to find the file in the Data Files folder.
3. After entering the file name, you can click the preview button to preview the file.
4. Enter the path to the destination folder on your local hard drive in the Destination File text box or click the browse files button to choose a location on your hard drive.
5. Click the Transfer button.

Previewing data

You can preview your data files. This is useful when you want to see the way your data is laid out when creating a transformation file. You can preview Database (mdb), Excel, and ASCII text data files.

You can preview data files that reside on the server in Company site or other Site directories. If you want to transfer data files to the server, see Transferring data files.

To preview data

1. Open Data Manager, and select eData > Data Preview.
2. Select a site on the left side.
3. Select a directory, and then a data file, and then click OK.
4. Do one of the following depending on the type of file you opened:
 - a. For text files, choose Delimited or Fixed-width from the Select a data type list. If you chose Delimited, select a delimiter from the Select a delimiter list.
 - b. For Excel files, select a worksheet from the Select a worksheet list.
 - c. For database files, select a table from the Select a table list.
5. Use the scroll bars to look at the data.

Note: The preview window loads only the first 200 lines of your data.

6. If you want to open a different file, click the Open button, select another file, and then click OK.
7. Click the Cancel button when you are done previewing your data.

Index

*	AuditBackup package	13
*COL(A)	AuditClear package	13
*CONVERSION	AuditPurge package	13
*CONVERSION Section	AuditRestore package.....	13
*If(.....	C	
*MAPPING section	Clear from FactTable package	13
*NEWCOL(A)	Clear package	11
*options	Clear the Journals Tables package	13
*OPTIONS section	Clearing data from SQL	27
*skip	Clearing journal data.....	28
*Str(string)	COL(A)	48
A	CONVERTAMOUNTWDIM.....	46
Accessing Data Manager	Copy Application package	13
adding packages	Copy from Fact Table package.....	13
administrative packages	Copy package.....	11, 18
AdminTask_MakeDim package.....	copying packages	15
AdminTask_Optimize package	CREDITNEGATIVE	46
AdminTask_Process package	D	
AdminTask_Security package	Data Preview.....	55
AdminTask_Validate package	default formulas	12
Append into Fact Table package.....	DELETESTATEMENT_FROM_FILE	39
Append package	DELIMITER.....	46
Assigning Packages to Sites	dimensions, mapping	52

DTS packages, samples	13	M	
E		mapping dimensions	52
editing packages	15	MAXREJECTCOUNT	46
Ev4DTSAvailable	34	member selector	9
Ev4DTSSendMail	31	Modifying packages	17
Ev4DTSTrigger	33	Move from Fact Table package	13
EvDTSFTP	35	Move package	11
Export from Fact Table package	12	moving packages	15
Export package	11	N	
Export the Journal tables package	13	NEWCOL(A)	48
Exporting from Cube	26	Nightly Process package	13
Exporting from SQL	26	NULLAMOUNTVALUE	46
Exporting journal data	27	O	
F		Organizing packages	15
FACTSELECTION_FROM_FILE	39	OUTPUTDELIMITER	46
FORMAT	46	OUTPUTHEADER	46
FTP task	35	P	
FX Restatement package	11	packages, maintaining	15
H		Pad(A,B,C,D)	48
HEADER	46	Previewing Data	55
I		processing dimension members	20
IC Eliminations package	11	Processing the application set	24
If(.....	48	R	
Import Category by Time into Fact Table package	13	removing packages	15
Import Category by Time package	13	Restore the Journals Tables package	13
Import Dimension package	13	Restoring journal data	27
Import Excel into Fact Table package	13	Running and scheduling Data Manager packages	18
Import Excel package	13	S	
Import into Fact Table with Clear package	12	sample DTS packages	13
Import package	11	saved prompt values	9
Import SQL into Fact Table package	13	Scenario Modeling package	13
Import SQL package	13	scheduling Data Manager packages	18
ImportAndSendmail package	13	scheduling dimension member processing	20
Importing into Cube	25	SELECTION_FROM_FILE	39
Importing into SQL	25	send an email message	31
ImportUsingFTP package	13	Simulation mode	36
L		SKIP	46, 53
LoadRootCause package	13	SKIPIF	46
logic with packages	15	SPECIFICMAPPING	46
		standard packages	11

Str(string) 48

System available 34

T

Testing packages 17

Transformation options 46

trigger task 33

V

VALIDATERECORDS 46

Viewing package schedule status 19

W

Wildcards 53