

**How-to Guide
SAP CPM**



How To Use MDX Dimension Formulas

Version 1.00 – November 8, 2006

**Applicable Releases:
SAP BPC 5.X**

© Copyright 2007 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, and Informix are trademarks or registered trademarks of IBM Corporation 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, R/3, mySAP, mySAP.com, xApps, xApp, 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.

These materials are provided "as is" without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement. SAP shall not be liable for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials.

SAP does not warrant the accuracy or completeness of the information, text, graphics, links or other items contained within these materials. SAP has no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third party web pages nor provide any warranty whatsoever relating to third party web pages.

SAP CPM "How-to" Guides are intended to simplify the product implementation. While specific product features and procedures typically are explained in a practical business context, it is not implied that those features and procedures are the only approach in solving a specific business problem using SAP. Should you wish to receive additional information, clarification or support, please refer to SAP Consulting.

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, except if such damages were caused by SAP intentionally or grossly negligent.

1.1 Purpose

This document has been written to provide guidance to field consultants for writing dimensional formulas in SAP BPC 5.x. The optimized schema design in SAP BPC 5.x has changed how Microsoft Analysis Services (AS) is used behind the scenes. In this design AS dimensions are used to implement alternative hierarchies. In release 4.2, the application used to have only one hierarchy with different trees to support H1, H2 etc. In 5.x, H1 and H2 are real OLAP hierarchies in dimensions.

In SQL 2000, AS does not recognize the separate dimensions as a single dimension when considering dimension formulas. In SQL 2005, AS does recognize the separation as a single dimension but consider formulas by hierarchy. These differences have created the need for this document.

Because of this change, dimension formulas (or any MDX formula) that used to work in 4.x won't work in BPC 5.X when multiple hierarchies are involved.

- When using SQL 2005 the use of MDX is not recommended. Using MDX can result in significant loss of performance. An SP scheduled to be released from Microsoft may increase performance but this has not been tested.
- Members used in formulas need to be fully qualified with dimension and hierarchy names.
- Members from different hierarchies (in one dimension) cannot be used in the formula at the same time.

1.2 NOTE

As in all versions, the use of MDX is not recommended. We realize that MDX appears to be simpler and performs well when you are in development. However, once you roll out an application MDX will significantly degrade the performance when multiple users are active. It is therefore recommended that you avoid the use of MDX when ever possible.

This includes your Excel templates. MDX will always require full qualification of the hierarchy. Consider the implications of using MDX:

- Removal of a hierarchy in any dimension referenced in a MDX formula will require manual maintenance to fix the formula. For example if you reference the entity dimension in an account formula, removing one of the hierarchies in the entity dimension will require you to edit the account dimension formulas.
- Changing the structure of a hierarchy in any dimension referenced in a MDX formula will require manual maintenance to fix the formula. For example if you reference a member to be included in another hierarchy of that dimension, all formulas referencing that member will have to be adjusted for the new hierarchy relationship.

1.3 Formula definition for SQL 2000

SAP BPC 5.x implements multiple hierarchies as separate dimensions in AS. For example if the account dimension has two hierarchies AS will create two dimensions named account.h1 and account.h2. The top member of these dimensions will be named [all account.h1] and [all account.h2.] All members of the dimension roll up to this top level member. An understanding of this concept is important because it relates to how MDX formulas need to be written.

Suppose we want to define a formula as following in a dimension with 3 hierarchies:

```
#CF_ST_FIN_RATE= ((#IFRSTRATE/#WKSYSR)*#WKSCURMTH)
```

Where IFRSTRATE, WKSYSR and WKSCURMTH are base members of hierarchy H1 of Account dimension, and CF_ST_FIN_RATE is a dimension formula in H1 of account dimension.

The correct formula for H1 is

```
IIF(Account.H2.CurrentMember is Account.H2.[All Account.H2] And  
    Account.H3.CurrentMember is Account.H3.[All Account.H3],  
    (Account.H1.IFRSTRATE/Account.H1.WKSYSR)*Account.H1.WKSCURMT  
H, NULL)
```

An example of this formula for H2:

```
IIF(Account.H1.CurrentMember is Account.H1.[All Account.H1] And  
    Account.H3.CurrentMember is Account.H3.[All Account.H3],  
    (Account.H2.IFRSTRATE/Account.H2.WKSYSR)*Account.H2.WKSCURMT  
H, NULL)
```

Note:

1. All variables need to be fully qualified with dimension name and hierarchy name, such as account.h1.IFRSTRATE. Otherwise AS returns a syntax error.
2. When defining a formula on H1, you have to specify that formula returns a valid result only if the current members on other hierarchies are at the top level of each hierarchy.

1.4 Discussion of the formula behaviors

Dimension formula works in most cases if it follows the above rules. The following discussion is for the purpose of understanding and debugging dimensional formulas.

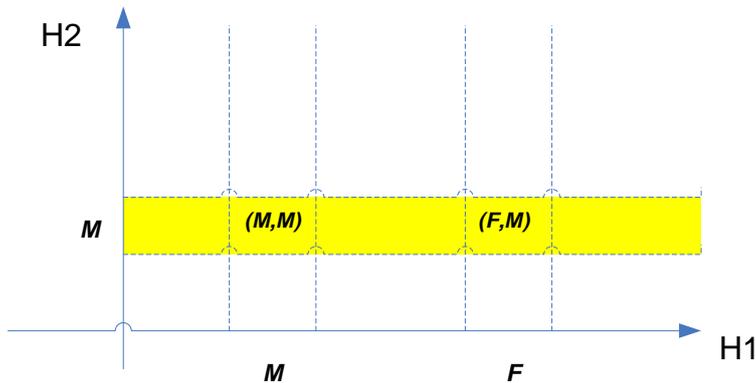
1.4.1 Data accuracy

We will run into data accuracy issues if you don't qualify your formula with these IIF statements.

Suppose you define the formula without IIF statement as following:

$(\text{Account.H1.IFRSTRATE}/\text{Account.H1.WKSYR}) * \text{Account.H1.WKSCURMTH}$
in FormulaH1 field, value of Account.H2.IFRSTRATE could return 1.#INF.

As you can see in following diagram, If you define a dimension formula F on hierarchy H1, and query value of member M on hierarchy H2, (H1.[ALL Dimension.H1], H2.M), Analysis Service returns aggregated value of the yellow slice as shown below:



There are two valid cells on the yellow slice, (H1.M, H2.M) and (H1.F, H2.M). (H1.M, H2.M) is the "normal" cell that returns the right value, the formula needs to ensure that (H1.F, H2.M) returns a NULL so that (H1.[ALL Dimension.H1], H2.M) returns the right value.

1.4.2 Fully qualified member definition (within a dimension)

Following is a correct and more efficient, but verbose and complicated formula for defining CF_ST_FIN_RATE

IIF(Account.H2.CurrentMember is Account.H2.[All Account.H2] And
Account.H3.CurrentMember is Account.H3.[All Account.H3],
(Account.H1.IFRSTRATE, Account.H2.IFRSTRATE,
Account.H3.IFRSTRATE) / (Account.H1.WKSYR, Account.H2.WKSYR,

```
Account.H3.WKSYR) * (Account.H1.WKSCURMTH,  
Account.H2.WKSCURMTH, Account.H3.WKSCURMTH),  
NULL)
```

By fully qualifying member definition with a tuple, the value of each member is pin pointed, and avoids confusion and the cost of further evaluation. The downside is that formula becomes complicated and some Analysis Service functions can't use tuple as parameter.

Note:

To avoid the 255 character limit that exists for dimension formulas it is recommended that you add functions to the logic libraries. For example these two functions can be added to the logic library:

```
Function Hir1(%Arg1%)  
    iif(ProdAcct.H2.CurrentMember is [ProdAcct].[H2].[All  
ProdAcct.H2],%Arg1%, Null)  
*endfunction
```

```
*Function Hir2(%Arg1%)  
    iif(ProdAcct.H1.CurrentMember is [ProdAcct].[H1].[All  
ProdAcct.H1],%Arg1%, Null)  
*endfunction
```

To define the calculation:

```
ProdAcct.H1.Cost1+ ProdAcct.H1.Cost2
```

The syntax would be:

```
Hir1(ProdAcct.H1.Cost1+ ProdAcct.H1.Cost2) for FormulaH1 column
```

```
Hir2(ProdAcct.H2.Cost1+ ProdAcct.H2.Cost2) for formulaH2 column
```

Adding these functions to the logic library will avoid the 255 character limitation issue with SQL DTS.

1.4.3 Evaluation order

By default, AS evaluates each member in a tuple according to the order of hierarchies which have been added to the cube. BPC 5.X SP1 assigns solve order 0 to each formula if no solve order is explicitly specified in order to make sure the dimension formula is evaluated first within a tuple.

1.5 Formula definition for SQL 2005

Currently the use of dimension formulas in SQL 2005 is not recommended due to significant performance degradation. Some test results have shown response times going from 2 to 90 seconds when dimensional formulas are used.

The presence of a formula property in a dimension will also cause performance degradation, even if the property column is left blank.

This is a Microsoft issue with SQL 2005. Microsoft has told us they will be releasing an SP for AS that should improve performance. This has not been tested and verified so at the present time it is not recommended to use dimensional formulas or create a formula property on a dimension when using SQL 2005.

SQL 2005 correctly handles the evaluation of the hierarchies, therefore, you are not required to write conditional formulas (IIF) nor does the software have to control the solve order (It will be set as 0 by default). However, you must define a formula for each hierarchy.

When a member in H1 has a formula that refers to a member in H2, the formula should use H1 member name instead of H2.

Here is an example. User wants to write a formula as below.

MemberA of H1 = MemberB of H2 + MemberC of H1

In that case, formulah1 should be MemberA of H1 = MemberB of H1 + MemberC of H1

It will not make a difference if the formula assigns MemberB in H1 because the application will automatically add all members to all hierarchies behind the scenes. To ensure correct result, you should use MemberB of H1 instead of MemberB of H2.

1.6 Rules for both SQL versions

When maintaining dimension files in the admin console some additional columns have been added to the layout. For SP1 formulas need to be defined for each hierarchy.

Here is an example of this new dimensional format for SP1:

ID	Description	ParentH1	FormulaH1	ParentH2	FormulaH2
Dev1	Development One	Asia	Formula with H1 reference	Dev	Formula with H2 reference
Dev2	Development Two	Asia	Formula with H1 reference	Dev	Formula with H2 reference

If you are going to use Dimensional formulas you will need to add an inapp property to the dimension called formula.

If a dimension has only one Hierarchy (parentH1) the dimensional formulas can be created in the formula column there is no need for a formulah1 column. If a dimension has multiple hierarchies you will still need to create an inapp property called formula, also ensure that there are inapp properties called formulahn for each hierarchy in the dimension.

We recommend that all field resources help your customer understand the importance of properly defining MDX and the reasons to avoid MDX.

1. All MDX formulas must be qualified:
Dimension. Hierarchy Number. Member (Account.H1.Sales)

2. All members referred to in a formula must exist in the referenced hierarchy (real defined parents in SQL 2000 or system assigned in SQL 2005)
3. Using IS operator instead of = operator
4. Member references are defined as tuple for all the hierarchies in that dimension
5. Formula length can be a challenge when managing multiple hierarchies in a single dimension. The use of the formula library is recommended to avoid the 255 character length issue in Excel.
6. Limit the number of hierarchies that have formulas to get better performance in both SQL 2000 and 2005. (Ideally all formulas should be in one hierarchy H1 and there is no formula in H2, H3, and etc.)
7. Recommendation to not use Formula property or dimensional formulas with 2005.

Note:

A key consideration is it is extremely difficult for the application to validate the formulas and if you use the generic Query Analyzer you may not detect a problem. Only when the formula is assigned in the dimension will you fully see the ramifications of not following the guidelines in this document.

www.sdn.sap.com/irj/sdn/bpx-cpm