

IBM Cognos Proven Practices: IBM Cognos BI - Return All Leaf Members of an Unbalanced Hierarchy

Nature of Document: Tip or Technique; Product(s): IBM Cognos Report Studio; Area of Interest: Reporting

Marc Reed
Consultant
IBM

29 June 2012

Unbalanced hierarchies are often found within OLAP data sources. One of the most common examples is a Chart of Accounts dimension. These are dimensions where not all members drill to the same depth as each other. This document illustrates the use of dimensional functions to return all leaf level members from such hierarchies.

[View more content in this series](#)

Introduction

Purpose

Unbalanced hierarchies are often found within OLAP data sources. One of the most common examples is a Chart of Accounts dimension. These are dimensions where not all members drill to the same depth as each other.

There is often a need to show all the leaf members of such dimensions when creating dimensional reports. For example, creating a flat list of accounts for a trial balance.

This technique shows how to add all the leaf members from an unbalanced hierarchy to a report.

Applicability

This technique is applicable to IBM Cognos 8 and upward versions of Report Studio.

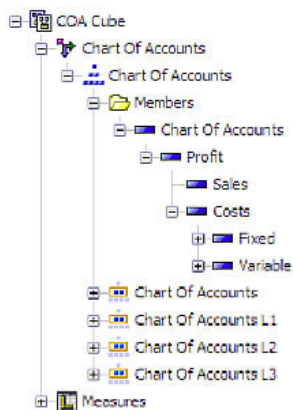
Assumptions

This document assumes knowledge of and experience with report authoring in IBM Cognos Report Studio.

Example

Here is one example. Illustration 1 below shows an unbalanced hierarchy where there are inconsistent depths for the levels. Under the Chart Of Accounts member, there is a Profit member. The Profit member drills down to a Sales and a Costs member. There are no members under Sales, but Costs drills down to more members, Fixed and Variable in this case.

Illustration 1: An unbalanced hierarchy shown within Report Studio's data tree



Within this dimension there are three leaf members:

1. Sales.
From the level [Chart Of Accounts L2]
2. Fixed.
From the level [Chart Of Accounts L3]
3. Variable.
From the level [Chart Of Accounts L3]

How can the report author create a report that shows the data for all the leaf members?

In our basic example, we could simply select the three members, but what if the dimension contained thousands of members? We cannot use the level functions as Level 3 does not contain all the leaf members. Also Level 2 contains leaf and non-leaf (the Costs member) members.

The Report

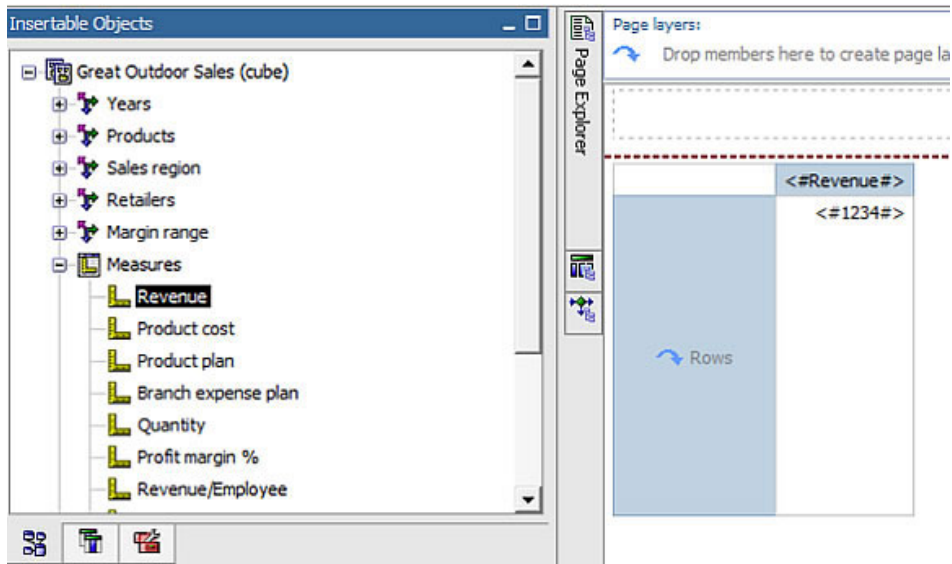
To address this problem we make use of the fact that leaf members have no children. From a high level, all we need do is filter the Chart of Accounts hierarchy by only returning those members where the children count is 0.

To illustrate this in a manner that you can test yourself using the IBM Cognos sample data, we will use the standard IBM Cognos Great Outdoors Sales (cube) PowerCube (great_outdoors_sales_en.mdc). Unfortunately the sample IBM Cognos cubes do not contain an unbalanced hierarchy, however, we can still demonstrate the technique with this cube.

1. Open **IBM Cognos Report Studio**, select the **Great Outdoors Sales (cube)** package, and then create a new **Crosstab** report.

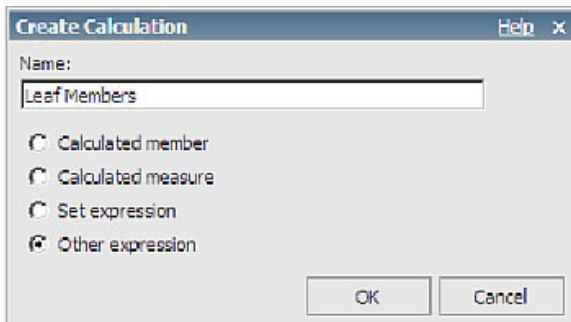
2. Drag the **Revenue** measure into the **Columns** drop zone as shown in Illustration 2.

Illustration 2: Report Studio with a crosstab report showing revenue added to the Columns drop zone



3. From the **Toolbox** tab under Insertable Objects, add a **Query Calculation** to the **Rows** drop zone. The Create Calculation dialog appears. If using IBM Cognos 10.1 and up, there will be more options than just a Name field.
4. In the **Name** field, type **Leaf Members**, and if using IBM Cognos 10.1 or higher, select **Other expression** (see Illustration 3).

Illustration 3: The Create Calculation dialog window



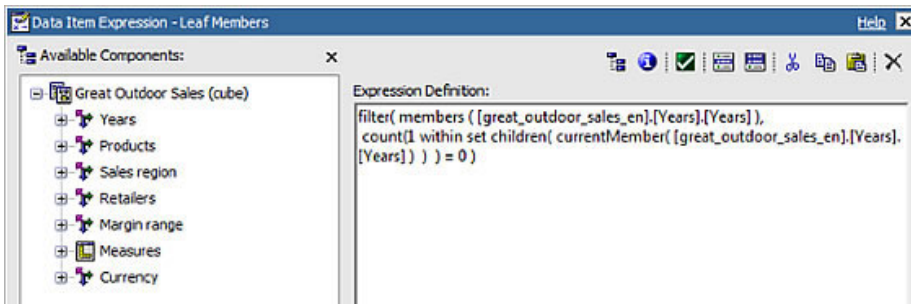
5. Click **OK**. This will then open up the data item expression window.
6. Enter the following into the Expression Definition window.

```
filter( members ( [great_outdoor_sales_en].[Years].[Years] ),  
count(1 within set children(  
currentMember( [great_outdoor_sales_en].[Years].[Years] ) ) ) = 0 )
```

The parts of this expression are explained in more detail in the section below titled “Explaining the Expression”. The portion `[great_outdoor_sales_en].[Years].[Years]` is the Years hierarchy from the Years dimension.

The Expression Definition dialog appears as shown below in Illustration 4.

Illustration 4: The data item expression editor with the previously mentioned expression



7. Click **OK**.

Your crosstab report should appear as shown below in Illustration 5 where the columns are Revenue and rows are the Leaf Members Query Calculation.

Illustration 5: Report Studio completed crosstab

	<#Revenue#>
<#Leaf Members#>	<#1234#>
<#Leaf Members#>	<#1234#>

8. Run the report.

In this example the report returns all the months in the cube. This is correct, months are the leaf levels. In this particular cube we could have achieved the same result by just using the Months level, but for demonstration purposes, you can see the expression works.

Illustration 6: IBM Cognos Viewer showing report output displaying all months in the cube



	Revenue
2004/Jan	72,741,622.65
2004/Feb	73,242,843.99
2004/Mar	75,720,238.67
2004/Apr	65,278,358.76
2004/May	74,695,218.83
2004/Jun	82,169,806.98
2004/Jul	80,723,457.30
2004/Aug	80,174,484.77
2004/Sep	74,892,374.18
2004/Oct	75,086,688.05
2004/Nov	75,470,246.45
2004/Dec	84,197,463.09
2005/Jan	94,996,202.61
2005/Feb	99,488,042.66

We shall demonstrate the technique again on the example unbalanced hierarchy shown earlier. This was the scenario where the Profit member drills down to a Sales and a Costs member and Sales has no children, but Costs does. Costs also has Fixed and Variable members below it.

For this example, the expression changes to use the Chart of Accounts hierarchy.

```
filter( members ( [COA Cube].[Chart Of Accounts].[Chart Of Accounts] ),
count(1 within set children( currentMember(
[COA Cube].[Chart Of Accounts].[Chart Of Accounts] ) ) ) = 0 )
```

When the report is run against this cube, the result returns the Sales, Fixed, and Variable, as seen in Illustration 7. These are all leaf members for the hierarchy.

Illustration 7: IBM Cognos Viewer window showing report output for all leaf members of the Chart Of Accounts hierarchy

IBM Cognos Viewer

	measure
Sales	100
Fixed	100
Variable	100

For the expression to work in your own cubes, simply replace the hierarchy within the expression.

Explaining the Expression

How does the expression work? Again, the expression is as follows.

```
filter( members ( [COA Cube].[Chart Of Accounts].[Chart Of Accounts] ),
        count(1 within set children( currentMember(
            [COA Cube].[Chart Of Accounts].[Chart Of Accounts] ) ) ) = 0 )
```

We only want members from the hierarchy who have a children count of zero.

1. First we obtain all the members of the hierarchy:

```
members ( [COA Cube].[Chart Of Accounts].[Chart Of Accounts] )
```

2. We then use the currentMember function to iterate through all the members:

```
currentMember( [COA Cube].[Chart Of Accounts].[Chart Of Accounts] )
```

3. Within that iteration we get the children of that member:

```
children( currentMember( [COA Cube].[Chart Of Accounts].[Chart Of Accounts] ) )
```

4. We count the number of children:

```
count(1 within set children( currentMember(
    [COA Cube].[Chart Of Accounts].[Chart Of Accounts] ) ) )
```

5. We then filter the set of all members to those that have a child count of 0. This is the final expression:

```
filter( members ( [COA Cube].[Chart Of Accounts].[Chart Of Accounts] ),
        count(1 within set children( currentMember(
            [COA Cube].[Chart Of Accounts].[Chart Of Accounts] ) ) ) = 0 )
```

About the author

Marc Reed

Marc Reed is a Principal Consultant working within the UK Cognos BI team. He has been working with Cognos BI software for over ten years. Within that time he has helped many customers (in many sectors) on a range of projects ranging from trouble shooting to large enterprise deployments. Marc has developed many best practices over the years and has shared these with many BI developers and consultants.

© Copyright IBM Corporation 2012

(www.ibm.com/legal/copytrade.shtml)

[Trademarks](#)

(www.ibm.com/developerworks/ibm/trademarks/)