BPS 1 BIRT Cross tab, OLAP Cube Support Specification

Author: Rima Kanguri

Document Revisions

Version	Date	Description of Changes
Draft 1	10/27/2006	Initial draft.
Draft 2	11/06/2006	Revised document
Draft3	11/29/2006	Added section 9 on chart from David M
Draft4	11/29/2006	Revised document with latest mockups
Draft5	01/18/2007	Revised document based on review comments

1. Introduction

This document covers details on BIRT support for OLAP and Cross tab Report Item. In BIRT 2.2 we plan to support the following

- Cross tab Report Item- A cross tabulation report item (often abbreviated as a Cross tab report item) that displays the relation between two variables in a matrix format.

- BIRT OLAP Cube – A multi dimensional cube. A cube is based on one or more BIRT data set.

2. Overview

2.1 OLAP Cube model

A new report element, it stores the metadata of the OLAP cube like dimensions, measures, hierarchies etc.

Note: In future it could be extended to support external cubes.

2.2 Cross Tab Report Item

A new report item which displays data in matrix format. It is based on the OLAP cube.

2.3 BIRT OLAP Engine

The BIRT OLAP Engine creates materialized cube from the specified cube model. It supports an interface which allows user to query the cube view. It also supports an interface for navigating result set and retrieving data values. The BIRT OLAP engine will support

Query API - defines the query management interfaces. It defines API to perform operations like sort, filter, and aggregations and define computed columns.

Cursor API defines the interface for managing the multidimensional result sets created as a result of executing the queries. It provides interfaces to navigate result set and extract data values.

Java OLAP API is an API for applications to access the OLAP engine. JOLAP API will be supported by BIRT Design Engine and BIRT OLAP Engine:

- BIRT Design engine to support JOLAP metadata API
- BIRT Data Engine (OLAP engine) will provide limited support of JOLAP API. Currently the plan is to support cursor based result set API

2.4 Java OLAP API

JOLAP API has been developed with Java Community process; more detail on the specification is available at http://jcp.org/en/jsr/detail?id=69. OLAP database vendors can expose product capabilities using standard JOLAP API. Also OLAP reporting and analysis tools that wish to access multidimensional data can use the JOLAP API.

- The JOLAP API is categorized into following

JOLAP Metadata model API defines the interface to access cubes, dimensions, hierarchies etc.

JOLAP Query API defines the query management interfaces. It defines API to perform operations like sort, filter, and aggregations and define computed columns.

JOLAP Cursor API defines the interface for managing the multidimensional result sets created as a result of executing the queries. It provides interfaces to navigate result set and extract data values.

3. Use cases

The following section describes some of the common use cases that we plan to support.

Note: The output shown in the use cases is just a sample output, the actuate BIRT output many not be the same.

3.1 Create a Cross tab report item

User wants to create a cross tab report item that displays products sold in different regions for each product line. (Sample output below in Figure 1)

		Vintage Cars		Classic Cars		Trucks Buses		Grand Total	
		Sale	Profit	Sales	Profit	Sales	Profit	Sales	Profit
USA	CA	50	100	200	400	10	20	260	520
	CT	10	20	100	200	10	20	120	240
	MA	40	80	100	200	10	10	150	290
Australia	Queensland	90	180	50	100	10	20	150	300
	Victoria	10	20	50	100	10	10	70	130
Grand total		200	400	500	1000	50	80	750	1480

Figure 1 – Sample output

User performs the following tasks

3.1.1 Adds a new cross tab report item to the layout from the palette.

- Clicks on "Define data Cube"

3.1.2 Defines a Data Cube

- Selects the dataset for the cube.
- Defines a new dimension CustomerRegion with the following hierarchy

Country -> State

- Defines a new dimension Products with the following hierarchy

Product Line -> Products

3.1.3 Defines the Cross tab Report Item

- Adds the dimension "ProductLine" to be shown in the row area

- Adds the dimension "CustomerRegion" to be shown in the column area

- Adds the measure "Sales" and "Profit" from the measures list to be shown the detail cell area.

3.2 Create a OLAP cube model

User wants to create a cube with three dimensions CustomersHierachy, Products, OrderDate. He decides to define the following measures Sales Amount, Profit.

User performs the following tasks

- In the data explorer view, user defines a new cube report element. Cube builder dialog is invoked.

- Selects the dataset (fact table) for the cube.

- Defines a new dimension CustomerRegion with the following hierarchy

Country -> State

- Defines a new dimension Products with the following hierarchy

Product Line -> Products

- Defines a new dimension OrderDate with the following hierarchy

Year->Quarter

- Defines the following measures

  o Sales Amount (Selects function = sum (SalesAmount))

  o Profit (Selects function = sum(Profit))

3.3 Create a Cross tab item with column area showing date field categorized by Year, Quarter

User wants to create a cross tab report item that displays the sales of all products categorized by year and quarter along the column area, and categorized by country and state along row area.

| | | 2005 | | | | 2006 | |
| | | Q1 | | Q2 | | Q1 | |
		Sale	Profit	Sale	Profit	Sale	Profit
Australia	Queensland	00.00	00.00	00.00	00.00	00.00	00.00
	Victoria	00.00	00.00	00.00	00.00	00.00	00.00
	Total	00.00	00.00	00.00	00.00	00.00	00.00
USA	CA	00.00	00.00	00.00	00.00	00.00	00.00
	CT	00.00	00.00	00.00	00.00	00.00	00.00
	Total	00.00	00.00	00.00	00.00	00.00	00.00
	Grand Total	00.00	00.00	00.00	00.00	00.00	00.00

Figure 2 – Sample output of Order Date by Customer Region

User performs the following tasks

- Adds a new cross tab report item. .
- Selects the cube created in case (3.2)
- Adds the dimension "Order Date" to be shown in the row area.
- Adds the dimension "CustomerRegtion" to be shown in the column area of the cross tab.
- Adds the measure "SalesAmount" and "Profit" from the list to be shown in the detail area. (User can choose multiple fields)

3.4 Apply sort and filter on a Cross tab item

User wants to further modify the cross tab in use case 3.1 (Figure 1) to show only product lines "Classic cars" and "Vintage cars". Also user would like to sort the field "Country" in row area in descending order

User performs the following tasks

- Selects the cross tab item

- Select the filter tab in the property editor (similar to table) and enters the filter expression
- Selects the sort tab in the property editor and specifies the sort field="Country" and sort order.

3.5 Define subtotal and grand total on a Cross tab item.

User wants to show subtotal at the state level and show column and row grand total

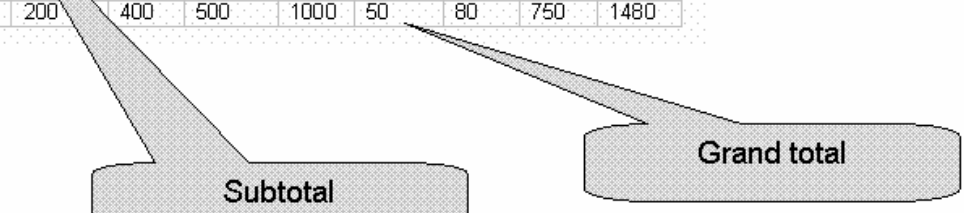|		Vintage Cars		Classic Cars		Trucks Buses		Grand Total	
		Sale	Profit	Sales	Profit	Sales	Profit	Sales	Profit
USA	CA	50	100	200	400	10	20	260	520
	CT	10	20	100	200	10	20	120	240
	MA	40	80	100	200	10	10	150	290
	Total	100	200	400	800	30	50	530	1050
Australia	Queensland	90	180	50	100	10	20	150	300
	Victoria	1	20	50	100	10	10	70	130
	Total	100	200	100	200	20	30	220	430
Grand total		200	400	500	1000	50	80	750	1480

Subtotal

Grand total

Figure 3 – Sample output of Product line by Customer Region with subtotals

User performs the following tasks

- Selects the cross tab item
- Selects the dimension "CustomerRegion" in row area
- Selects the levels at which the aggregations are to be defined. In this case it is state level
- Selects the option to show column and row grand total

3.6 Create a cross tab item with multiple dimension stacked in row or column area

			Vintage Cars		Classic Cars		Trucks Buses		Grand Total	
			Sale	Profit	Sales	Profit	Sales	Profit	Sales	Profit
2006	USA	CA	50	100	200	400	10	20	260	520
		CT	10	20	100	200	10	20	120	240
		MA	40	80	100	200	10	10	150	290
		Total	100	200	400	800	30	50	530	1050
	Australia	Queensland	90	180	50	100	10	20	150	300
		Victoria	10	20	50	100	10	10	70	130
		Total	100	200	100	200	20	30	220	430
2005	USA	CA	100	200	400	800	20	40	520	1040
		CT	10	20	100	200	10	20	120	240
		MA	40	80	100	200	10	10	150	290
		Total	100	200	400	800	30	50	530	1050
	Australia	Queensland	100	200	50	100	50	100	200	400
		Victoria	10	20	50	100	10	10	70	130
		Total	100	200	100	200	20	30	220	430

Figure 4 – Sample output with multiple dimensions stacked in row area

User performs the following tasks

- Selects the cross tab defined in user case 3.1

- Adds another dimension "Order Date" to be shown in the row area along with dimension "CustomerRegion".

3.7 Create a Cross tab item with fixed interval

User has to create a cross tab report item that shows sales of products in 5 year interval

3.7.1 Creates a cube

- Selects the dataset,

- Defines the dimension order date and adds the level order date. User selects the interval = "Year" and enters range = 5

- Defines the dimension product and adds the level productName

3.7.2 Creates a cross tab item

Creates a cross tab item based on the above cube

3.8 Cross tab item with fixed number of rows

A static cross tab is one in which the number of columns or rows are fixed.

User has to create a cross tab report item that displays the patients admitted in a hospital categorized in different age range.

Reporting Year ☑	2001	2002	
AgeGroup			
AgeGroup	58736	60781	
Age below 12	9084	8786	
Age between 13 and 30	10097	9721	
Age between 30 and 60	20996	21993	
Age above 60	18559	20281	

Figure 5 – Sample cross tab with fixed number of rows

User performs the following tasks

3.8.1 Defines the cube

- Specifies the cube name and select the dataset (fact table) for the cube.
- Defines a new dimension PatientAge.
- Adds the following level "Age" with level type = static

  The age level has the following list of labels and conditions

  "Age below 13 = Age <= 12

  "Age between 12 and 30" = Age >12 and Age<=30

  "Age beween 31 and 60" = Age >30 and <= 60

  "Age 60 and above" = Age >60
- Defines a new dimension Year
  - Adds the field "AdmitDate"
- Defines the following measure
  - Selects function = Count

3.8.2 Creates a cross tab item

Similar to steps in 3.5.2

3.9 Create a Cross tab with multiple attributes for a field.

User has to modify the cross tab in figure 1, to show country capital along with the country name. i.e he want to define multiple attributes for a level.

- Edits the cube in use case 3.2.
- Selects the dimension CustomerRegion.
- Edits the level "Country", adds the following attribute "Capital" to the level.
- Edits the cross tab layout to show the attribute cross tab cell.

4. Cube Report Element

BIRT model will support a new report element called cube. BIRT DE API will be enhanced to support API to create and manipulate a cube model.

☞   *Note: The following section could be referred for terminologies too*

Cube Report Element -.

Following are some of the characteristics of a cube.

- A cube is a collection of dimensions and measures.
- Supports start schema and snow flask schema
- A cube refers to a fact data set and several dimension dataset. The dimension dataset are optional.
- A cube is uniquely identified by name.
- Cube supports defining computed measures.
- Supports defining access control on the cube and dimensions. This will allow role based access to the cubes and dimensions.
- Filter condition list can be specified for every cube dataset.
- Limit on the number of dimensions that can be defined by cube.

4.1 Dimension

A dimension is a set of data fields arranged in hierarchies and levels. Example of a dimension is Customer Region; it could have the following fields arranged in a hierarchy Country -> State -> City. Following are some of the characteristics of a dimension.

- A cube can contain one or more measure dimensions.

- Date is a special type of dimension.

- It contains one or more hierarchy (In BIRT 2.2 UI we plan to support single hierarchy)

4.2 Hierarchy

JOLAP defines two types of hierarchy

- Level based hierarchy – It is used to define hierarchy between the levels in dimension

- Value based hierarchy – Defines hierarchical orderings of members in which; the topological structure of hierarchy conveys meaning. Such hierarchies are generally used to model situations where members are classified or ranked according to their distance from common root. A very good example is organization chart.

☞ *In BIRT 2.2 we plan to support only the LevelBaseHierarchy*

4.3 Level

A level identifies a dimension member. Following are some of the characteristics of a level.

- Name to identify the level.

- levelType (static and dynamic).

- Example of level Type = dynamic.
  User defines a dimension Region with the following 3 levels

  - Country (type = dynamic)

  - State (type = dynamic)

  - City (type = dynamic)

- Example for level Type = static
  User defines a dimension called AgeGroup with the following level

  - AgeGroup (type = static)

    The age level has following additional details i.e a list of conditions

    Label and Condition A = Age <10

    Label and Condition B = Age >=10 and < 50

    Label and Condition C = Age >50

- List of condition items, where each item contains label and condition expression. This field applies only if the level Type = dynamic- List of attributes (fields).

- Interval (Possible value are Hour, Minute, Second, Prefix, Year, Month, Quarter, Week, Day, Interval)

- Range specifies the interval range.

4.4 Measures

A measure is a special type of dimension.

- Each measure is defined by a
    - Name
    - Database fieldname or java script expression.
    - The aggregation function.
- Several measures can be grouped into a group. A cube can have several measure groups.
- In measure the following aggregation functions are supported
    - SUM
    - COUNT
    - AVERAGE
    - WEIGHTEDAVG
    - MAX
    - MIN
    - STDDEV
    - FIRST
    - LAST
    - MEAN
    - MODE

5. Cross tab Report Item

BIRT will support a new report item called cross tab.

Following are some of the characteristics of a cross tab item.

- A cross tab report item refers to a cube
- Supports a list of filters. Each filter contains filter condition. A filter is specified for a level.
- Supports sort on levels
- Supports highlight rules
- Supports defining subtotals on different levels of the cross tab column and row area. Each subtotal can have a different aggregation function.
- Supports defining grand total.
- Supports style for cross tab item, row headers, column headers, aggregation cells, detail cell, header cell.

- A cross tab cell supports all the report items that table cell supports (except for list and table report item)
- Column area contains list of dimensions to be shown in column area (from the cube)
- Row area contains the list of dimensions to be shown in row are (from the cube)
- Detail area contains list of measures from the cube to be shown in the detail area.
- Supports java script event handler onCreate(), onPrepare(), onRender() methods
- Supports hyperlink and drill through.
- Supports java scripts in expressions.

6. Cube Schema Builder

A new builder will be provided to define the cube schema. It allows user to

- Select the datasets

- Define dimensions

- Define measures.

Step1

- Select the primary dataset

- If user wants to specify multiple datasets, he can do so by clicking on "Add joint dataset"

- User can specify the filter by clicking on the "Filter" command button
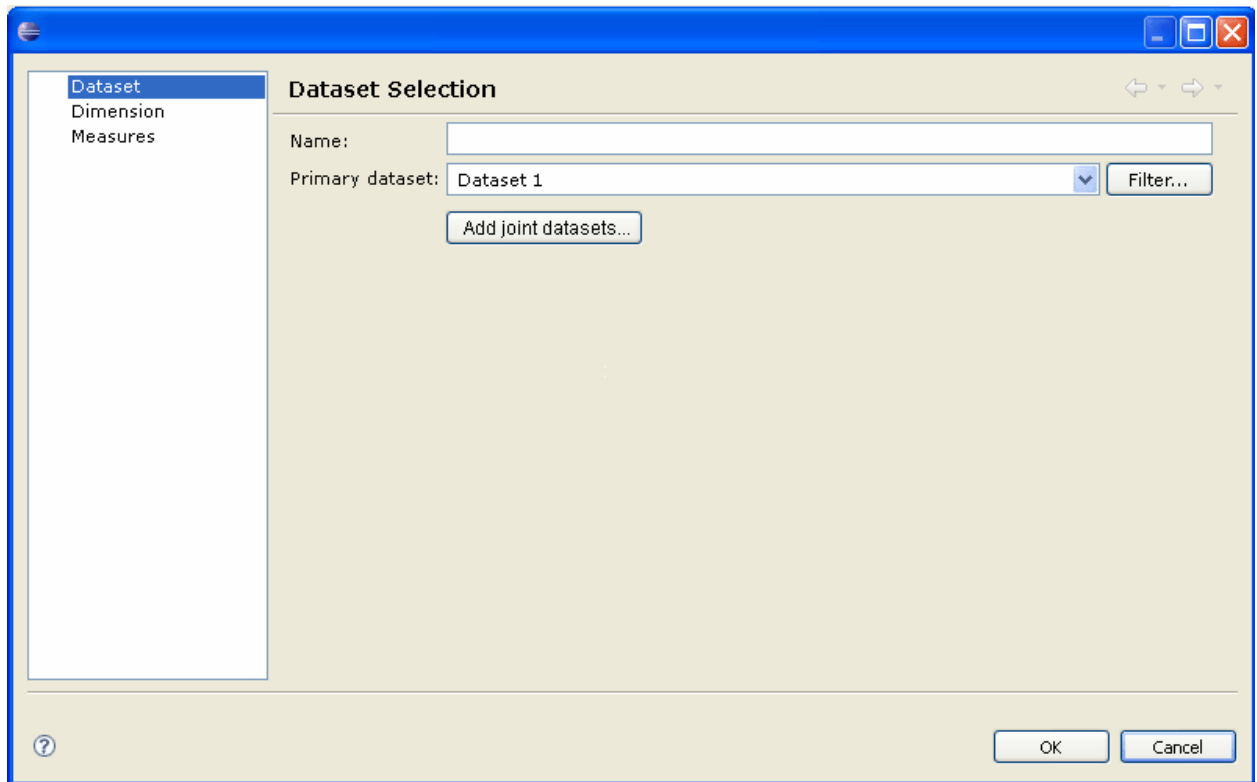
Step2 Manage dimensions.

- Click on the Add button to add a new dimension or level

- Click on the "Edit" to edit a dimension or level

- The add behavior is defined by which item is selected in the tree. If a user selects a cube and clicks "Add" button, the Add dimension dialog is shown. If a user selects a dimension or level and clicks "Add" button the add level dialog is shown.

- The properties button is used to specify advanced properties of a level. It is enabled only when a dimension level is selected.
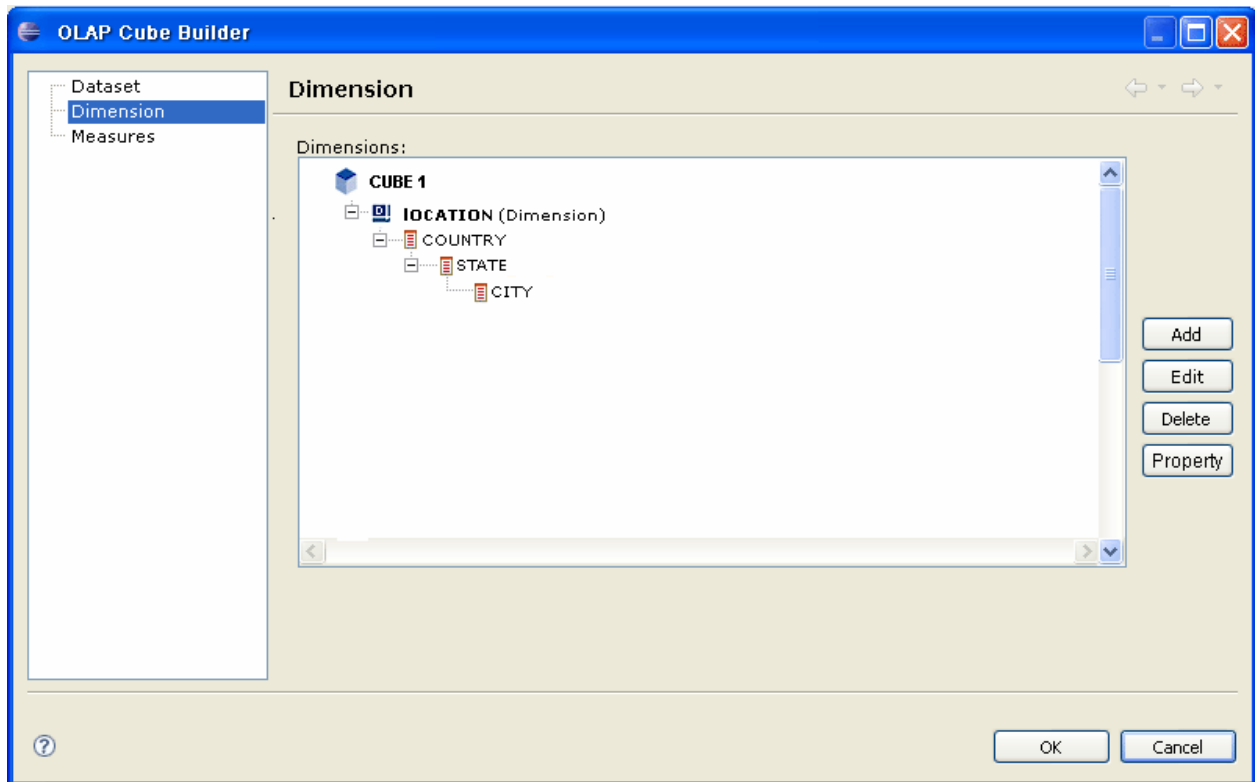
Step 2 (a) Add/Edit a dimension

- User can add fields to a dimension by DnD the field from the available list to dimension list.

- If a user drags a data type field, it is treated as a special type of dimension. The UI will automatically create a hierarchy for date field with the following subcategories Year -> Quarter -> Month -> Week ->Day . By default the Year and Quarter are checked.
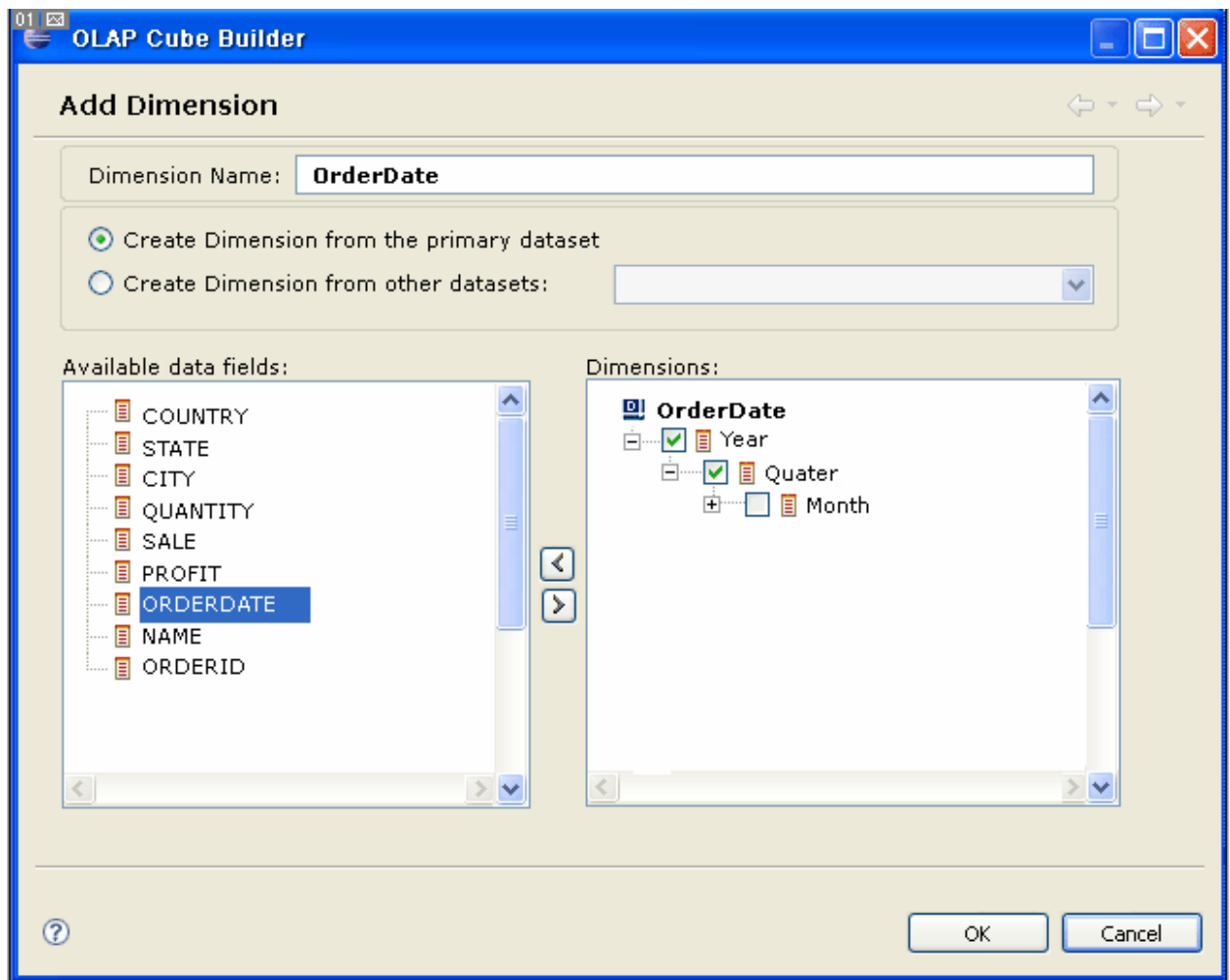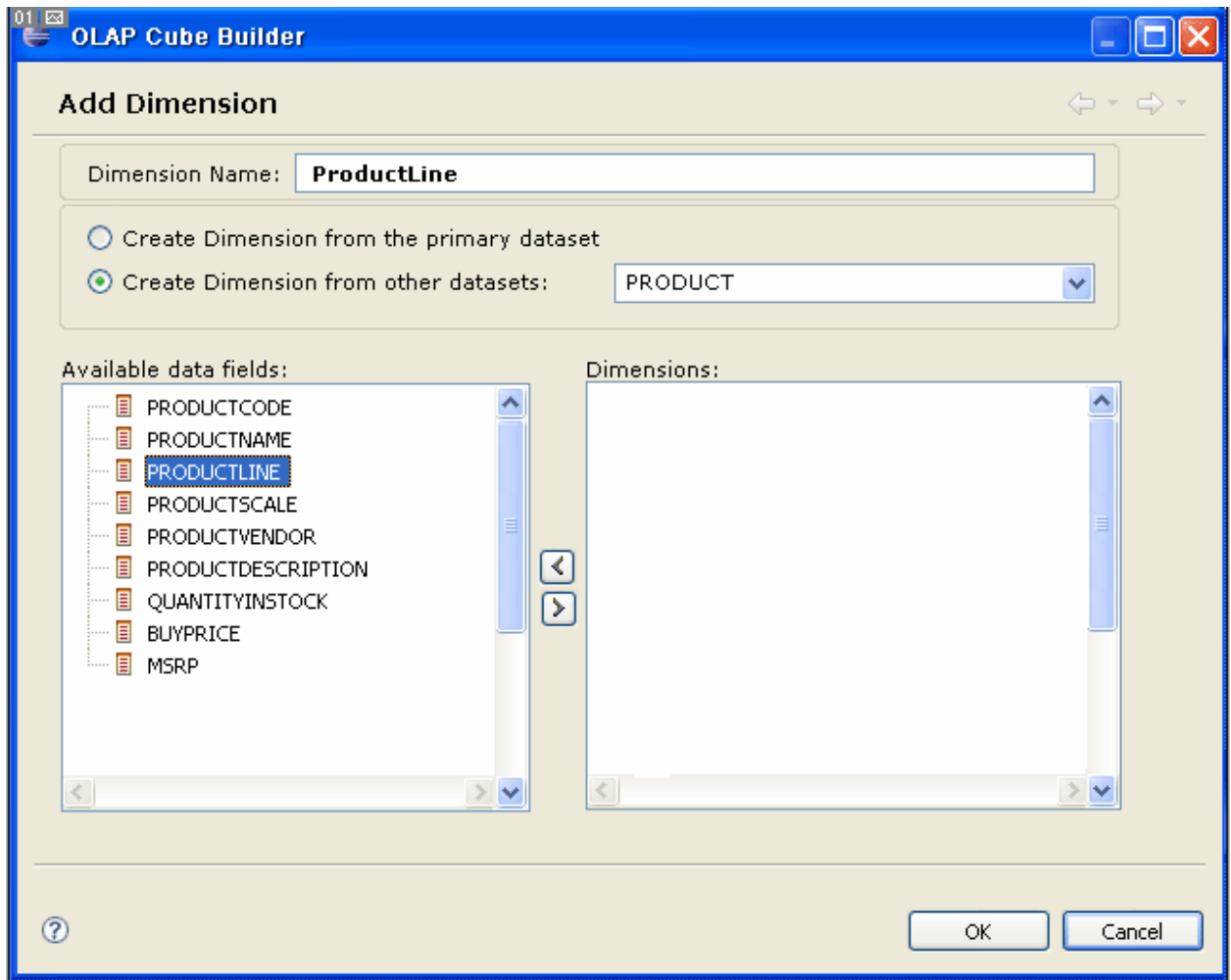
Step 2 (b) Add a level

- User can add a level to a dimension by selecting the dimension node and clicking on the "Add" buttion

- In the Add Level dialog, user can choose the name and the field name to be used for the level.
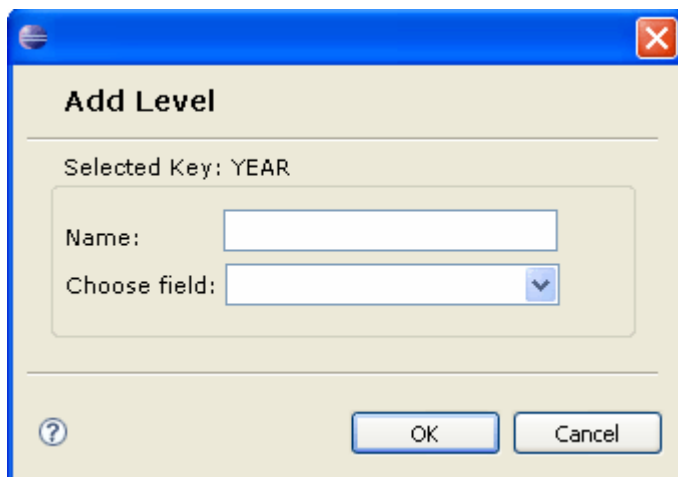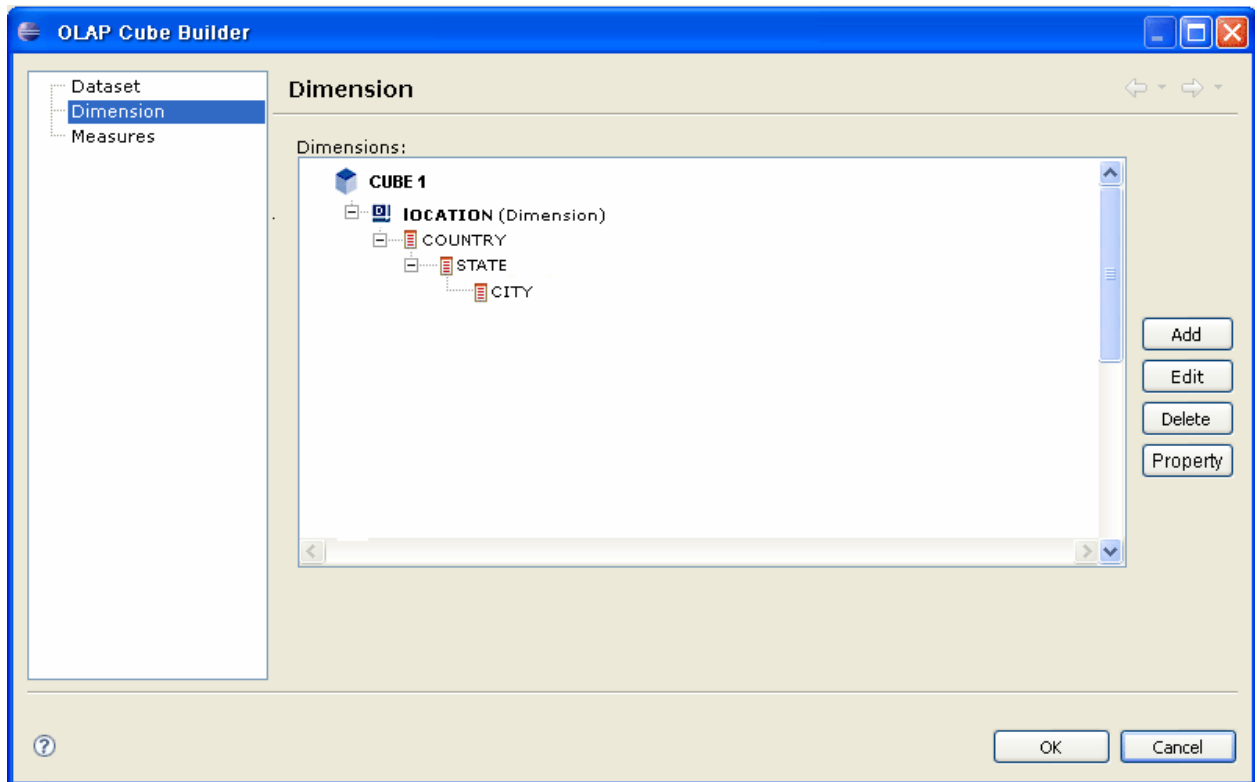
Step 2 (C) Specify level details like attributes

- User selects a level and clicks on the "Properties" button to specify the level details

Step 3 – Define one or more measures

-   A cube can have multiple measure groups

-   Each measure group can contain one or measures.

-   A measure contains the following properties name, field name and the function name
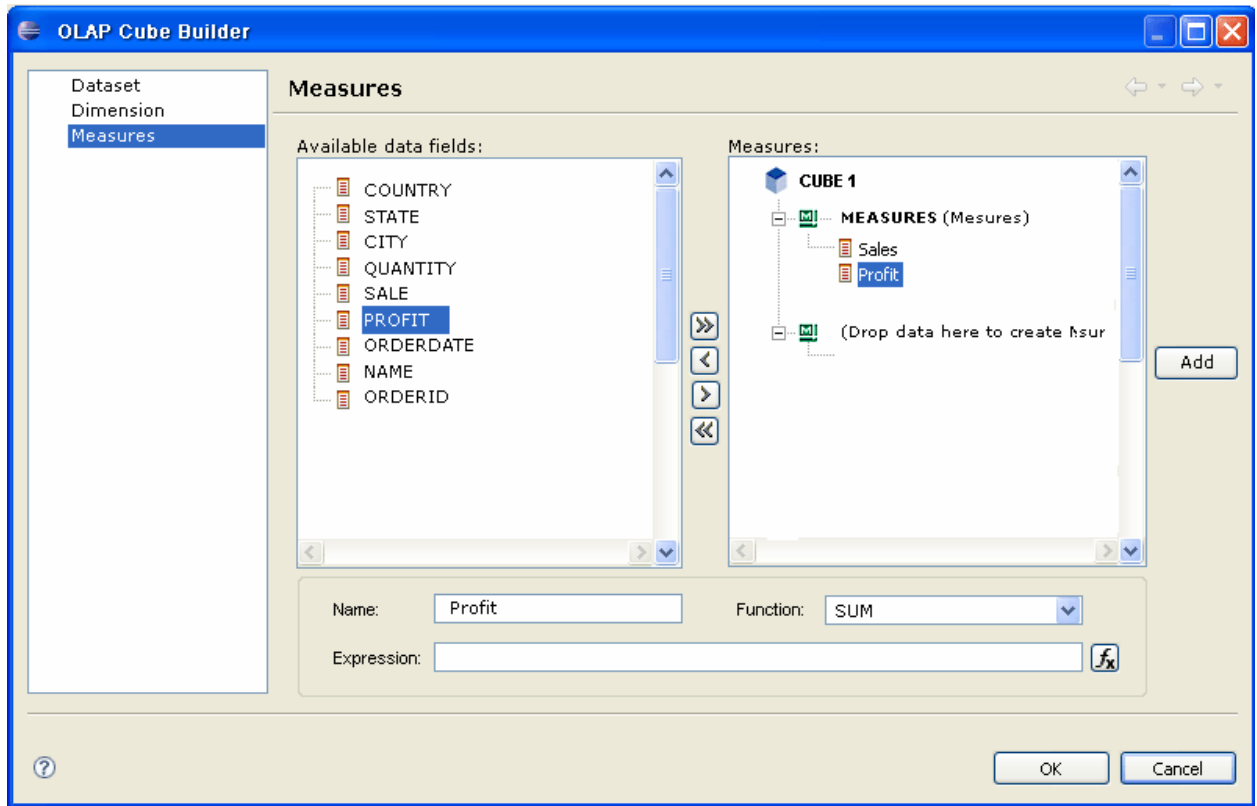
Figure 6 – Cuber builder UI to specify measures

- A new measure can be adding by clicking on the arrow button or clicking on the "Add" button.

- The field of the measure can be a dataset field or a java script expression.
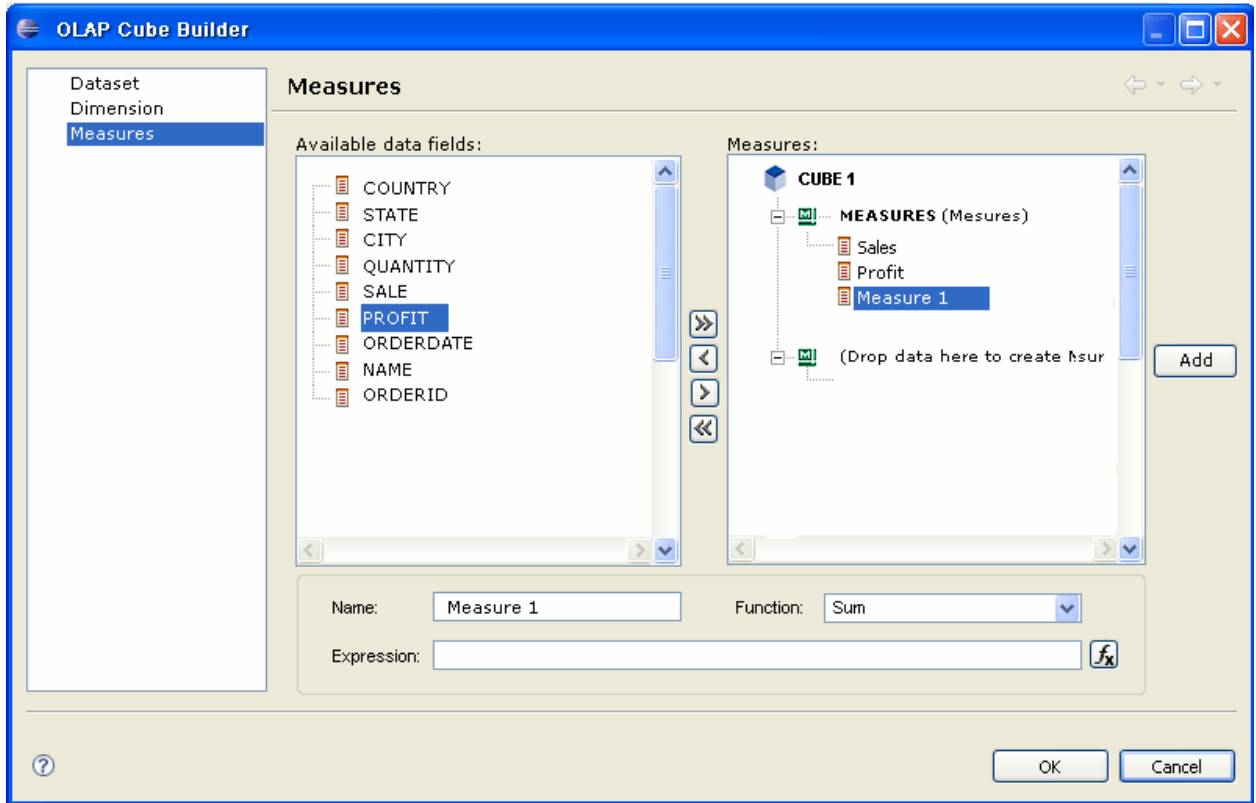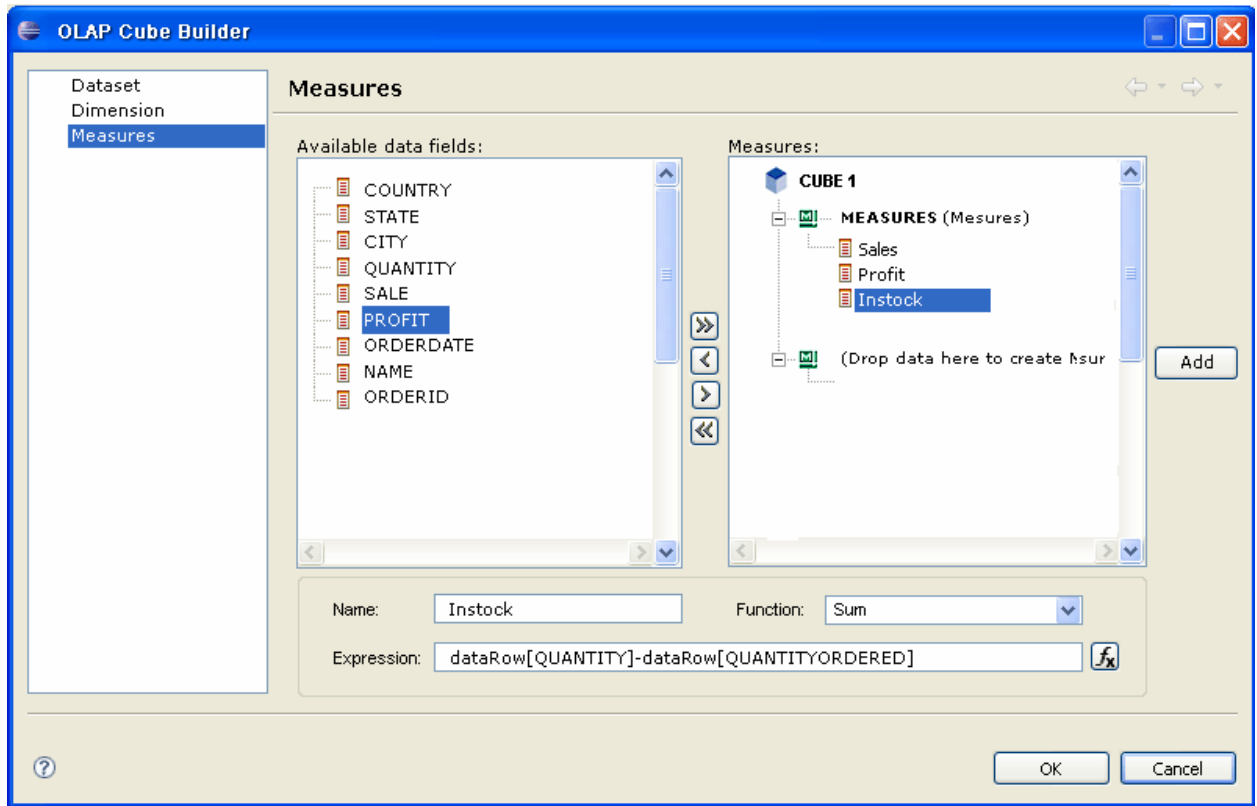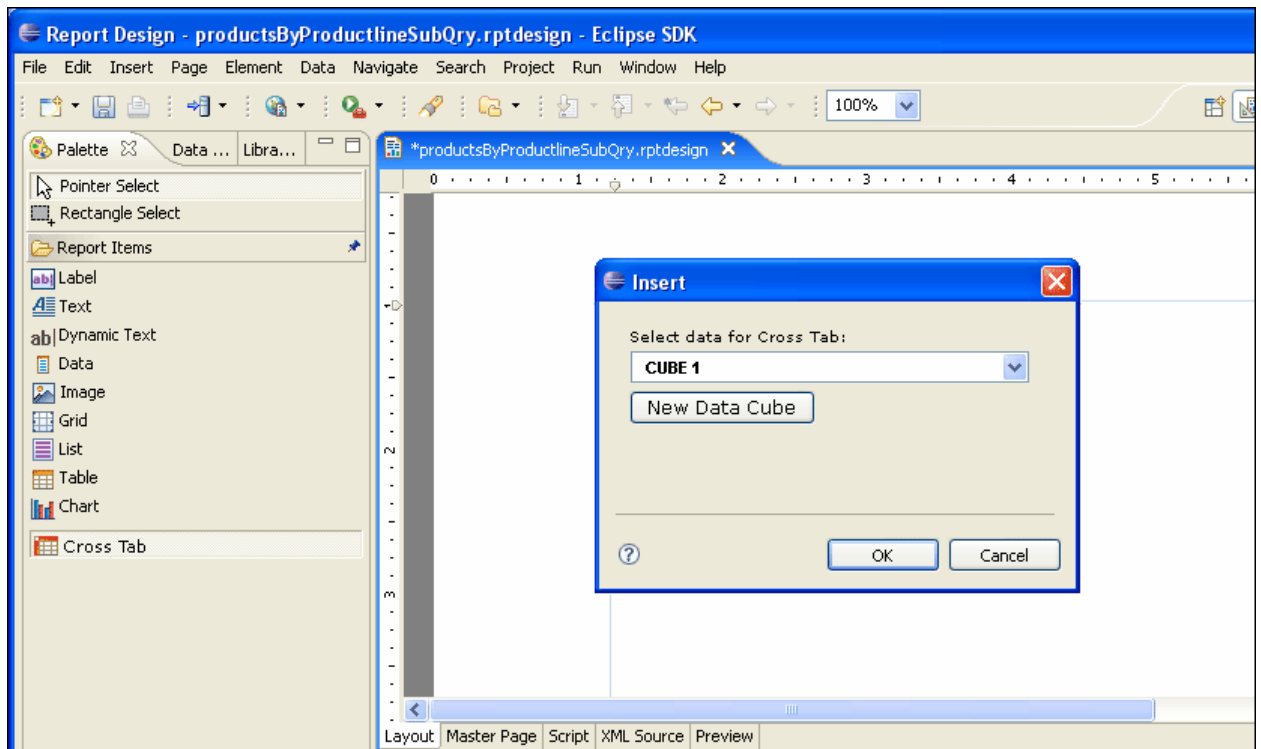
7.   Data Explorer View

Data explore view will be enhanced to show all the cube models defined in the report design. The cube node contains multiple dimensions (including measure dimension)
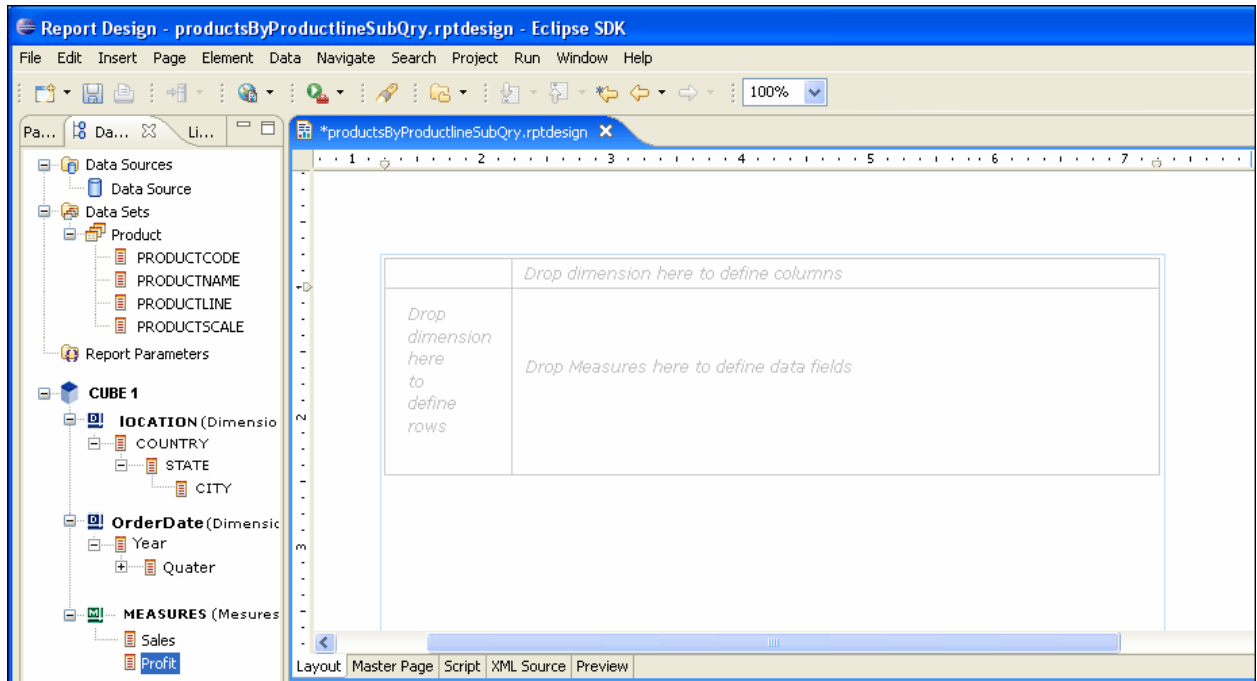
8. Creating a Cross tab Report Item

User can define a new cross tab item by dragging and dropping the cross tab report item from the palette to the layout view. The following section describes the steps involved in defining a cross tab item.

Step1 – Drag and drop a cross tab report item from palette to the layout view. Choose an existing cube schema or creates a new cube schema
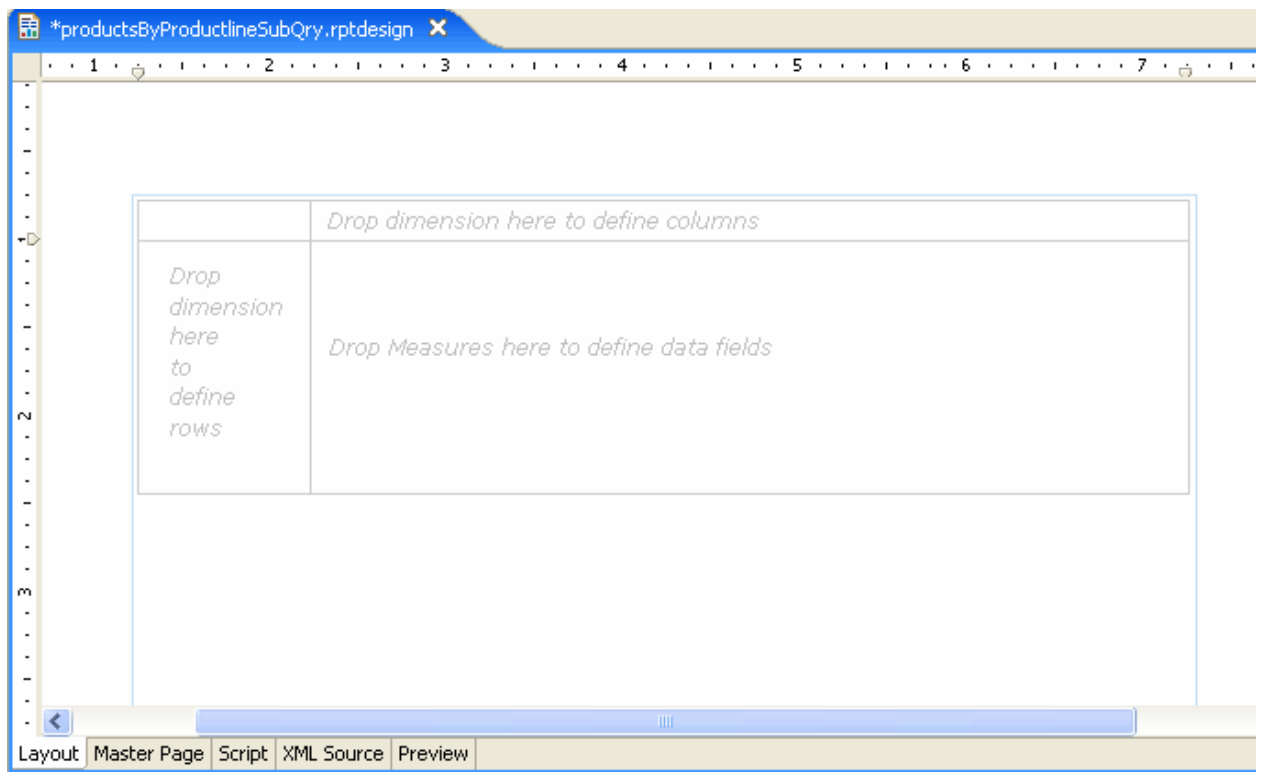
Step 2 – Select the fields to be shown in the row area

User drags and drops the dimension from the data explores view to the row area of the layout view

Note: Multiple dimensions can be stacked in row area

Step3 – Select the fields to be shown in the column area

User drags and drops the dimensions from the data explore view to column area of layout view. If there are multiple levels, only the topmost level is visible by default. User can show other levels by clicking on show/hide level menu.

Note: Multiple dimensions can be stacked in column area.
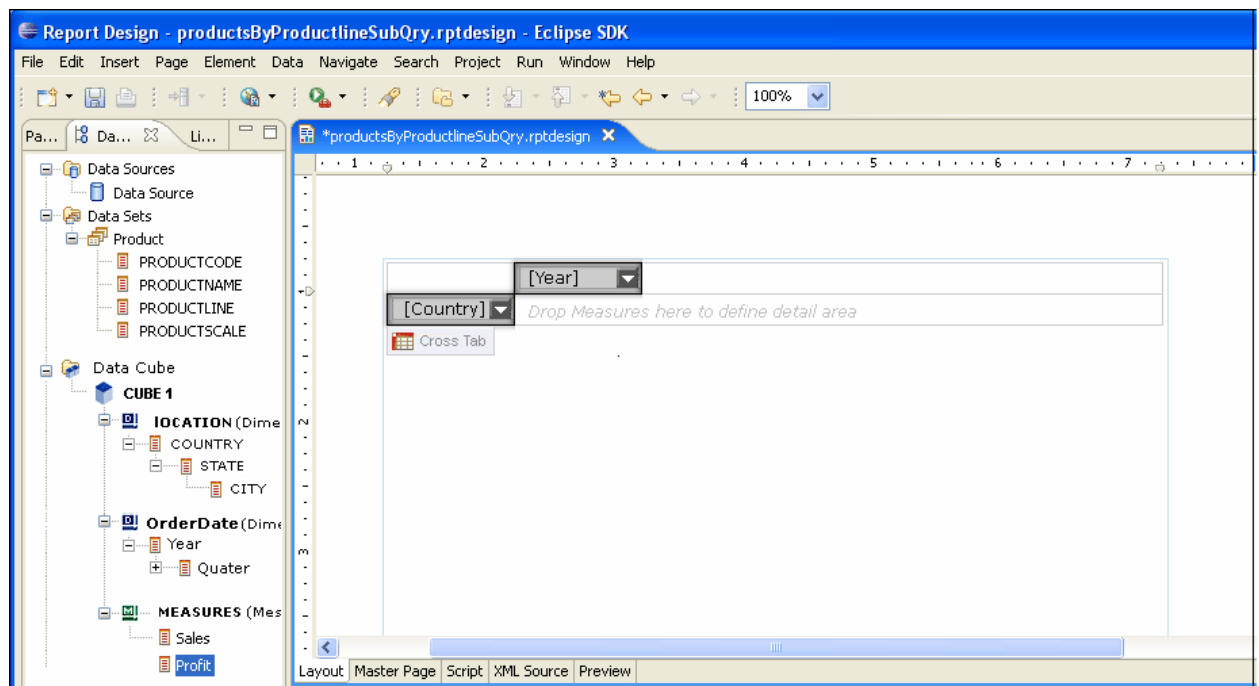
Step 4 – Selects the measures to be shown in the cross tab detail area

User drags and drops the measures from data explorer view to layout view, to. Multiple measures can be shown in the detail cell. For each measure a new detail cell is created. The sample layout below shows 2 measures in detail area.
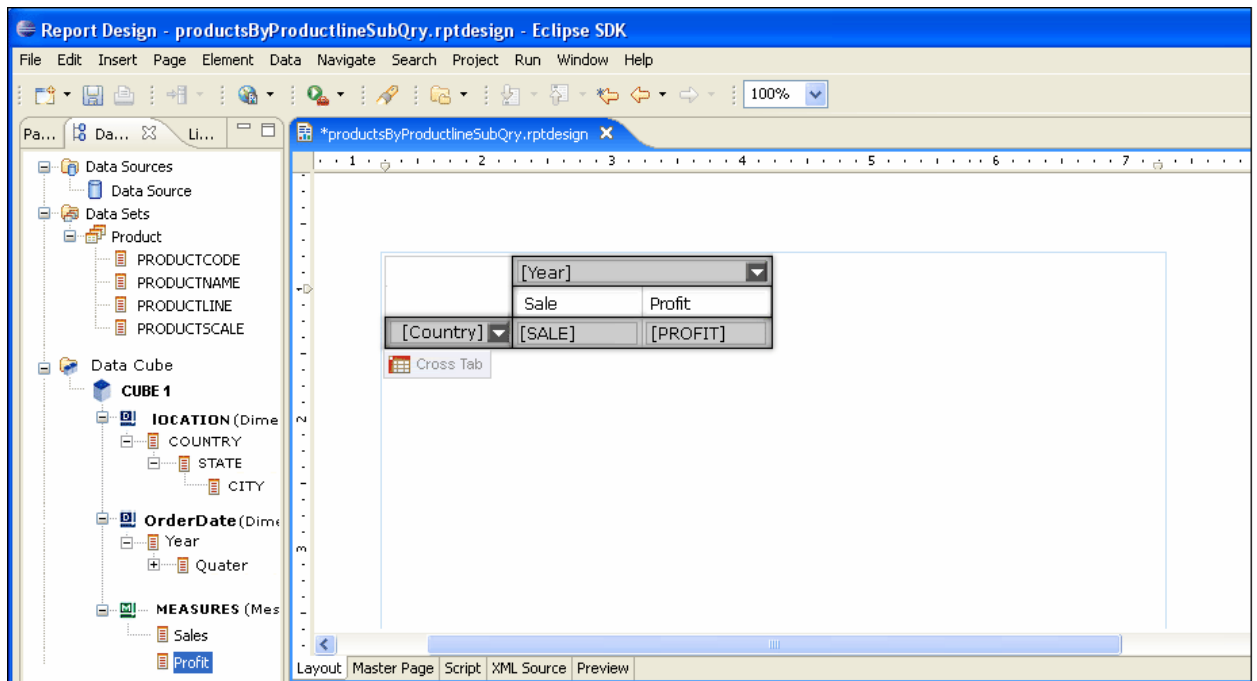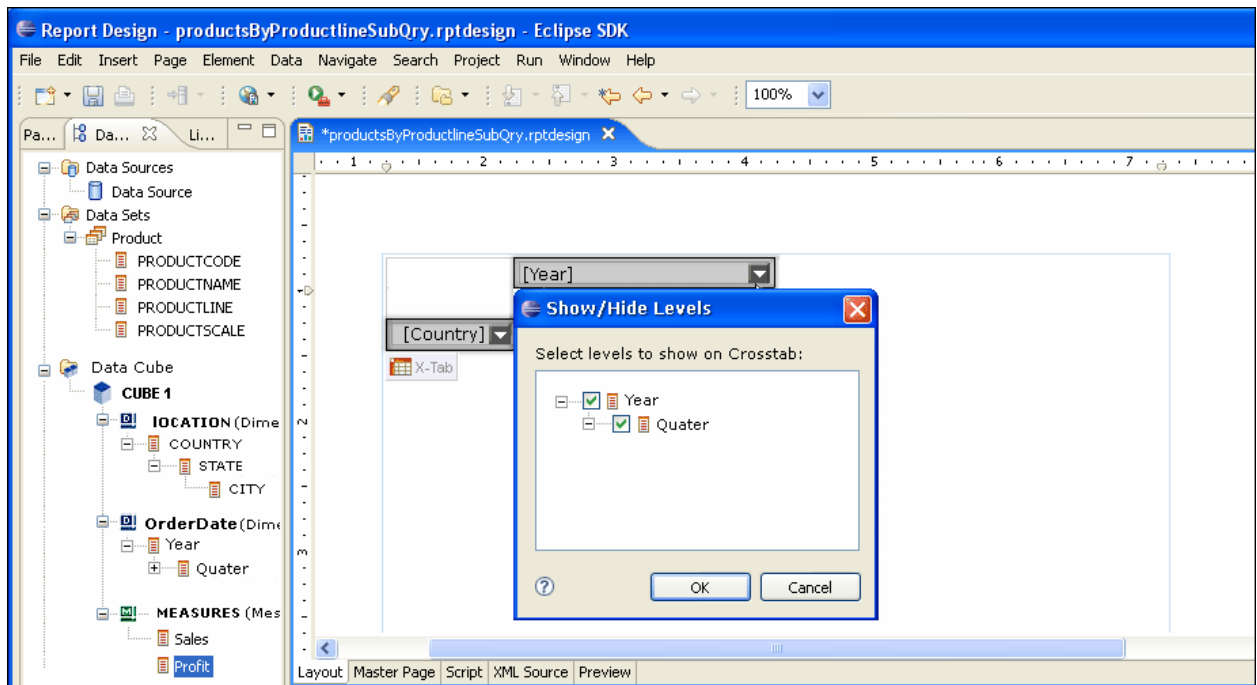
Step 5 – Pop-up menu to perform different operations on the dimension
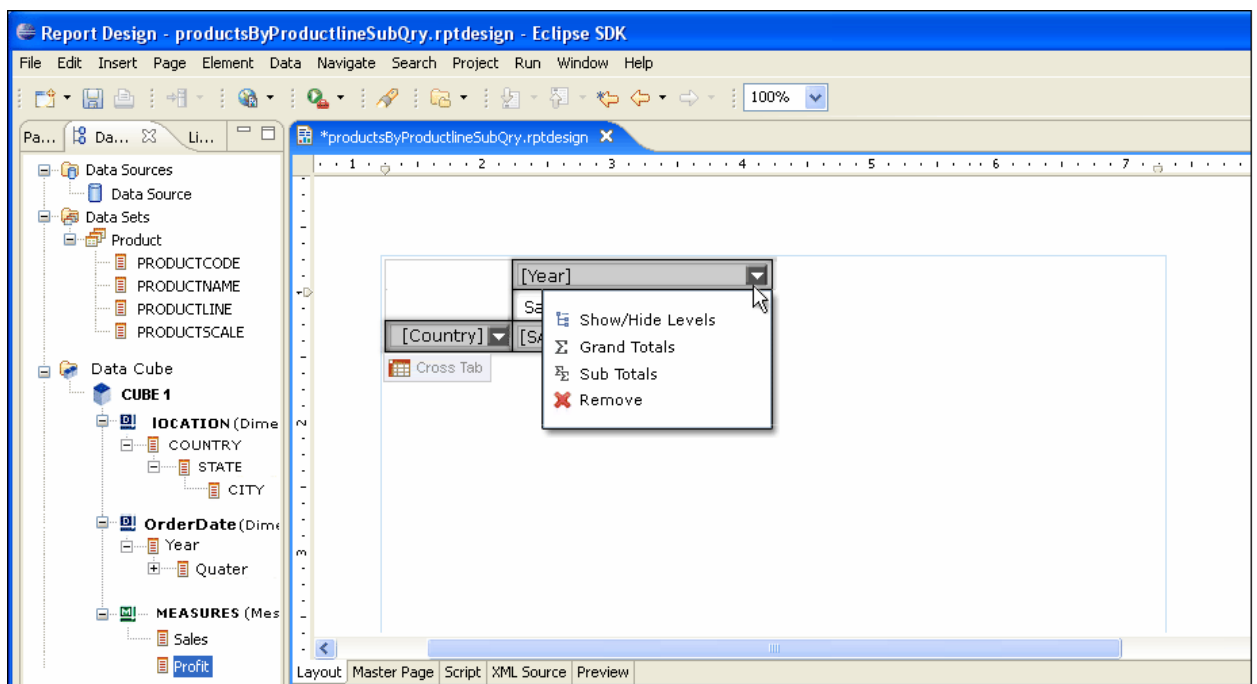
Step 5 –

- Click on "Show/Hide Levels" menu item

- Select the levels to be shown in row/column area

9. Future Plans

Following are some of the post BIRT 2.2 enhancements planned for this feature.

- Enhancements to Chart report item to support BIRT cube
- Enhancements to cube to support Snowflake schema