BPS 38 - BIRT Chart Types

Functional Specifications

Draft 6: Oct 4, 2006

Abstract

This document defines the new chart types to be added in BIRT 2.2.

Document Revisions

Draft	Date	Primary Author(s)	Description of Changes
1	2006-7-28	Zhiqiang Qian	Initial Draft
2	2006-8-1	Zhiqiang Qian	Added Difference Chart section
3	2006-8-2	Zhiqiang Qian	Rework the Mockups
4	2006-8-3	David Michonneau	Modified Difference Charts section
5	2006-9-6	Zhiqiang Qian	Added Tube Chart section
6	2006-10-04	David Michonneau	Updated Tube UI and Bubble definition

Contents

1. Introduction

1.1 Scope

Chart type consists of series types and chart attributes. To introduce new chart types, new series types will be provided, as well as new attributes.

This document will focus on the functional specifications of the new types and also list the model/API change

1.2 New Types

Four types are introduced in this document: Gannt, Bubble, Tubes and Difference charts. The Gannt and Bubble types refer to the contribution in this bugzilla entry: https://bugs.eclipse.org/bugs/show_bug.cgi?id=147770

2. Gantt Chart

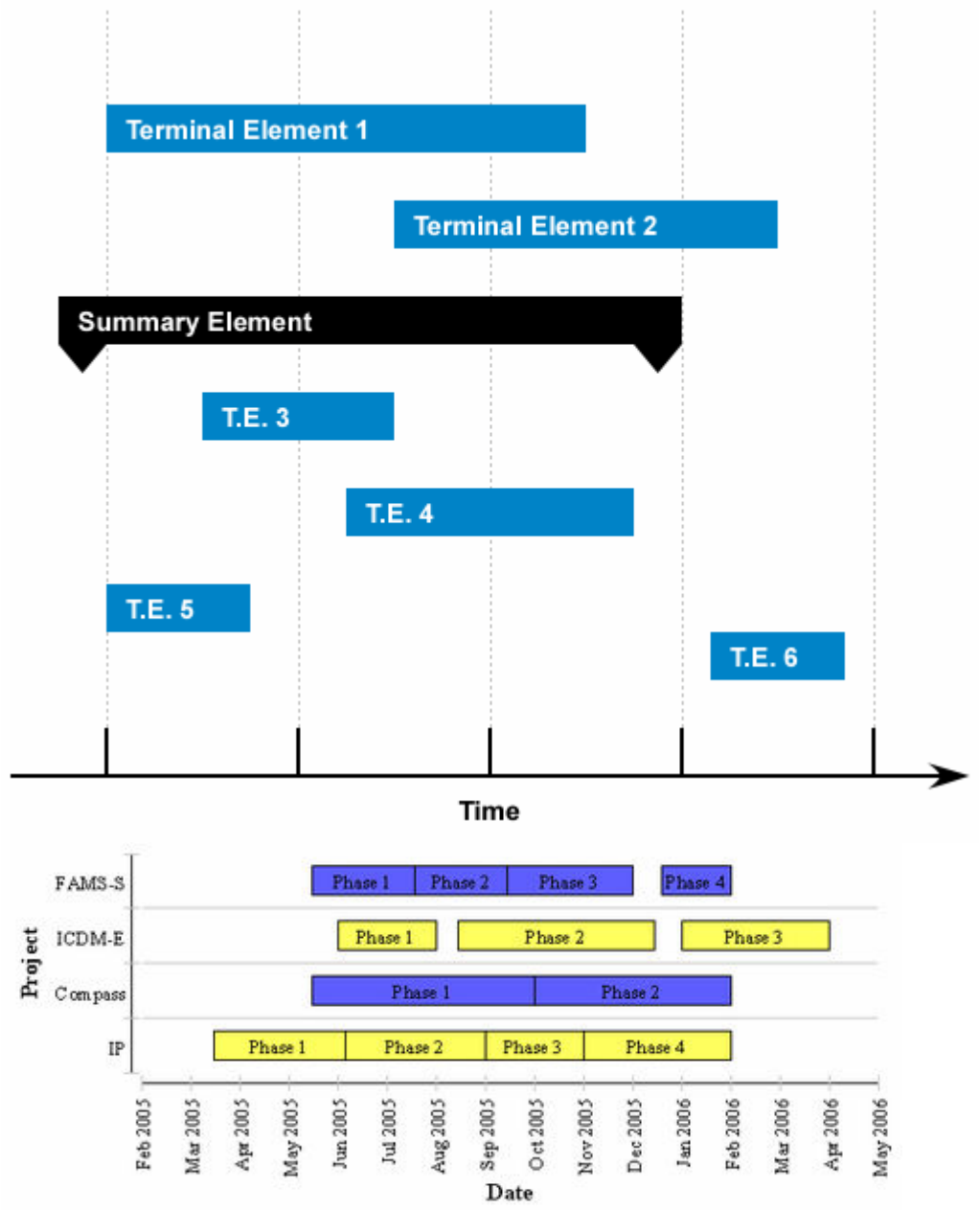
2.1 Definition

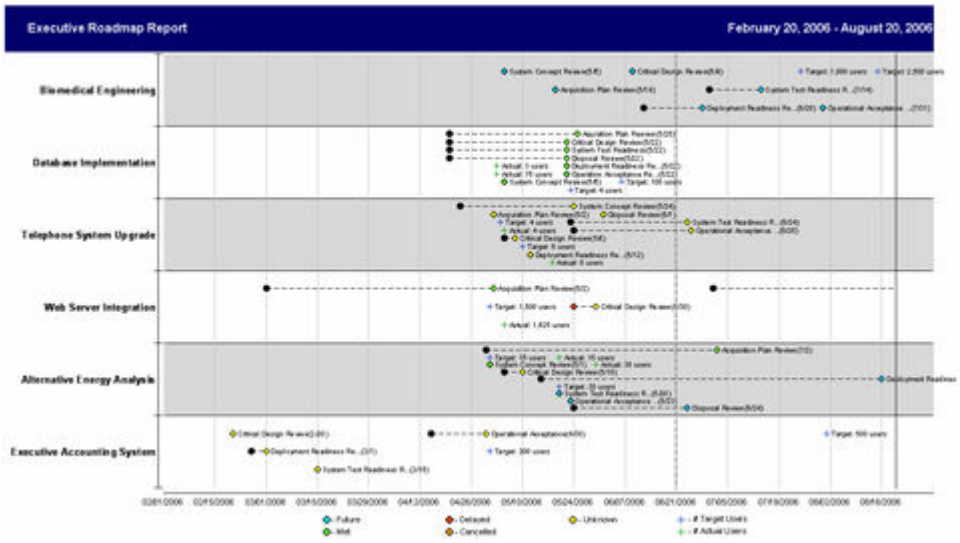
A Gantt chart is a popular type of bar chart that aims to show the timing of tasks or activities as they occur over time. Although the Gantt chart did not initially indicate the relationships between activities this has become more common in current usage as both timing and interdependencies between tasks can be identified.

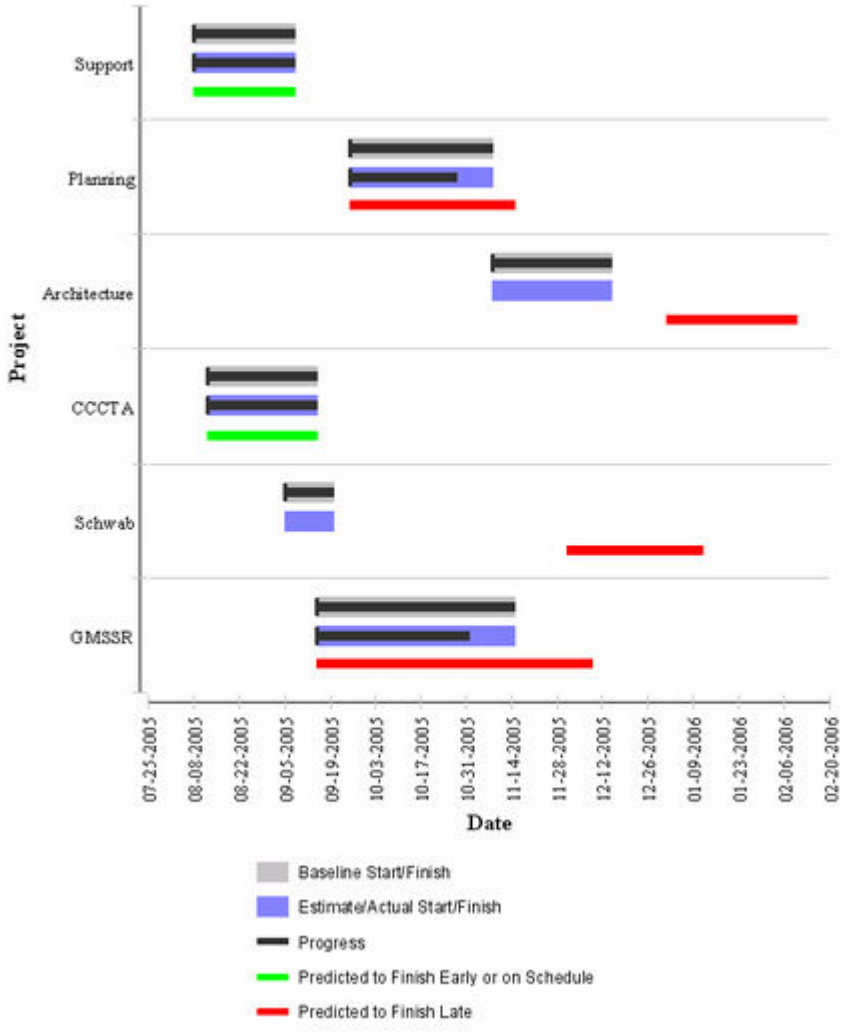
In project management, a Gantt chart can show when the project terminal elements start and finish, summary elements or terminal element dependencies. A terminal element is defined as the smallest task tracked as part of the project effort.

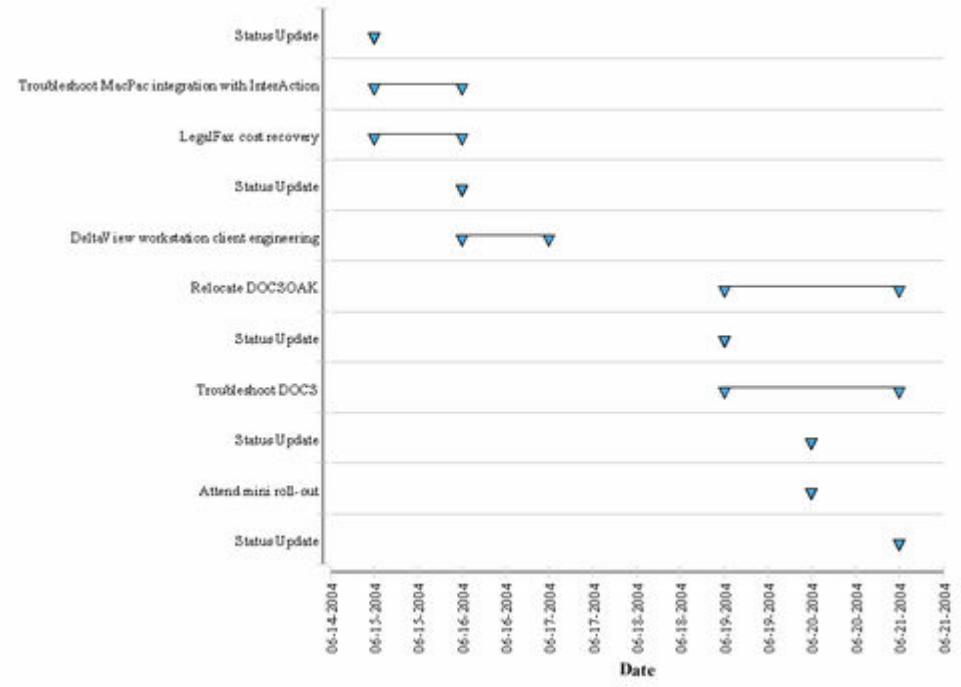
Gantt charts have become an industry standard as a key project management tool for representing the phases, tasks and activities that are scheduled as part of a project work breakdown structure or timeline of activities.

Here are some typical Gantt chart examples:


Terminal Element 1
Terminal Element 2
Summary Element
T.E. 3
T.E. 4
T.E. 5
T.E. 6
Time
Project
FAMS-S
ICDM-E
Compass
IP
Phase 1
Phase 2
Phase 3
Phase 4
Feb 2005
Mar 2005
Apr 2005
May 2005
Jun 2005
Jul 2005
Aug 2005
Sep 2005
Oct 2005
Nov 2005
Dec 2005
Jan 2006
Feb 2006
Mar 2006
Apr 2006
May 2006
Date


Executive Roadmap Report
February 20, 2006 - August 20, 2006
Biomedical Engineering
Database Implementation
Telephone System Upgrade
Web Server Integration
Alternative Energy Analysis
Executive Accounting System


Support
Planning
Architecture
CCCTA
Schwab
GMSSR
Project
07-25-2005
08-08-2005
08-22-2005
09-05-2005
09-19-2005
10-03-2005
10-17-2005
10-31-2005
11-14-2005
11-28-2005
12-12-2005
12-26-2005
01-09-2006
01-23-2006
02-06-2006
02-20-2006
Date
Baseline Start/Finish
Estimate/Actual Start/Finish
Progress
Predicted to Finish Early or on Schedule
Predicted to Finish Late



2.2 Model Change

2.2.1 Type

```
      <!--  Gantt Series      -->
      <xsd:complexType name="GanttSeries">
            <xsd:annotation>
                  <xsd:documentation xml:lang="en">This is a Series type that
holds data for Gantt Charts.
                  </xsd:documentation>
            </xsd:annotation>
            <xsd:complexContent>
                  <xsd:extension base="component:Series">
                        <xsd:sequence>
                              <xsd:element name="StartMarker"
type="attribute:Marker">
                                    <xsd:annotation>
                                          <xsd:documentation xml:lang="en">
Specifies the marker to be used for displaying the start data point on the
line in the chart.
</xsd:documentation>
                                    </xsd:annotation>
                              </xsd:element>
                              <xsd:element name="EndMarker"
type="attribute:Marker">
                                    <xsd:annotation>
                                          <xsd:documentation xml:lang="en">
Specifies the marker to be used for displaying the end data point on the line
in the chart.</xsd:documentation>
                                    </xsd:annotation>
                              </xsd:element>
                              <xsd:element name="ConnectionLine"
type="attribute:LineAttributes">
                                    <xsd:annotation>
                                          <xsd:documentation xml:lang="en">
```

```
Specifies the attributes for the line used to represent this
series.</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                                <xsd:element name="Outline"
type="attribute:LineAttributes">
                                    <xsd:annotation>
                                        <xsd:documentation xml:lang="en">
Specifies the attributes for the line used to represent this
series.</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                                <xsd:element name="OutlineFill"
type="attribute:Fill" minOccurs="0">
                                    <xsd:annotation>
                                        <xsd:documentation
xml:lang="en">Defines the extra fill to be used.
</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                                <xsd:element name="UseDecorationLabelValue"
type="xsd:boolean">
                                    <xsd:annotation>
                                        <xsd:documentation>Specifies if use
decoration value as the datapoint label</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                                <xsd:element name="DecorationLabel"
type="component:Label">
                                    <xsd:annotation>
                                        <xsd:documentation xml:lang="en">
                                        Holds the properties for the
decoration label, which could be used to decorate the primary base Axis.
                                        </xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                                <xsd:element name="DecorationLabelPosition"
type="attribute:Position">
                                    <xsd:annotation>
                                        <xsd:documentation xml:lang="en">
                                        Holds the position property for the
decoration label.
                                        </xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                                <xsd:element name="PaletteLineColor"
type="xsd:boolean">
                                    <xsd:annotation>
                                        <xsd:documentation>Indicates if use
the series palette color to draw the line instead of the color in
ConnectionLine</xsd:documentation>
                                    </xsd:annotation>
                                </xsd:element>
                            </xsd:sequence>
                        </xsd:extension>
                    </xsd:complexContent>
                </xsd:complexType>
```

GanttSeries extends form Series. Following attributes are added to support the Gantt Chart rendering.

Attribute Name	Description
StartMarker	Specifies the marker to be used for displaying the start data point on the line in the chart.
EndMarker	Specifies the marker to be used for displaying the end data point on the line in the chart.
ConnectionLine	Specifies the attributes for the line used to represent this series. This line connects the start and end markers.
Outline	Specifies the attributes for the outline used to represent this series
OutlineFill	Defines the extra fill to be used for outline. If this is not specified, Color for ConnectionLine will be used.
UseDecorationLabelValue	Specifies if use decoration value as the datapoint label, otherwise, label defined in DataComponent will be used.
DecorationLabel	Holds the properties for the decoration label, which is used to decorate the primary base Axis.
DecorationLabelPostion	Holds the position property for the decoration label.
PaletteLineColor	Indicates if use the series palette color to draw the ConnectionLine instead of the color defined in ConnectionLine

2.2.2 Data

```
      <xsd:complexType name="GanttDataSet">
            <xsd:annotation>
                  <xsd:documentation>This type holds the Gantt data associated
with a series.</xsd:documentation>
            </xsd:annotation>
            <xsd:complexContent>
                  <xsd:extension base="data:DataSet"/>
            </xsd:complexContent>
      </xsd:complexType>

GanttDataSet simply extends from DataSet, it holds the data needed for Gantt
chart rendering.
```

2.3 API Change

A new interface IAxesDecorator is introduced. This interface defines the ability to do custom axes decoration.

During computation, engine will take the decorator for consideration and pre-check the extra axes space needed. During rendering, Engine will callback the decorator to perform their custom rendering.

Currently this is mainly to resolve the problem for Gantt chart to render custom label on base axis. But this can also be used for other future types to perform other decorations for axes and the decoration is not limited to text.

```
/**
```

```
 * This interface defines a decorator renderer for Axes
 */
public interface IAxesDecorator
{

      /**
       * Returns the thickness for use with decoration.
       *
       * @param xs
       *            The display server.
       * @param ax
       *            The runtime axis model.
       * @return
       */
      double[] computeDecorationThickness( IDisplayServer xs, OneAxis
ax )  throws ChartException;

      /**
       * Perform decoration.
       *
       * @param ipr
       *            The renderer instance.
       * @param isrh
       *            Series rendering hints.
       * @param ax
       *            The runtime axis model.
       */
      void decorateAxes( IPrimitiveRenderer ipr, ISeriesRenderingHints
isrh, OneAxis ax ) throws ChartException;
}
```

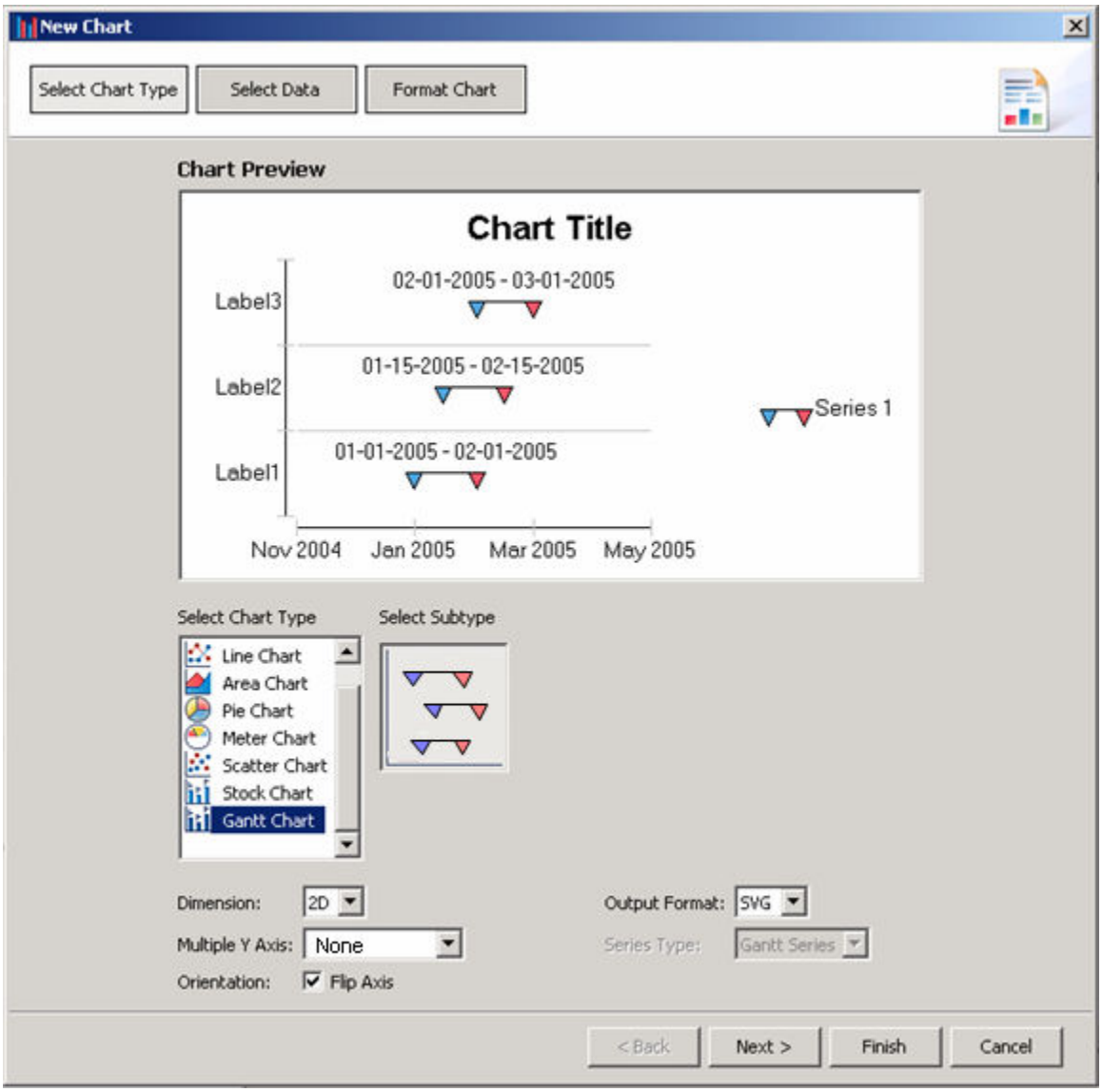
AxesRenderer will have one new method:

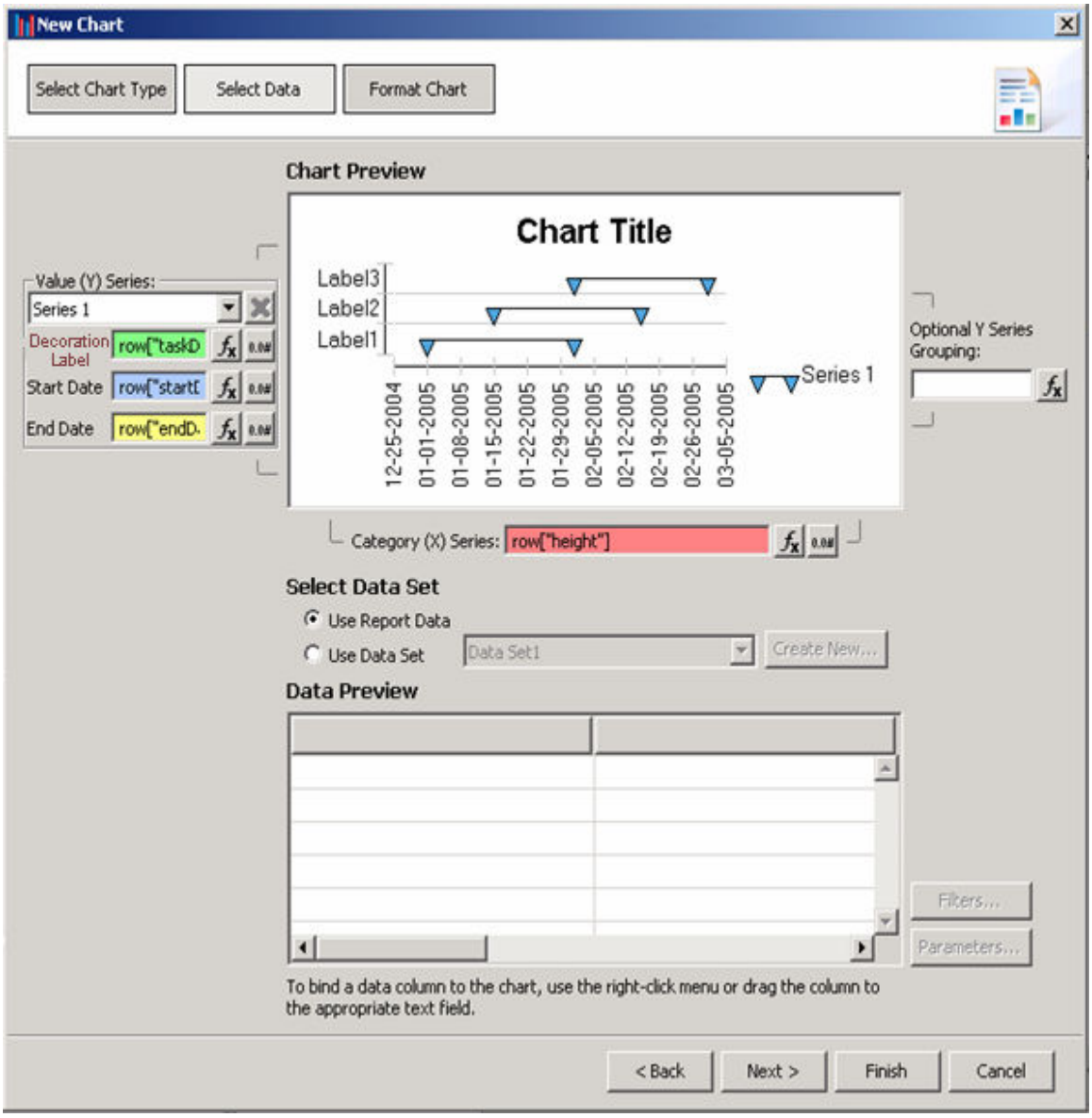
```
public IAxesDecorator getAxesDecorator( OneAxis ax );
```

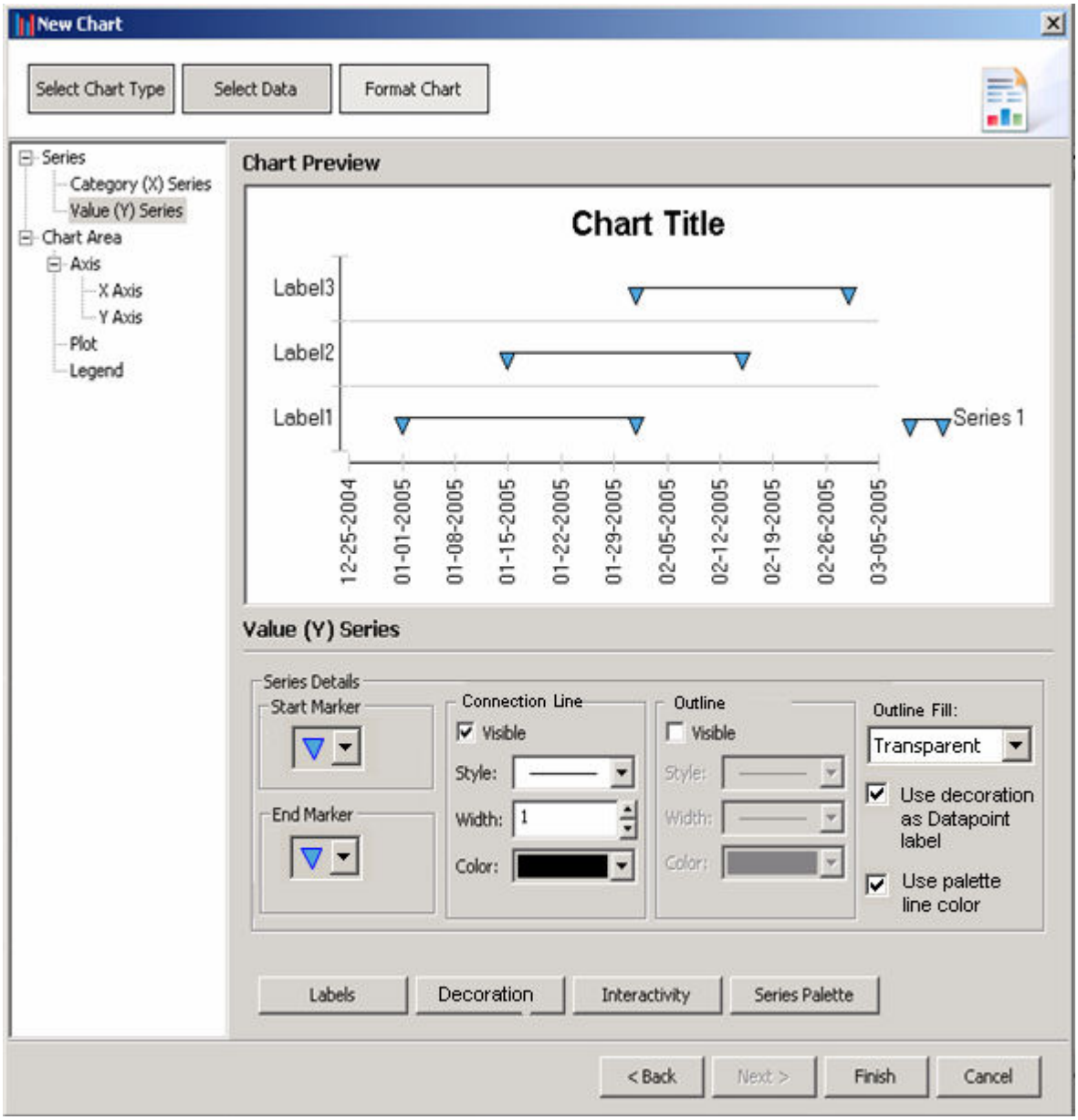
This is to retrieve the decorator instance from the subclass for each axis. The subclass should check the axis argument and decide which axis would be decorated. The default implementation simply returns null.

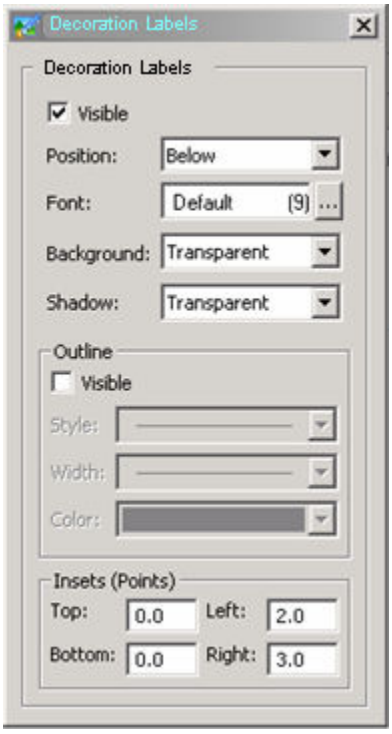
.

2.4 UI Mockup


New Chart
Select Chart Type
Select Data
Format Chart
Chart Preview
Chart Title
02-01-2005 - 03-01-2005
01-15-2005 - 02-15-2005
01-01-2005 - 02-01-2005
Label3
Label2
Label1
Series 1
Nov 2004
Jan 2005
Mar 2005
May 2005
Select Chart Type
Select Subtype
Line Chart
Area Chart
Pie Chart
Meter Chart
Scatter Chart
Stock Chart
Gantt Chart
Dimension:
2D
Output Format:
SVG
Multiple Y Axis:
None
Series Type:
Gantt Series
Orientation:
Flip Axis
< Back
Next >
Finish
Cancel


New Chart
Select Chart Type
Select Data
Format Chart
Chart Preview
Chart Title
Label3
Label2
Label1
Series 1
12-25-2004
01-01-2005
01-08-2005
01-15-2005
01-22-2005
01-29-2005
02-05-2005
02-12-2005
02-19-2005
02-26-2005
03-05-2005
Value (Y) Series:
Series 1
Decoration Label
row["taskD
Start Date
row["start[
End Date
row["endD-
Optional Y Series Grouping:
Category (X) Series:
row["height"]
Select Data Set
Use Report Data
Use Data Set
Data Set1
Create New...
Data Preview
Filters...
Parameters...
To bind a data column to the chart, use the right-click menu or drag the column to the appropriate text field.
< Back
Next >
Finish
Cancel


New Chart
Select Chart Type
Select Data
Format Chart
Series
Category (X) Series
Value (Y) Series
Chart Area
Axis
X Axis
Y Axis
Plot
Legend
Chart Preview
Chart Title
Label3
Label2
Label1
Series 1
12-25-2004
01-01-2005
01-08-2005
01-15-2005
01-22-2005
01-29-2005
02-05-2005
02-12-2005
02-19-2005
02-26-2005
03-05-2005
Value (Y) Series
Series Details
Start Marker
End Marker
Connection Line
Visible
Style:
Width:
1
Color:
Outline
Visible
Style:
Width:
Color:
Outline Fill:
Transparent
Use decoration as Datapoint label
Use palette line color
Labels
Decoration
Interactivity
Series Palette
< Back
Next >
Finish
Cancel



3. Bubble Chart

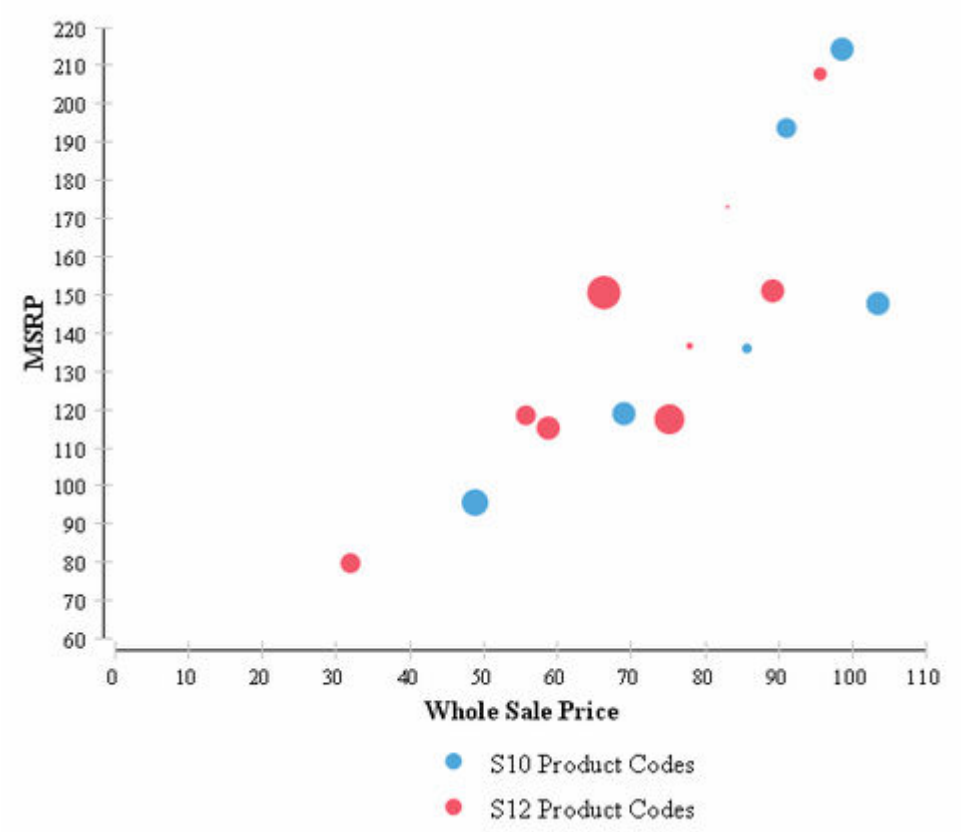
3.1 Definition

A Bubble chart is a variation of a Scatter chart in which the data points are replaced with bubbles. A Bubble chart can be used instead of a Scatter chart if your data has three data series, each of which contains a set of values. In a Bubble chart, the size of the bubbles is determined by the values in the third data series.

Bubble charts are often used to present financial data. Use a Bubble chart when you want specific values to be more visually represented in your chart by different bubble sizes

The biggest bubble always has the same size, which is approximately 1/5 of the width or height of the chart plot. Other bubbles are sized proportionally to the biggest one.

Negative sizes are represented by empty circles (unless a negative fill is available, in that case it will use that one).



3.2 Model Change

3.2.1 Type

```
<!--  Bubble Series     -->
      <xsd:complexType name="BubbleSeries">
            <xsd:annotation>
                  <xsd:documentation xml:lang="en">This is a Series type that
holds data for Bubble Charts.
                  </xsd:documentation>
            </xsd:annotation>
            <xsd:complexContent>
                  <xsd:extension base="type:ScatterSeries">
                        <xsd:sequence>
                              <xsd:element name="AccLineAttributes"
type="attribute:LineAttributes" minOccurs="0">
                                    <xsd:annotation>
                                          <xsd:documentation xml:lang="en">
                                          Specifies the attributes for the
line used to represent the acceleration line to the Bubble.
                                          </xsd:documentation>
                                    </xsd:annotation>
                              </xsd:element>
                              <xsd:element name="AccOrientation"
type="attribute:Orientation" minOccurs="0">
                                    <xsd:annotation>
                                          <xsd:documentation>Specifies the
orientation of the acceleration line.</xsd:documentation>
                                    </xsd:annotation>
                              </xsd:element>
```

```
                </xsd:sequence>
            </xsd:extension>
        </xsd:complexContent>
        </xsd:complexType>
```

BubbleSeries extends from ScatterSeries. It adds two new attributes to support the acceleration line feature.

Attribute Name	Description
AccLineAttribute	Defines the attributes for acceleration line, include line style, color, thickness, visibility, etc. The line renders from either plot left or plot bottom to the bubble.
AccOrientation	Defines the orientation of the acceleration line. Could be Horizontal or Vertical.

3.2.2 Data

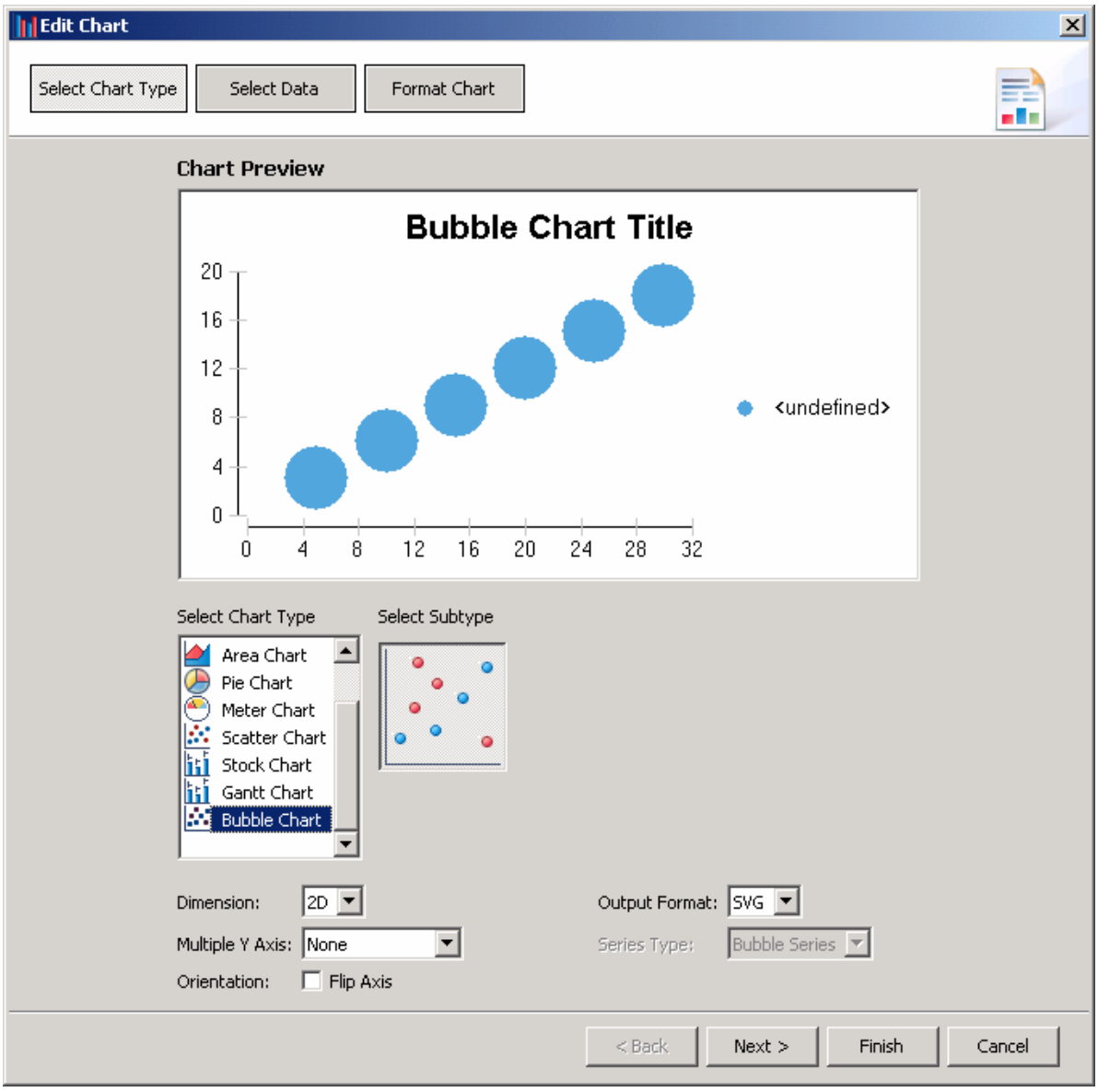
```
<xsd:complexType name="BubbleDataSet">
        <xsd:annotation>
            <xsd:documentation>This type holds the Bubble data
associated with a series.</xsd:documentation>
        </xsd:annotation>
        <xsd:complexContent>
            <xsd:extension base="data:DataSet"/>
        </xsd:complexContent>
        </xsd:complexType>
```

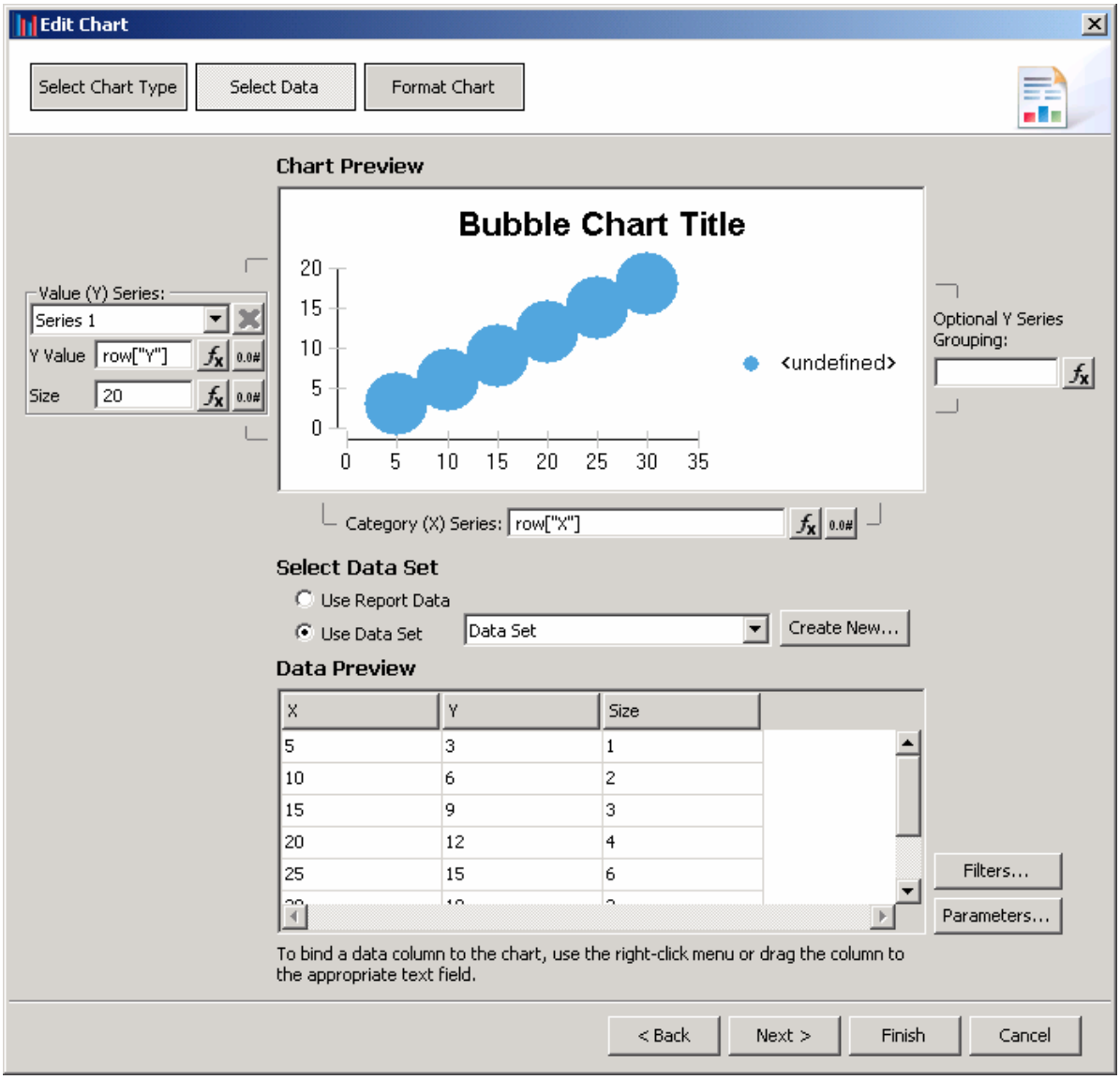
BubbleDataSet simply extends from DataSet, it holds the data needed for Bubble Chart rendering.

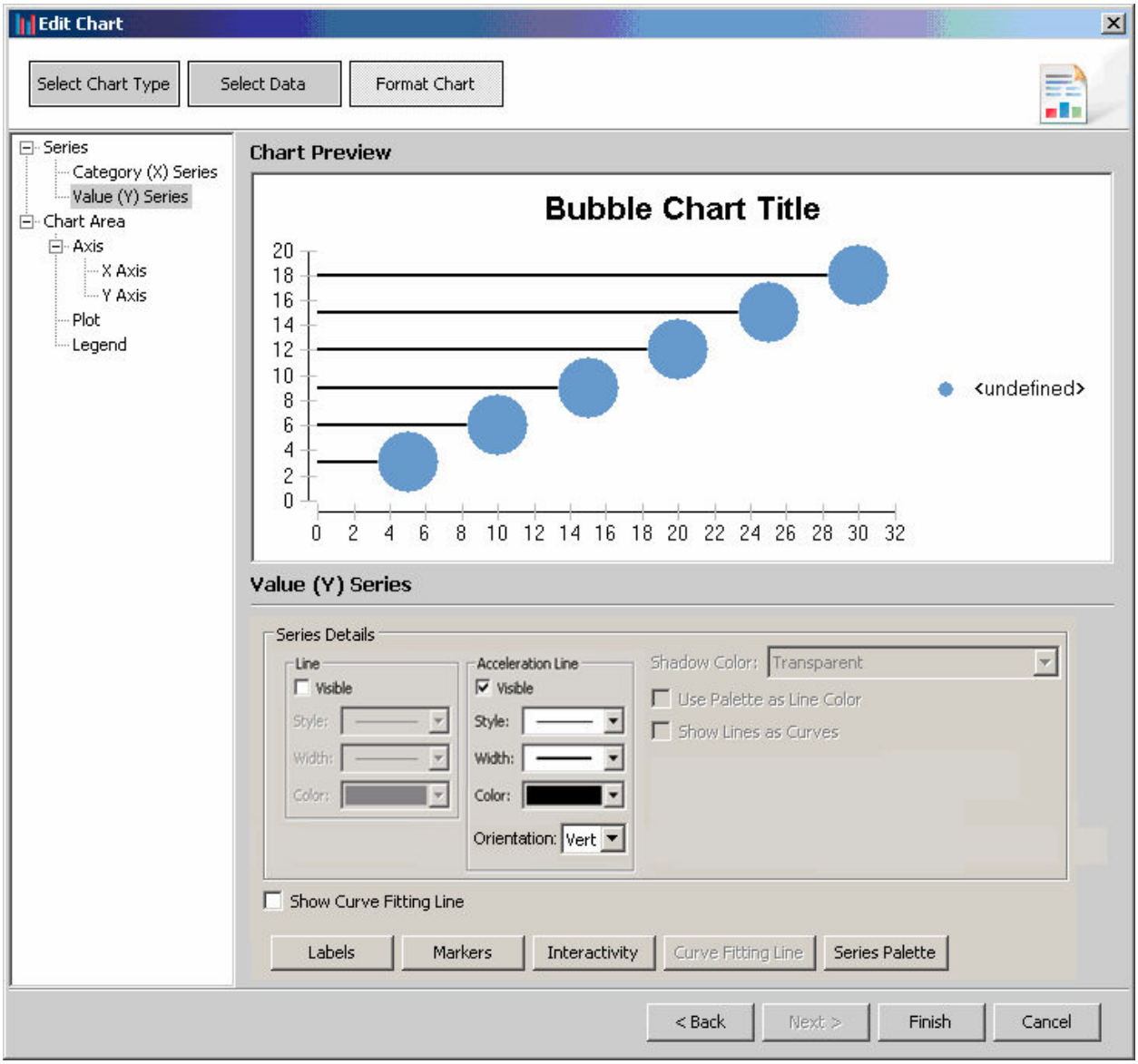
3.3 API Change

None

3.4 UI Mockup


Edit Chart
Select Chart Type
Select Data
Format Chart
Chart Preview
Bubble Chart Title
20
16
12
8
4
0
0
4
8
12
16
20
24
28
32
<undefined>
Select Chart Type
Select Subtype
Area Chart
Pie Chart
Meter Chart
Scatter Chart
Stock Chart
Gantt Chart
Bubble Chart
Dimension:
2D
Output Format:
SVG
Multiple Y Axis:
None
Series Type:
Bubble Series
Orientation:
Flip Axis
< Back
Next >
Finish
Cancel

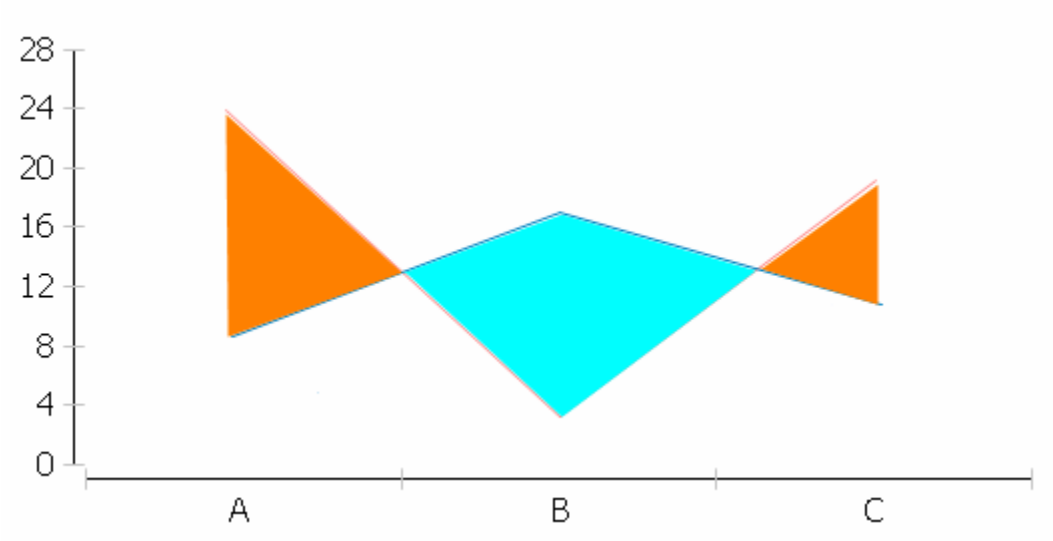

Edit Chart
Select Chart Type
Select Data
Format Chart
Chart Preview
Bubble Chart Title
20
15
10
5
0
0 5 10 15 20 25 30 35
<undefined>
Value (Y) Series:
Series 1
Y Value
row["Y"]
Size
20
0.0#
Optional Y Series Grouping:
Category (X) Series:
row["X"]
Select Data Set
Use Report Data
Use Data Set
Data Set
Create New...
Data Preview
X
Y
Size
5 3 1
10 6 2
15 9 3
20 12 4
25 15 6
Filters...
Parameters...
To bind a data column to the chart, use the right-click menu or drag the column to the appropriate text field.
< Back
Next >
Finish
Cancel



4. Difference Chart

4.1 Definition

A Difference Chart is usually used to represent difference between two data fields. It's almost the same as a stacked area chart, except it uses two fills to represent the positive and negative areas. Of course the user can choose to display both areas in the same colors.



4.2 Model Change

4.2.1 New Difference Series

<!-- Difference Series-->

<xsd:complexType name="DifferenceSeries">

<xsd:annotation>

<xsd:documentation>This is a Series type that holds data for Difference Charts.</xsd:documentation>

</xsd:annotation>

<xsd:complexContent>

<xsd:extension base="type:AreaSeries"/>

</xsd:complexContent>

TBD (Marker+Lineattribute of second series)

</xsd:complexType>

The DifferenceSeries extends the AreaSeries, it also adds two new attributes for the marker and lineattribute of the second line of data (since the difference chart series corresponds to two lines and the area in between).

4.2.2 New MultipleFill for changing colors

A new Fill type will be available in the palette to let the user choose a positive and negative color for a given series. This change goes beyond the Difference Series, since other series types will also be able to use it:

Bar/Area/Line Series: Negative values on the Y Axis will be plotted with the negative color

Stock Series: If Open>Close, the candlestick will use the negative color (currently it's just blank)

Difference Series: If the first value becomes greater than the second, the area will use the negative color.

In the future more colors could be added to this MultipleFill for further enhancements

4.3 Data

```
<xsd:complexType name="DifferenceDataSet">
            <xsd:annotation>
                  <xsd:documentation>This type holds the Bubble data
associated with a series.</xsd:documentation>
            </xsd:annotation>
            <xsd:complexContent>
                  <xsd:extension base="data:DataSet"/>
            </xsd:complexContent>
            </xsd:complexType>
```

`DifferenceDataSet` simply extends from DataSet, it holds the data needed for Difference Chart rendering.

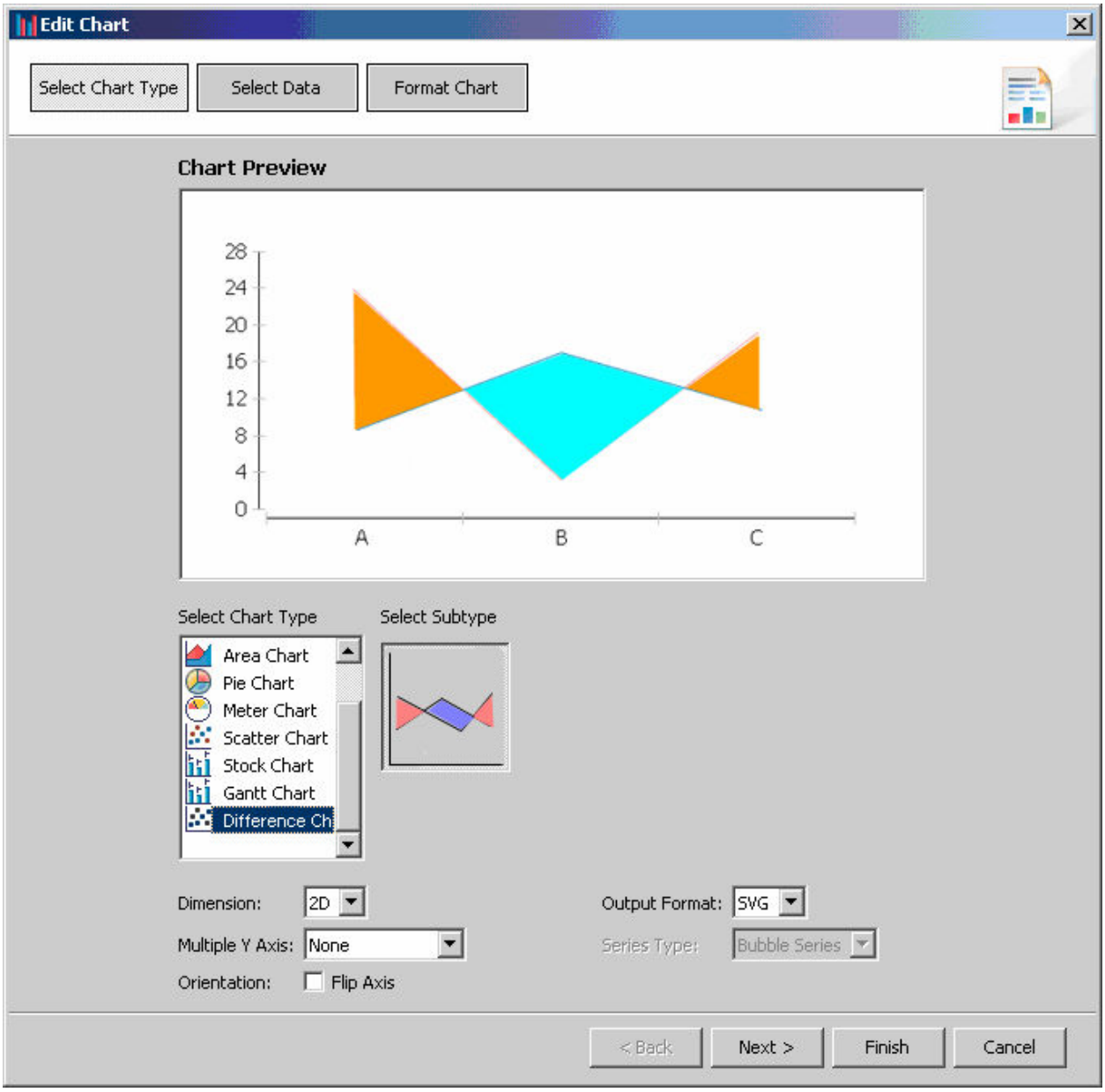
4.4 API Change.

None

4.5 UI Mockups

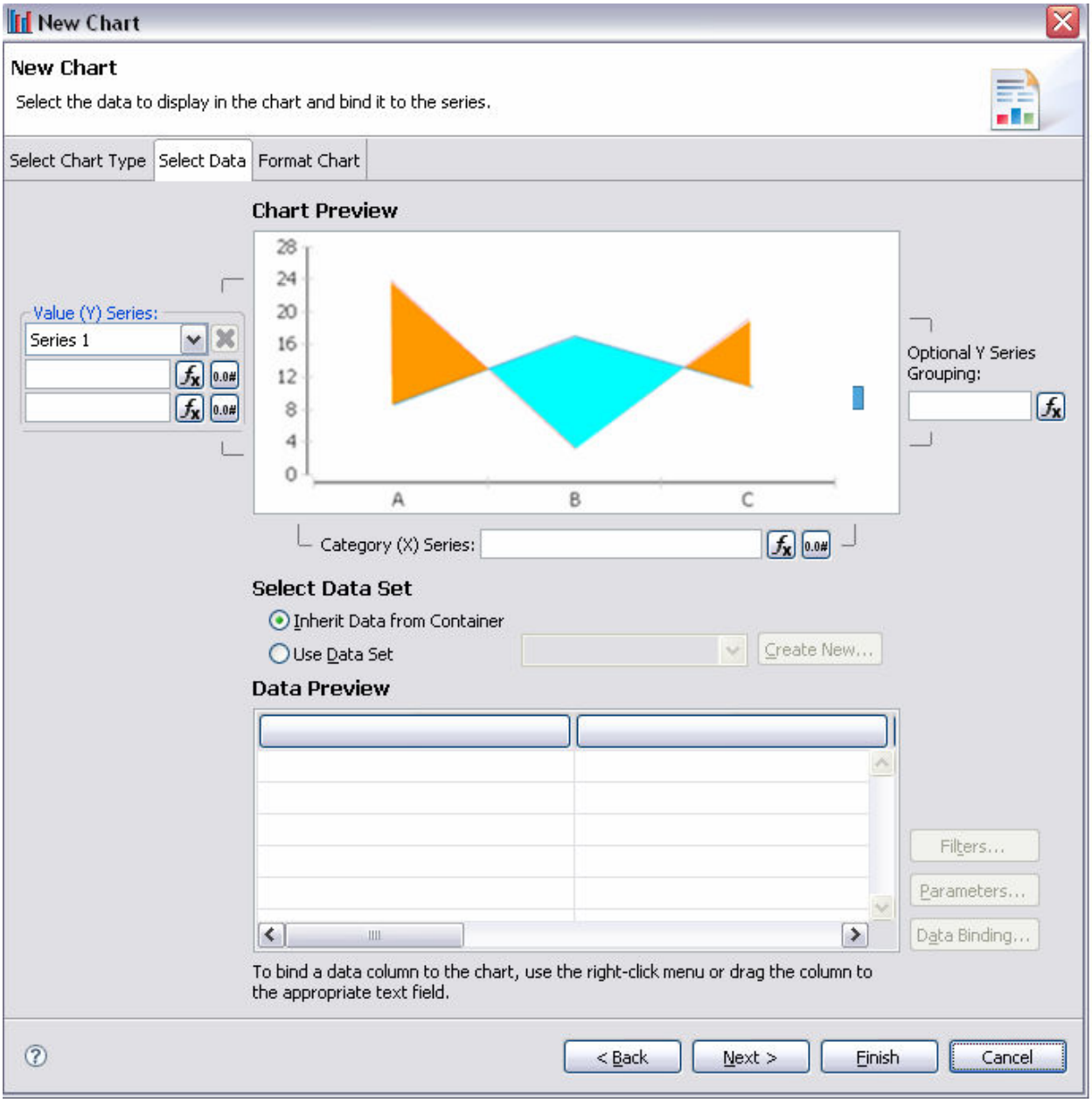
4.5.1 Select Chart Type

This is a mockup of the select chart tab. There is no subtype, and 2d with depth and 3d are not supported.



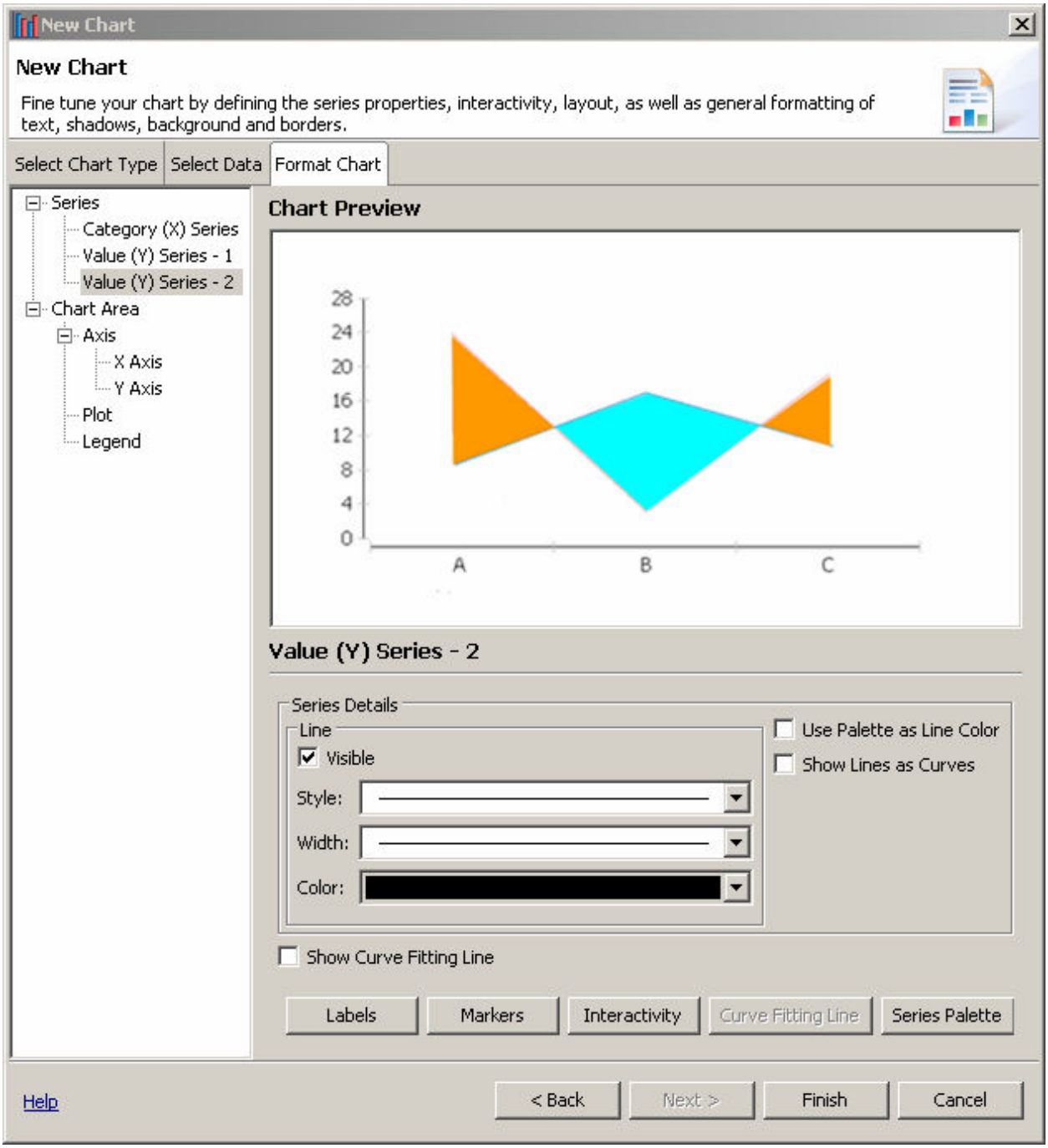
4.5.2 Select Data

The data tab will let the user enter two values:Low and High (or Reference and Value). Each value correspond to a line, and the area between both lines will be colored.



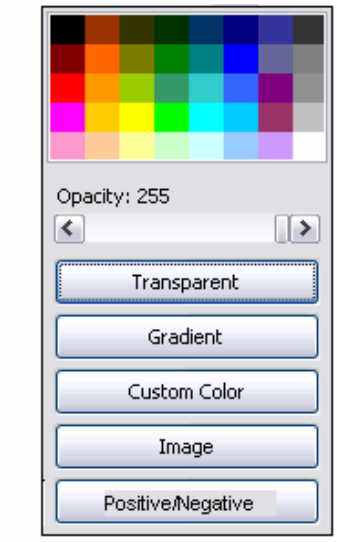
4.5.3 Format Chart

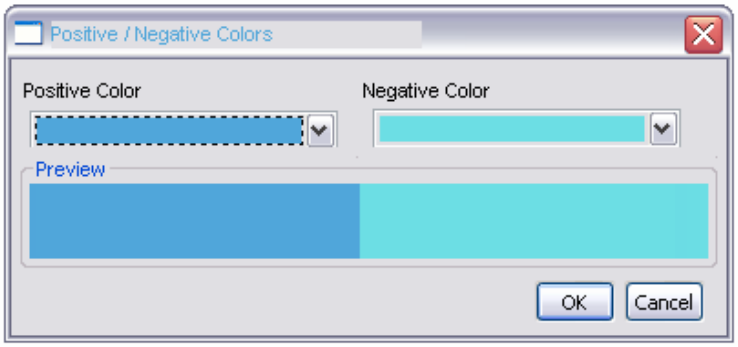
Finally the format tab looks like the Area chart one.



4.5.4 New Positive/Negative Fill UI in Palette

Clicking on Series Palette will bring this new UI for all types to let the user choose a Positive and Negative color for one Series:





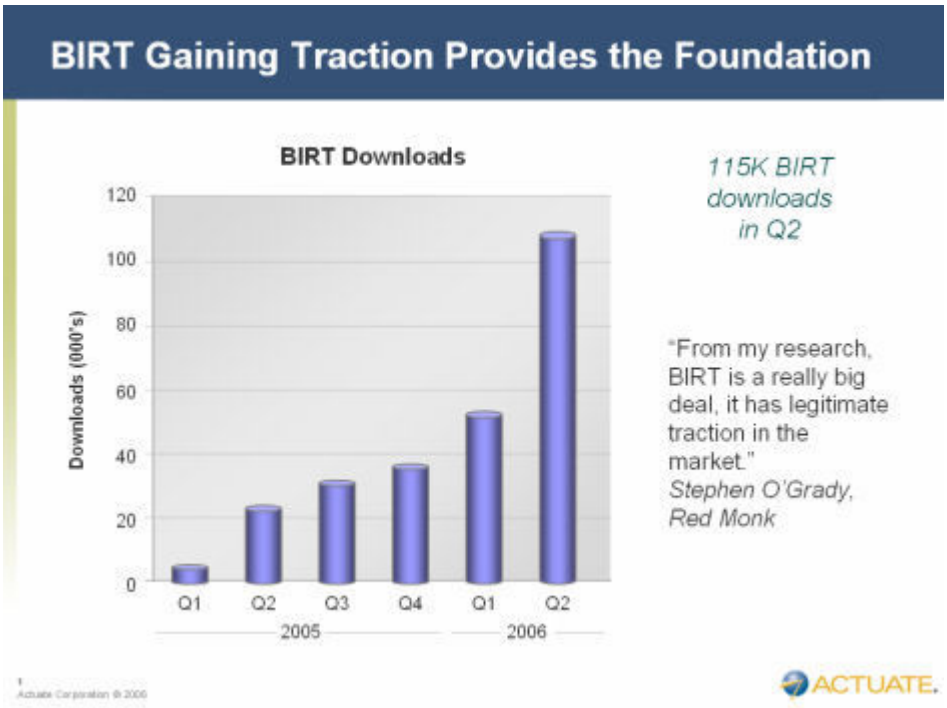
.

5. Tube Chart

5.1 Definition

A tube chart is just a special bar chart. It uses a tube instead of a bar to represent the data point. This type of chart is commonly used in commercial analysis report.

The tube chart can be represented by 2d, 2d-with-depth and 3d dimensions.



5.2 Model Change

```
<xsd:simpleType name="RiserType">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
        This type represents the possible values for risers supported for Bar Series.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="Rectangle"/>
        <xsd:enumeration value="Triangle"/>
        <xsd:enumeration value="Tube"/>
    </xsd:restriction>
</xsd:simpleType>
```

A new riser type “Tube“ is added. The concrete renderer implementation will recognize this type and render the series as tube.

5.3 API Change

None

5.4 UI Mockup

No UI change is necessary, only a new “Tube” choice will be added in the Riser type dropdown for tube charts.