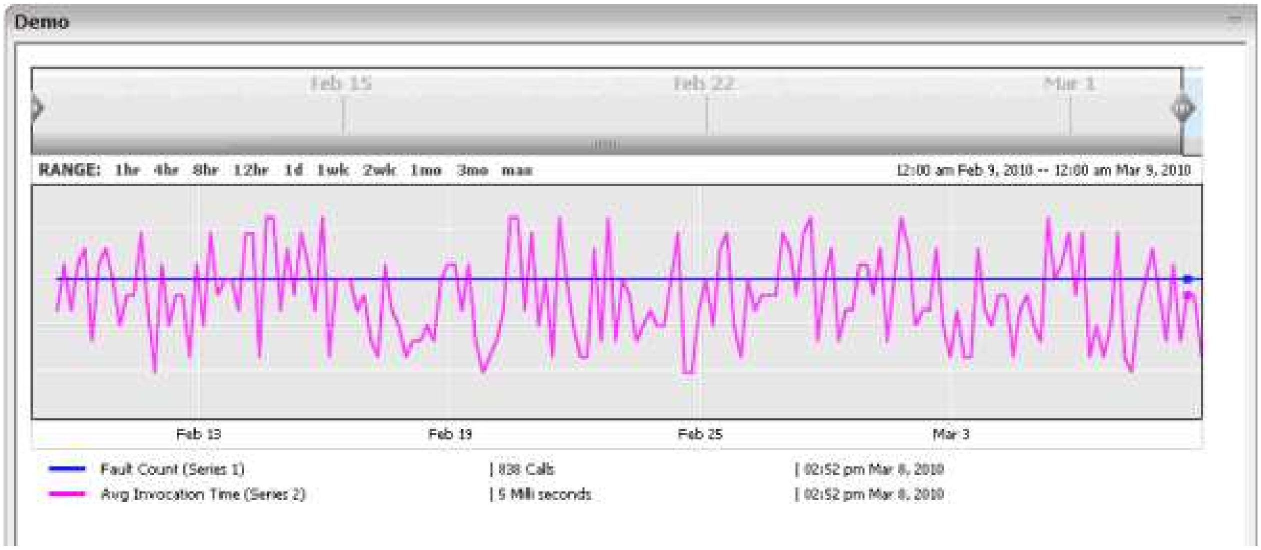
LINECHART

The LINECHART control allows you to build line charts. For each line chart, you can define a time range and render multiple series within this time range. For each series, you specify name, measures and the values you would like to see for the series.

The following topics are covered below:

- Example
- Properties

Example


Demo
Feb 15
Feb 22
Mar 1
RANGE: 1hr 4hr 8hr 12hr 1d 1wk 2wk 1mo 3mo max
12:00 am Feb 9, 2010 -- 12:00 am Mar 9, 2010
Feb 13
Feb 19
Feb 25
Mar 3
Fault Count (Series 1) | 838 Calls | 02:52 pm Mar 8, 2010
Avg Invocation Time (Series 2) | 5 Milli seconds | 02:52 pm Mar 8, 2010

The above example shows two series in the time range of Februar 9th through March 9th.

Note:
The **80_linechart** example in the **cisdemos** project contains a complete working example including layout and source code.

The XML layout definition of above example is:

```
<page model="com.softwareag.cis.test80.LINECHARTAdapter">
  <titlebar name="LINECHART Control">
  </titlebar>
  <pagebody takefullheight="true">
    <rowarea name="Demo" height="100%">
      <itr takefullwidth="true" height="100%">
        <linechart linechartinfoprop="lineChartInfo" width="100%" height="350px">
        </linechart>
      </itr>
    </rowarea>
  </pagebody>
  <statusbar withdistance="false">
  </statusbar>
</page>
```

In the underlying Java adapter, the LINECHART control is represented by an instance of the class `com.softwareag.cis.server.util.LINECHARTInfo`. See also the corresponding Java documentation.

```
public class LINECHARTAdapter
    extends Adapter
{

    // property >lineChartInfo<
    LINECHARTInfo m_lineChartInfo=new LINECHARTInfo();

    public LINECHARTInfo getLineChartInfo()
    {
       return m_lineChartInfo;
    }
    public void setLineChartInfo(LINECHARTInfo value)
    {
       m_lineChartInfo = value;
    }
}
```

Properties

Basic			
linechartinfoprop	Name of adapter property that represents the line chart on server side. The property must be of type "LINECHARTInfo".	Obligatory	
width	Width of the control. There are three possibilities to define the width: (A) You do not define a width at all. In this case the width of the control will either be a default width or - in case of container controls - it will follow the width that is occupied by its content. (B) Pixel sizing: just input a number value (e.g. "100"). (C) Percentage sizing: input a percantage value (e.g. "50%"). Pay attention: percentage sizing will only bring up correct results if the parent element of the control properly defines a width this control can reference. If you specify this control to have a width of 50% then the parent element (e.g. an ITR-row) may itself define a width of "100%". If the parent element does not specify a width then the rendering result may not represent what you expect.	Optional	100 120 140 160 180 200 50% 100%

height	Height of the control. There are three possibilities to define the height: (A) You do not define a height at all. As consequence the control will be rendered with its default height. If the control is a container control (containing) other controls then the height of the control will follow the height of its content. (B) Pixel sizing: just input a number value (e.g. "20"). (C) Percentage sizing: input a percantage value (e.g. "50%"). Pay attention: percentage sizing will only bring up correct results if the parent element of the control properly defines a height this control can reference. If you specify this control to have a height of 50% then the parent element (e.g. an ITR-row) may itself define a height of "100%". If the parent element does not specify a width then the rendering result may not represent what you expect.	Optional	100 150 200 250 300 250 400 50% 100%
align	Horizontal alignment of control in its column. Each control is "packaged" into a column. The column itself is part of a row (e.g. ITR or TR). Sometimes the size of the column is bigger than the size of the control itself. In this case the "align" property specifies the position of the control inside the column. In most cases you do not require the align control to be explicitly defined because the size of the column around the controls exactly is sized in the same way as the contained control. If you want to directly control the alignment of text: in most text based controls there is an explicit property "textalign" in which you align the control's contained text.	Optional	left center right
valign	Vertical alignment of control in its column. Each control is "packaged" into a column. The column itself is part of a row (e.g. ITR or TR). Sometimtes the size of the column is bigger than the size of the control. In this case the "align" property specify the position of the control inside the column.	Optional	top middle bottom

colspan	Column spanning of control. If you use TR table rows then you may sometimes want to control the number of columns your control occupies. By default it is "1" - but you may want to define the control to span over more than one columns. The property only makes sense in table rows that are snychronized within one container (i.e. TR, STR table rows). It does not make sense in ITR rows, because these rows are explicitly not synched.	Optional	1 2 3 4 5 50 int-value
rowspan	Row spanning of control. If you use TR table rows then you may sometimes want to control the number of rows your control occupies. By default it is "1" - but you may want to define the control two span over more than one columns. The property only makes sense in table rows that are snychronized within one container (i.e. TR, STR table rows). It does not make sense in ITR rows, because these rows are explicitly not synched.	Optional	1 2 3 4 5 50 int-value