Table of Contents

The JVx Enterprise Application Framework offers the convenience of platform independent user interfaces.However, as it is not necessary for certain tasks, it is also possible to use a specific technology, such as classic Java Swing.

Since JVx already offers swing implementation of the user interfaces, you cannot only directly integrate JVx´s swing components in your applications, but also integrate any swing component in JVx, or simply use it, without adaptation and implementation efforts.

Integration is not limited to the swing components. For example, SWT is also possible as long as a JVx implementation is available.

The following example shows the integration of standard swing components in theJVx Enterprise Application Framework.

Example

We want to include a chart from JFreeChart in an existing JVx application. The following is the integrated component:



Source code for this example:

```
//create the chart
DefaultPieDataset dataset = new DefaultPieDataset();
dataset.setValue("Germany", 29);
dataset.setValue("Spain", 20);
dataset.setValue("Austria", 51);

JFreeChart chart = ChartFactory.createPieChart3D("JVx integration",
                                                 dataset,
                                                 true,
                                                 true,
                                                 false);

PiePlot3D plot = (PiePlot3D) chart.getPlot();
plot.setStartAngle(290);
plot.setDirection(Rotation.CLOCKWISE);
plot.setForegroundAlpha(0.5f);

ChartPanel chartPanel = new ChartPanel(chart);

panDetails.setLayout(layout);
panDetails.add(gpanDedails, layout.getConstraints(0, 0, -1, 0));

//ADD the chart as custom component
panDetails.add(new UICustomComponent(chartPanel), layout.getConstraints(0,
1, -1, -1));
```

Using a single line of code

```
panDetails.add(new UICustomComponent(chartPanel), layout.getConstraints(0,
1, -1, -1));
```

Any swing component can be integrated in JVx with full access to the JVx and swing components.

With the integration of technology dependent components, as shown in our example, the use of other technologies is very limited or not possible (using Swing components with SWT is difficult).

However, JVx allows for the implementation of technology-independent special components which can then be integrated in the application. The respective technology will then always use compatible components.

From:
https://doc.sibvisions.com/ - **Documentation**

Permanent link:
https://doc.sibvisions.com/jvx/client/gui/userdefined_components

Last update: **2020/06/08 13:37**