

Eclipse SWTChart

Some of you may have used SWTChart [1] already. It's a powerful library for SWT to create bar, line and scatter charts. The project was initially started by Yoshitaka Yanagida in 2008. Since then, it has gained a lot of attraction due to its clear and well designed API. In 2018, SWTChart has been proposed to become an official Eclipse project. The proposal was posted on June, 26th. The review was successful by October, 3rd. It was a milestone, due to the fact that SWTChart is used heavily as part of Eclipse Chemclipse to plot chemical data. The task in Chemclipse is to visualize data from chromatography / spectrometry in a comprehensive way. Many improvements and extension have been implemented inside of Chemclipse. In January 2017, Philip Wenig started to refactor the SWTChart extensions of Chemclipse and moved them as a separate bundle to the Eclipse Advanced Visualization Project (EAVP). Over time, the extensions have reached a mature status. Thus it was a good timing that SWTChart has been proposed as an official Eclipse project. Both SWTChart and the extensions have been merged and are officially available here now:

<https://projects.eclipse.org/projects/science.swtchart>

<https://github.com/eclipse/swtchart>

The GitHub repository contains a description how to easily build the project:

<https://github.com/eclipse/swtchart/blob/develop/org.eclipse.swtchart.cbi/Build.txt>

Image 1 shows some typical chart extensions of bar, line and scatter plots. In the upper left is a line chart with additional labels. The option to add labels has been also refactored so that it is easy now to show user defined tags. More line charts are shown in upper right and bottom center. Via the settings, one can define whether to show a negative range and/or to restrict zooming outside of the available data range. Even a reversed axis can be displayed. It means that the axis direction is not „negative to positive“, but „positive to negative“. That's a functionality especially needed by scientific charts, e.g. to display FTIR or NMR data. Moreover, a slider can be enabled via settings for each chart. This is a helpful extension when one has to review data series quickly. A bar chart is shown in lower left, in this case a mass spectrum with an automatic label option of the displayed 5 highest m/z values. A scatter plot is shown in upper center and lower right, e.g. to display results from a PCA analysis. To inspect the various settings, please have a look at the example charts.

The source code contains a run configuration „SWTChart Extension Examples.launch“ which lets you explorer many chart demos out of the box.

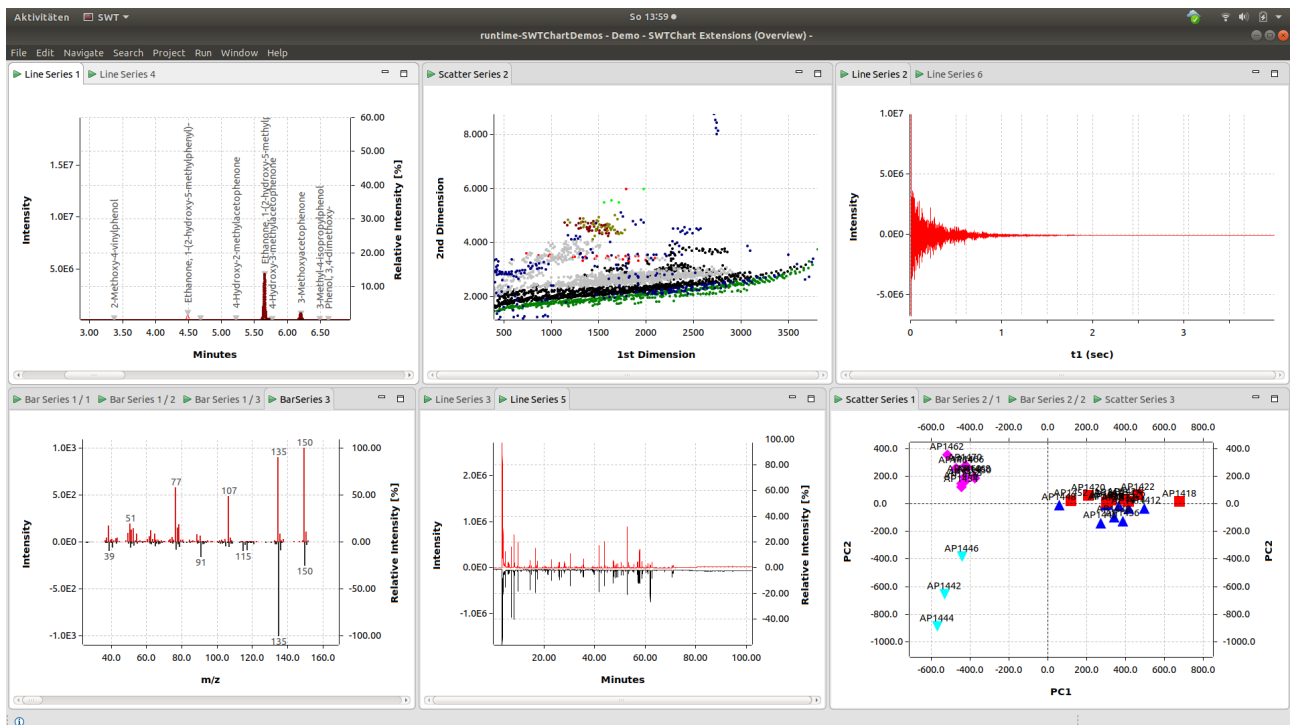


Image 1: SWTChart extension examples of line, bar and scatter plots.

How does it look like to create a chart? You only need to extend from BarChart, LineChart or ScatterChart. That's it. The primary x and y axis can be modified via the chart settings. Additional secondary x and y axes can be also added/removed dynamically via the chart settings.

```
public class MyChart extends BarChart {

    public MyChart(Composite parent) {
        super(parent, SWT.NONE);
        initialize();
    }

    private void initialize() {

        IChartSettings chartSettings = getChartSettings();
        chartSettings.setCreateMenu(true);
        applySettings(chartSettings);
        //
        List<IBarSeriesData> barSeriesDataList = new
        ArrayList<IBarSeriesData>();
        IBarSeriesData barSeriesData = new
        BarSeriesData(SeriesConverter.getSeriesXY(SeriesConverter.BAR_SERIES_1));
        barSeriesDataList.add(barSeriesData);
        //
        addSeriesData(barSeriesDataList);
    }
}
```

A menu can be enabled/disabled for each chart. The menu is highly configurable, so that new menu items can be added easily or some or all existing menu items can be replaced by user specific actions. By default, the menu lets one export the data to various output formats, Image 2. This works seamlessly as each axis can be modified dynamically and the only accepted data format is double[] arrays for x and y series. No specific data container is used which makes it easy to write export extensions. For example, an export to a R script is no problem, Image 3.

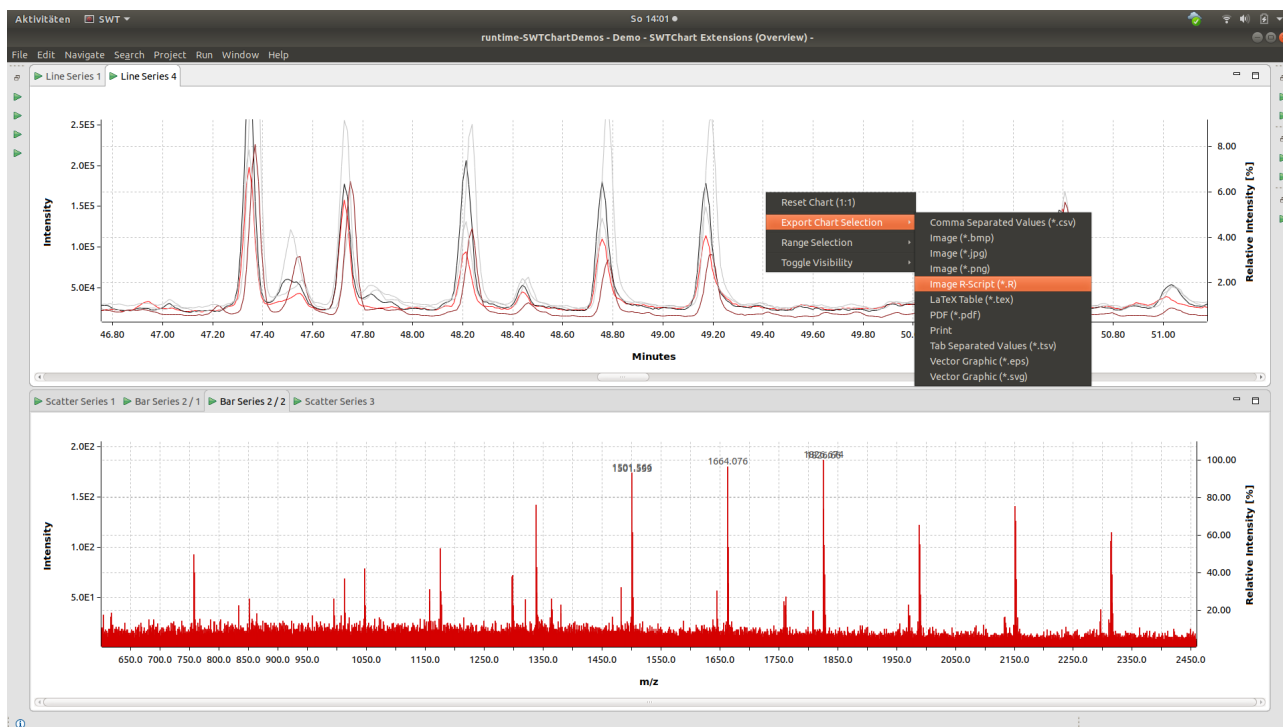


Image 2: Menu export options and display of large data sets.

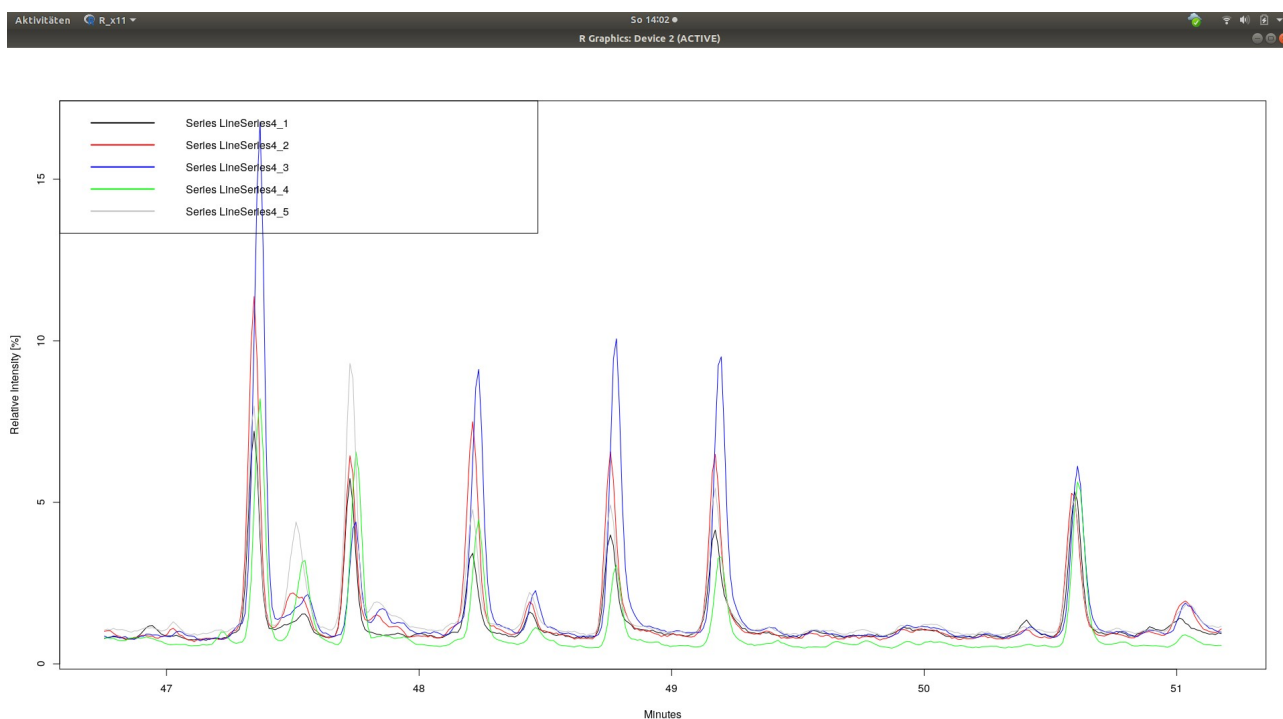


Image 3: Chart data exported to a R script to create a scientific publication.

I hope that you like the Eclipse SWTChart project. Contributions and issues can be filed here:

<https://github.com/eclipse/swtchart/issues>

[1] <http://www.swtchart.org>