



Tutorial

Build PFGGrid from scratch

PFGGrid

E-Mail: info@pfgrid.com

Homepage: <http://eclipse.pfgrid.com>

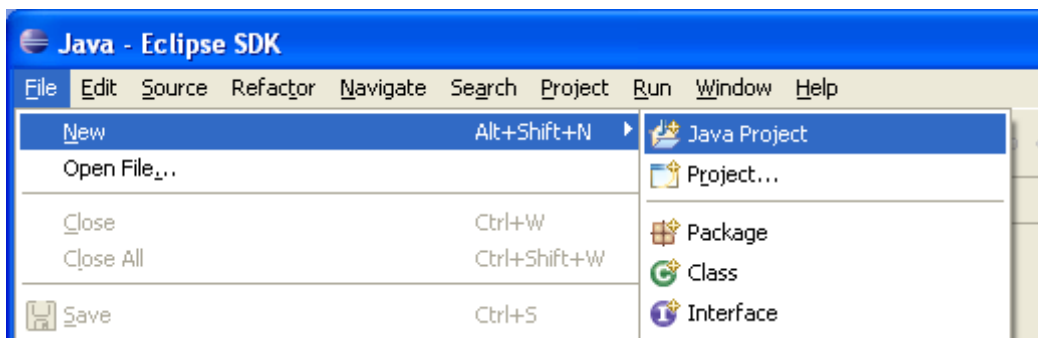
Date: 09.07.2010

Introduction

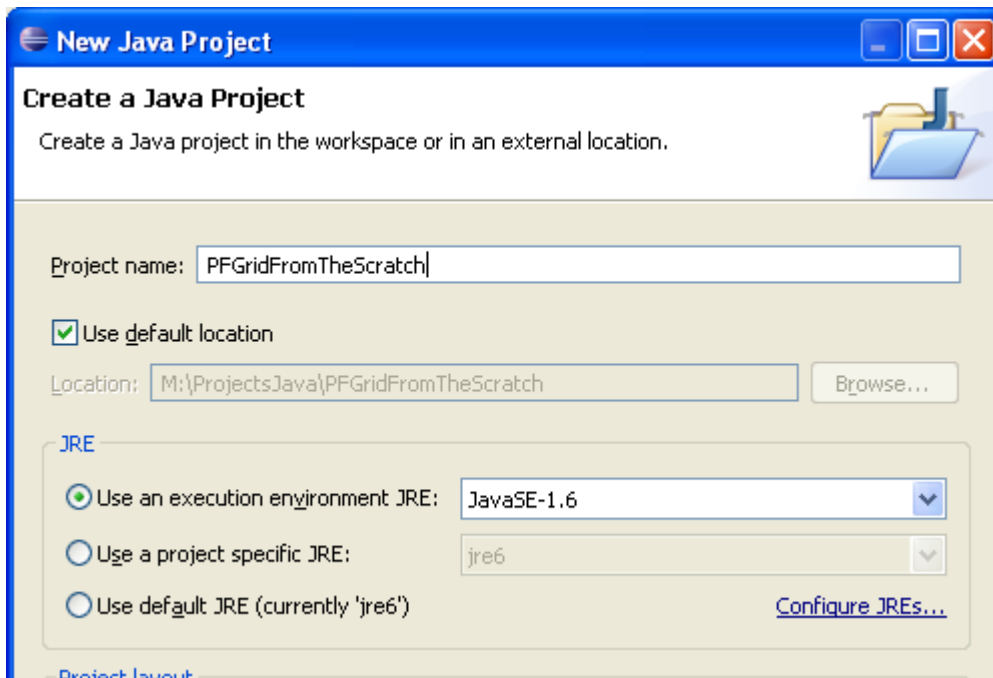
This tutorial consists of two steps. In the first step we create a window within a embedded PFGrid with six columns. In the second step we add diverse data into the grid with multiline content, icons, progress bars and checkboxes.

Step 1: Create Java Project

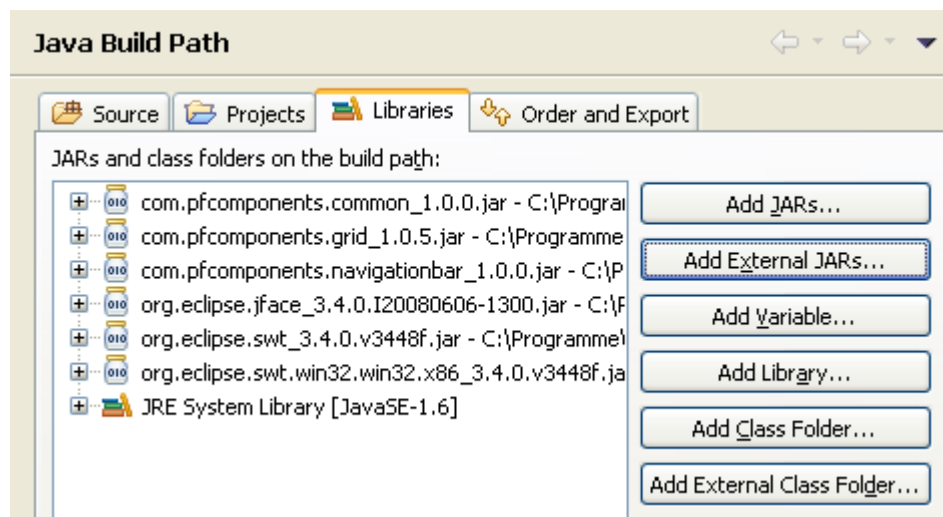
To build PFGrid from scratch start Eclipse and create a new Java Project (menu File/JavaProject).



Specify the project name “*PFGridFromTheScratch*” and press button “*Next >*”.



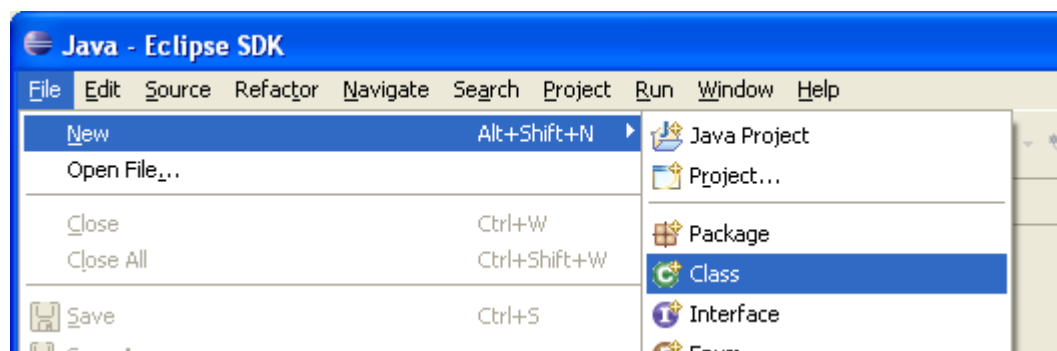
As next you get a dialog for project properties where you have to register the references to the library files for PFGrid in the section “*Java Build Path*” register tab “*Libraries*”. Select all libraries delivered with the package. You find it on the following path: “*%ProgramFiles%\PFGrid SWT\lib*”.



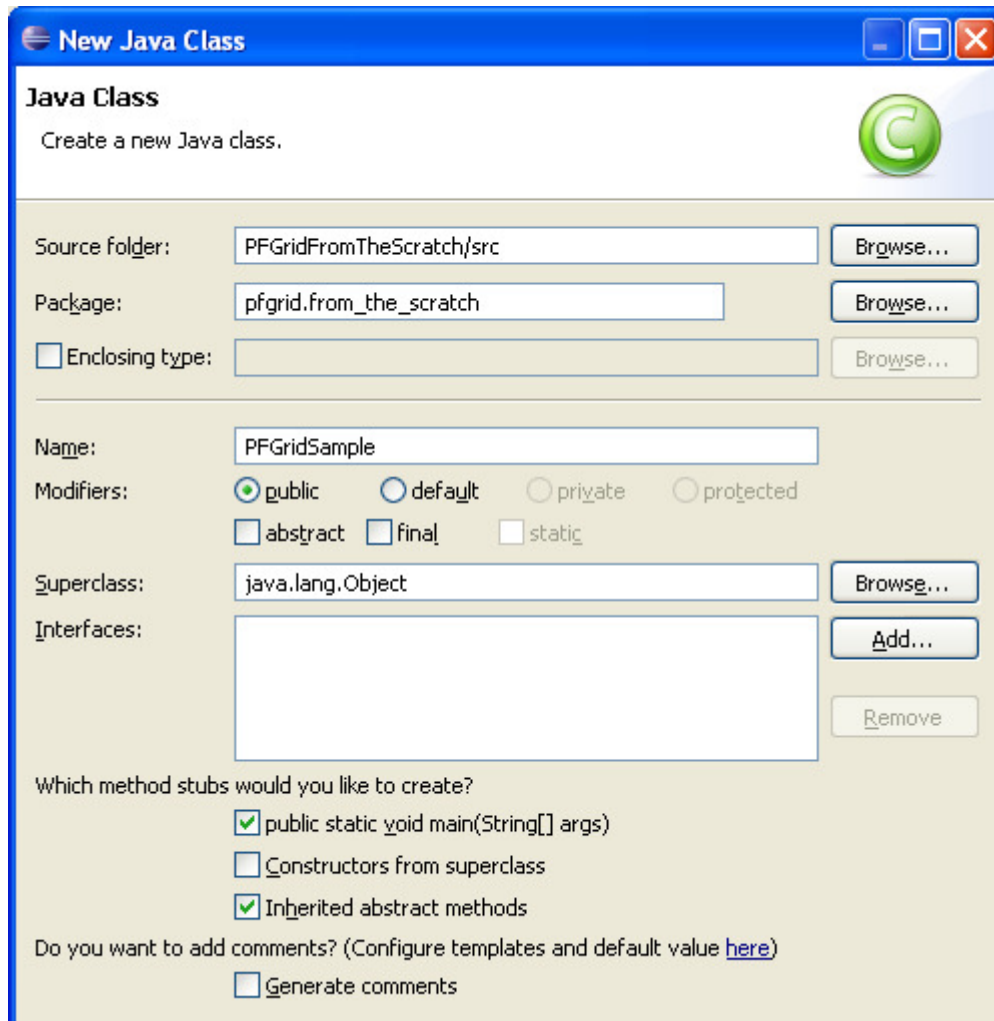
To confirm the additional libraries press “OK” and the project is created.

Create Java Class

As next create a Java Class, menu File/New/Class



In the next dialog fill in source folder, package, the class name and hook the checkbox for “*public static void main(String[] args)*”.



Open the file PFGridSample.java and copy the code above into that window. With this step you get the first simple application with the embedded PFGrid control.

```

package pfgrid.from_the_scratch;

import org.eclipse.swt.SWT;
import org.eclipse.swt.layout.FillLayout;
import org.eclipse.swt.widgets.Display;
import org.eclipse.swt.widgets.Shell;
import com.pfcomponents.grid.TreeListColumn;
import com.pfcomponents.grid.TreeListView;
import com.pfcomponents.grid.enums.Align;
import com.pfcomponents.grid.enums.SelectionType;
import com.pfcomponents.grid.render.VistaTreeListRenderer;

public class PFGridSample {

    private static TreeListView treeList;
    /**
     * @param args
     */
    public static void main(String[] args) {
        // prepare SWT environment display and shell
        Display display = new Display();
        Shell shell = new Shell(display);
    }
}

```

```
shell.setLayout(new FillLayout());
shell.setText("PFGrid-Demo with Vista renderer");
shell.setSize(700, 600);
shell.setLocation(350, 300);
// create TreeListView instance
treeList = new TreeListView(shell, SWT.V_SCROLL|SWT.H_SCROLL);
treeList.setShowGroupbox(false);
treeList.setShowCellSelectionBorder(true);
treeList.setAllowDragCopy(true);
treeList.setAlternateBackColor(true);
treeList.setSelectionType(SelectionType.Row);
treeList.setRenderer(new VistaTreeListRenderer());
// create columns (title and width) for TreeListView
TreeListColumn colFirstname = new TreeListColumn(treeList);
colFirstname.setText("Firstname");
colFirstname.setWidth(140);
TreeListColumn colLastname = new TreeListColumn(treeList);
colLastname.setText("Lastname");
colLastname.setWidth(140);
colLastname.setAllowEdit(false);
TreeListColumn colStreet = new TreeListColumn(treeList);
colStreet.setText("Street");
colStreet.setWidth(120);
TreeListColumn colZip = new TreeListColumn(treeList);
colZip.setText("Zip");
colZip.setWidth(100);
TreeListColumn colSkill = new TreeListColumn(treeList);
colSkill.setText("Skill");
colSkill.setWidth(80);
TreeListColumn colActive = new TreeListColumn(treeList);
colActive.setText("Active");
colActive.setWidth(60);
colActive.setAlign(Align.Center);
// show component on screen
shell.open();
while (!shell.isDisposed()) {
    if (!display.readAndDispatch())
        display.sleep();
}
display.dispose();
}
}
```

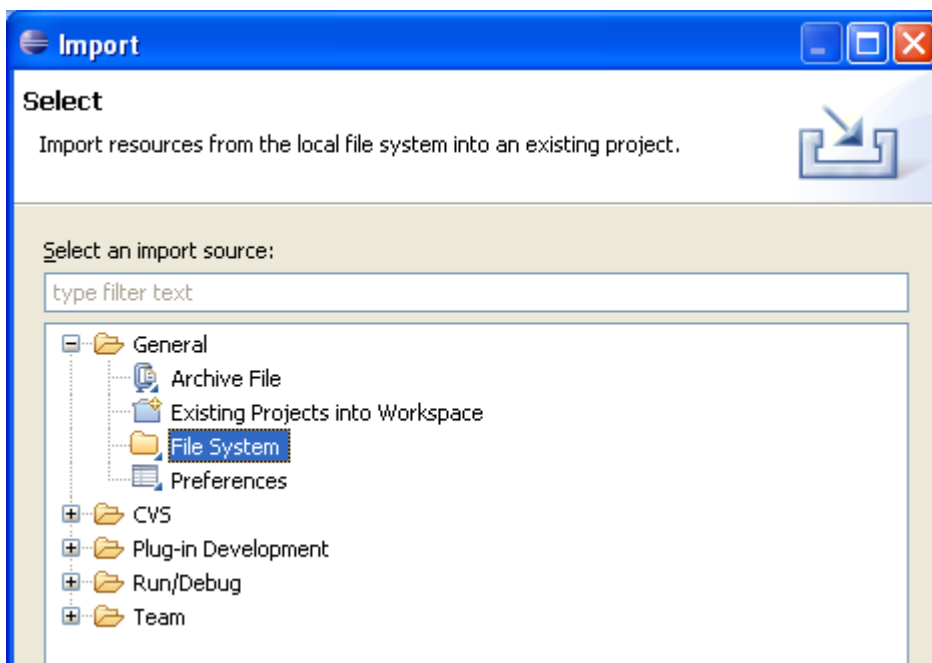
First executable Grid-application

If you start the application (menu Run/Run) you get the following window. The embedded PFGrid control contains six columns.



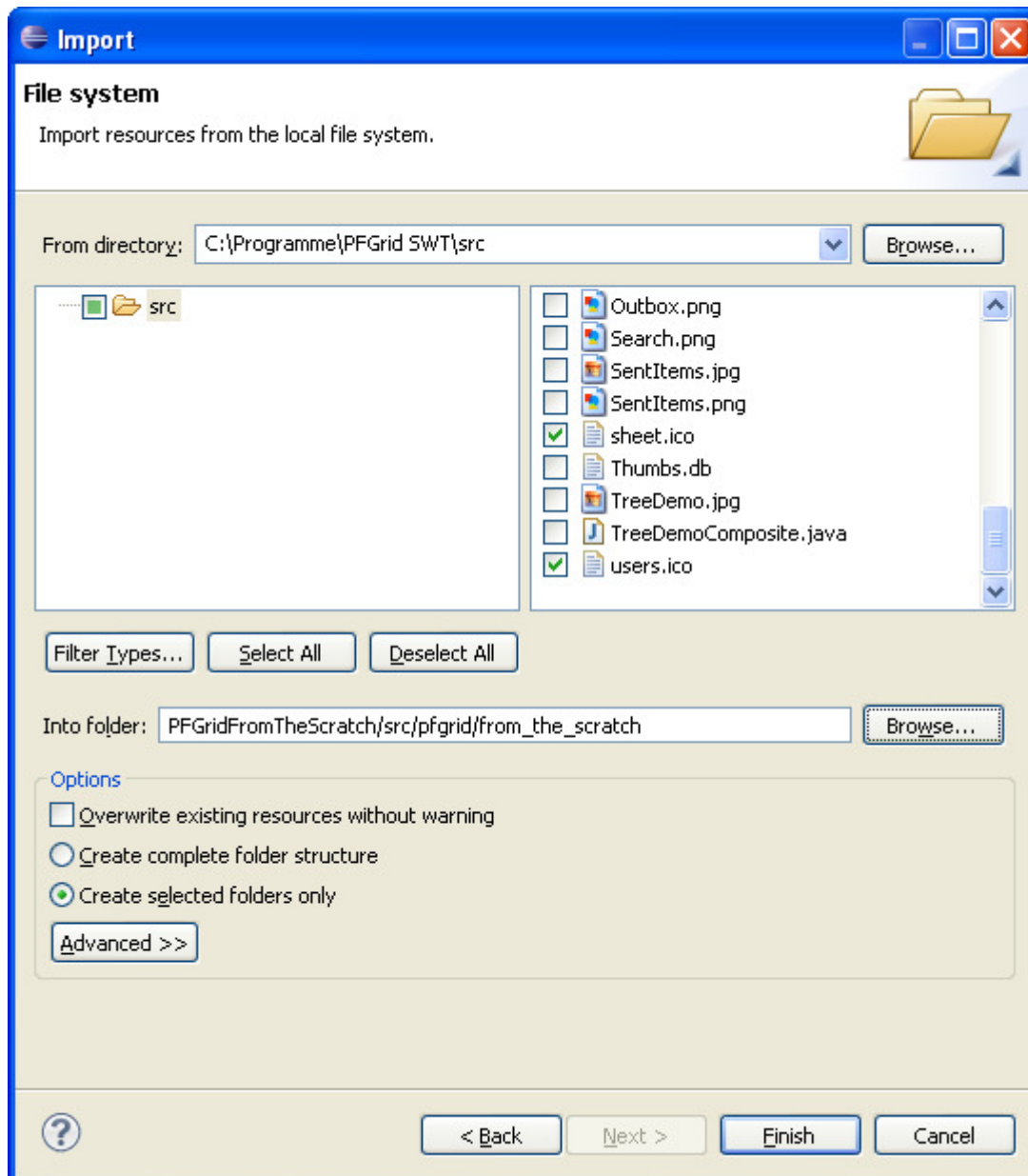
Step 2: Fill the grid component with data

The sample application uses icon resources delivered with the package. Use menu “File/Import.../General/File System” and press “Next >”.



Navigate to the installation path “%ProgramFiles%\PFGrid SWT\lib” and select in the source folder “samples\DemoBrowserGrid” the icon files “sheet.ico” and “users.ico”.

Specify the destination folder “*PFGridFromTheScratch/src/pfgrid/from_the_scratch*” and confirm with button “*Finish*”.



Copy the following source code fragments and insert them on the specified places (the grey lines represents existing lines).

Add the following import statements to the import section.

```
import java.io.InputStream;
import java.util.Random;
import org.eclipse.swt.graphics.Image;
import com.pfcomponents.grid.TreeListCell;
import com.pfcomponents.grid.TreeListRow;
import com.pfcomponents.grid.cells.CheckBoxCell;
import com.pfcomponents.grid.cells.ProgressBarCell;
```

Append the variable declaration after the last declaration statement.

```
private static TreeListView treeList;
private static Image cellImageSheet;
private static Image cellImageUser;
```

Add the following code after the last `TreeListColumn`-definition in the source-code to insert data into the tree.

```
TreeListColumn colActive = new TreeListColumn(treeList);
colActive.setText("Active");
colActive.setWidth(60);
colActive.setAlign(Align.Center);
// define images
InputStream ins1 = PFGridSample.class.getResourceAsStream("users.ico");
cellImageUser = new Image(display, ins1);
InputStream ins2 = PFGridSample.class.getResourceAsStream("sheet.ico");
cellImageSheet = new Image(display, ins2);
// add the root-nodes
for (int i = 0; i < 100; i++) {
    TreeListRow row = new TreeListRow(treeList);
    TreeListCell cellFirst = new TreeListCell(row, "Firstname" + i);
    cellFirst.setCellImage(cellImageUser);
    new TreeListCell(row, "Lastname " + i);
    new TreeListCell(row, "Street No. " + i);
    new TreeListCell(row, i * 33);
    createProgressBarCell(row);
    createCheckBoxCell(row);
    // call method addChildRows
    addChildRows(treeList, cellImageSheet, row);
}
```

The methods above create a progress bar and a checkbox.

```
public static void createProgressBarCell(TreeListRow row) {
    Random r = new Random();
    int value = (int) (r.nextDouble() * 100);
    new ProgressBarCell(row, value);
}
public static void createCheckBoxCell(TreeListRow row) {
    Random r = new Random();
    boolean value = r.nextBoolean();
    new CheckBoxCell(row, value);
}
```

The last method extends a column to an multiline entry.

```
private static void addChildRows(TreeListView treeList,
                                Image cellImageSheet, TreeListRow row) {
    for (int j = 0; j < 5; j++) {
        TreeListRow childRow = new TreeListRow(row);
        childRow.setChildrenLazy(true);
        TreeListCell cellInnerFirst = new TreeListCell(childRow,
                                                       "Child-Firstname " + j);
        cellInnerFirst.setCellImage(cellImageSheet);
        new TreeListCell(childRow, "Child-Lastname " + j);
        new TreeListCell(childRow, "Child-Street " + j);
        new TreeListCell(childRow, "Child " + j * 17);
        createProgressBarCell(childRow);
        createCheckBoxCell(childRow);
    }
}
```

When you start the application you get the following window.

Firstname	Lastname	Street	Zip	Skill	Active
Firstname0	Lastname 0	Street No. 0	0	██████████	<input type="checkbox"/>
Firstname1	Lastname 1	Street No. 1	33	██	<input checked="" type="checkbox"/>
Firstname2	Lastname 2	Street No. 2	66	██████	<input type="checkbox"/>
Firstname3	Lastname 3	Street No. 3	99	██	<input type="checkbox"/>
Firstname4	Lastname 4	Street No. 4	132	████	<input type="checkbox"/>
Firstname5	Lastname 5	Street No. 5	165	██████████	<input type="checkbox"/>
Firstname6	Lastname 6	Street No. 6	198	██	<input type="checkbox"/>
Firstname7	Lastname 7	Street No. 7	231	█	<input checked="" type="checkbox"/>
Firstname8	Lastname 8	Street No. 8	264	████	<input checked="" type="checkbox"/>
Firstname9	Lastname 9	Street No. 9	297	██	<input type="checkbox"/>
Firstname10	Lastname 10	Street No. 10	330	██	<input type="checkbox"/>
Firstname11	Lastname 11	Street No. 11	363	████	<input checked="" type="checkbox"/>
Firstname12	Lastname 12	Street No. 12	396	██████████	<input checked="" type="checkbox"/>
Firstname13	Lastname 13	Street No. 13	429	██████████	<input checked="" type="checkbox"/>
Firstname14	Lastname 14	Street No. 14	462	████	<input checked="" type="checkbox"/>
Firstname15	Lastname 15	Street No. 15	495	█	<input type="checkbox"/>
Firstname16	Lastname 16	Street No. 16	528	████	<input checked="" type="checkbox"/>
Firstname17	Lastname 17	Street No. 17	561	██████████	<input type="checkbox"/>
Firstname18	Lastname 18	Street No. 18	594	██████████	<input type="checkbox"/>
Firstname19	Lastname 19	Street No. 19	627	████	<input checked="" type="checkbox"/>
Firstname20	Lastname 20	Street No. 20	660	█	<input type="checkbox"/>
Firstname21	Lastname 21	Street No. 21	693	█	<input type="checkbox"/>