

# DWT Features

Frank Benoit  
26. Sep 2008

# DWT Features - Agenda

- DWT History+Projects
- Features
  - SWT
  - JFace
  - UI.Forms & Draw2D
- Why SWT & Eclipse?
- Future of DWT

# DWT History

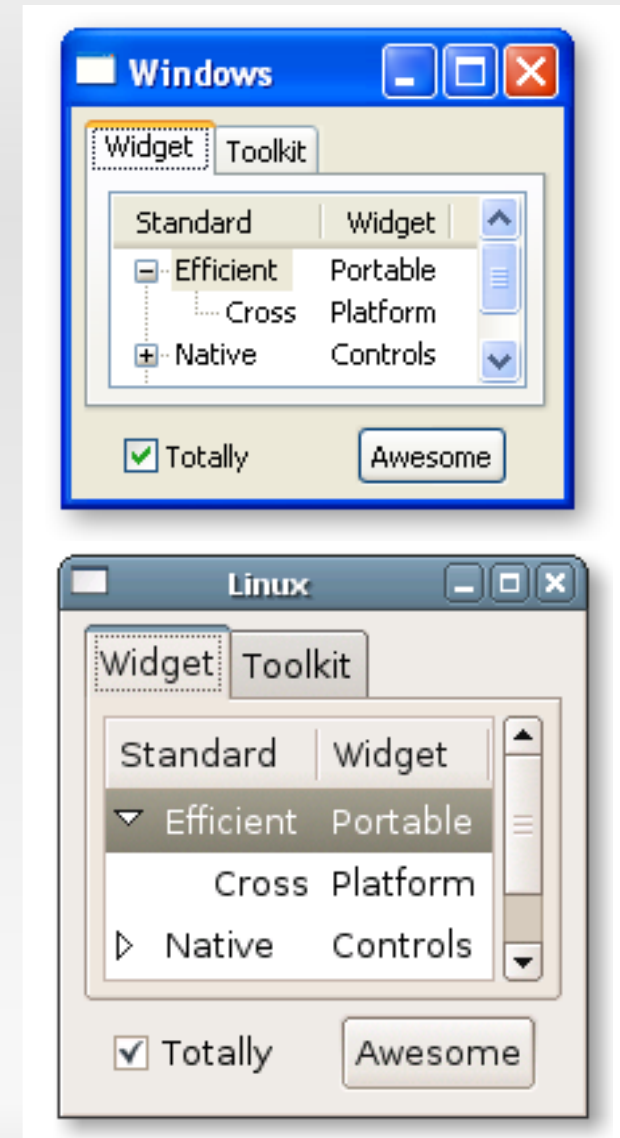
- 2004 DSource project
- 2005 Shawn Liu  
SWT 3.1 Win
- 2006 TioPort  
SWT 3.2 Win/Lin/Mac
- 2008 DWT  
SWT 3.3 Win/Lin
- July 2008 the first  
upgrade merge  
SWT 3.4 Win/Lin

# DWT Projects

- SWT Ports
  - DWT-Win 357/207kL/ 7.4MB
  - DWT-Linux 544/200kL/ 8.2MB
  - DWT-Mac
- DWT-Addons 1331/343kL/12.4MB
  - jface, jface.text, ui.forms, draw2d
- DWT-Samples 224/ 36kL/ 1.3MB

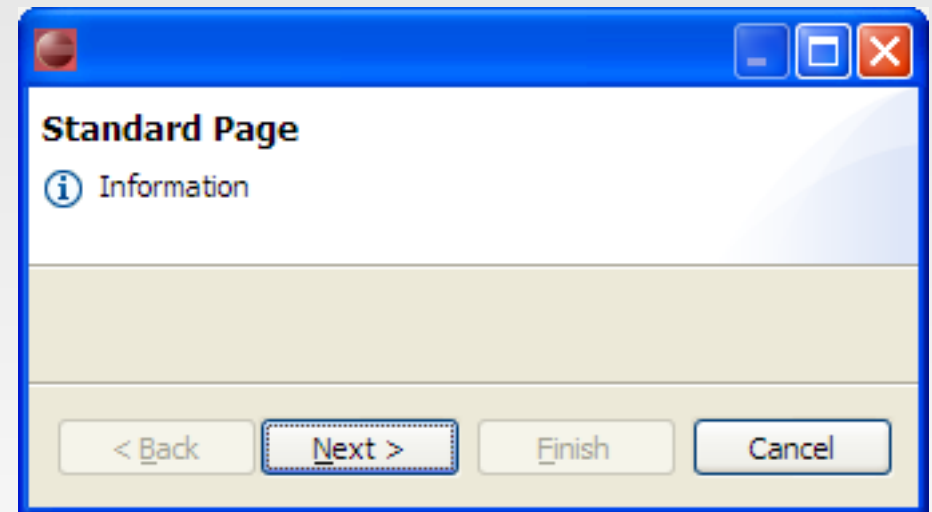
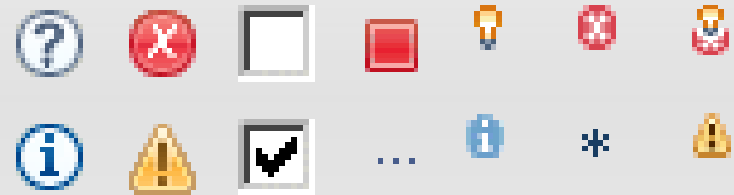
# SWT

- ~30 Standard Widgets with native implementation, more with non-native implementation
- DnD, Image formats, Printing, OpenGL
- Own widgets possible
- In Progress: Mac Cocoa port, Browser Mozilla & IE, OLE



# JFace: Resource Management

- Fonts, Colors, Images
- Lazy initialization
- Change notification
- LocalResourceManager
- Standard resources
  - Icons, font selections, (texts for buttons)

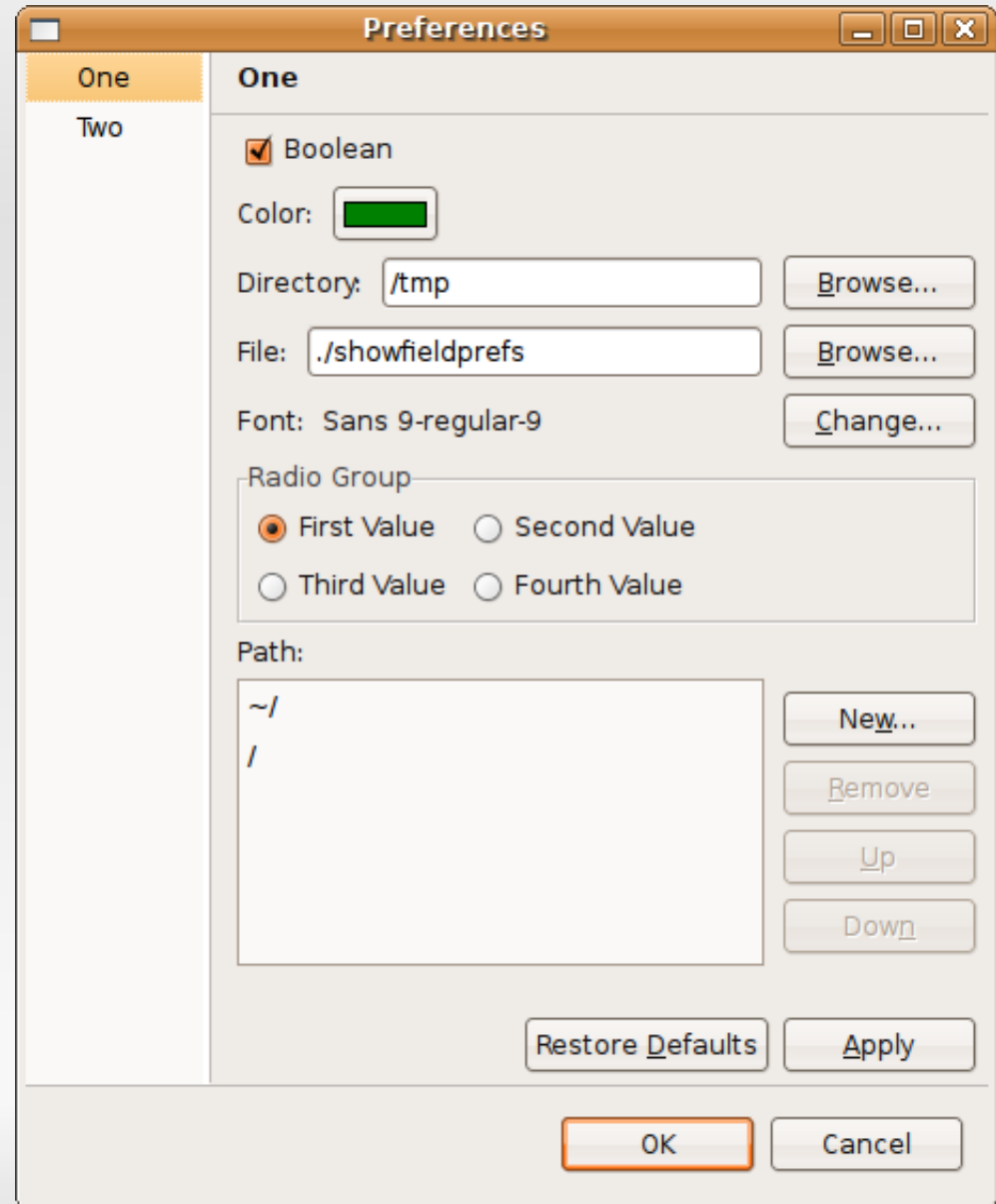


Tip: see Tool `dwtx.sleak.Sleak`

# JFace: Preferences

- PreferenceStore
- Preference Dialogs
  - Various Field Editors
  - Automatic Layout
  - Tree navigation for Pages

```
public class FieldEditorPageOne : FieldEditorPreferencePage {
    public this() {
        super(FLAT);
    }
    protected void createFieldEditors() {
        auto parent = getFieldEditorParent();
        addField(new BooleanFieldEditor("myBoolean", "Boolean", parent));
        addField(new ColorFieldEditor("myColor", "Color:", parent));
        addField(new DirectoryFieldEditor("myDirectory", "Directory:", parent));
        addField(new FileFieldEditor("myFile", "File:", getFieldEditorParent());
        addField(new FontFieldEditor("myFont", "Font:", getFieldEditorParent());
        addField(new RadioGroupFieldEditor("myRadioGroup", "Radio Group", 2,
            [ [ "First Value", "first"], [ "Second Value", "second"],
            [ "Third Value", "third"], [ "Fourth Value", "fourth"]], parent, true);
        addField(new PathEditor("myPath", "Path:", "Choose a Path", parent);
    }
}
```



# JFace: Decoration & Field Assist

## Snow Decoration

 Type into the text field

 with Decoration

without Decoration



- Show decoration icons on the horizontal edges of controls
- Content proposals with text and optional icons.

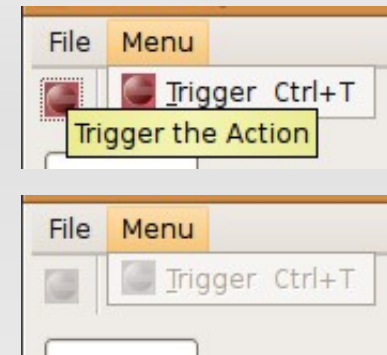
 The very first proposal

 second proposal

 and a last one

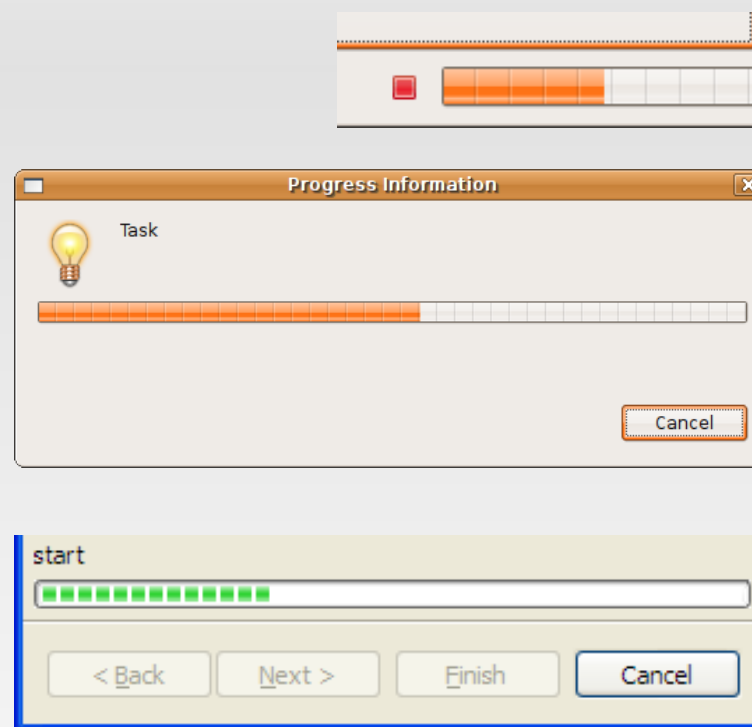
# JFace: Actions, Contributions

- Actions with Shortcuts, Icons, Tooltip, Enabled, Handler.
- Management of Menus, Statusline and Toolbars
  - Decoupling of Menu construction and places of where the action is created.
  - Marker groups, symbolic names
- Undoable operations with context → `dwtx.core.commands.operation`



# JFace: Long running operations

- Showing progress
  - Nestable scaled ProgressMonitors
- Modal operations
- Background activities  
→ `core.runtime.jobs`

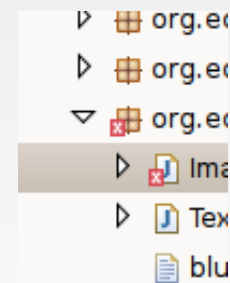


Tip: `dwtx.jface.dialogs.ControlEnableState`

# JFace: Viewers

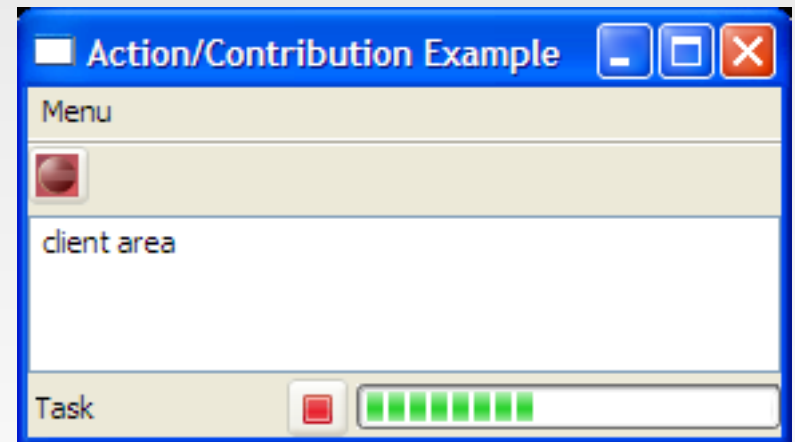
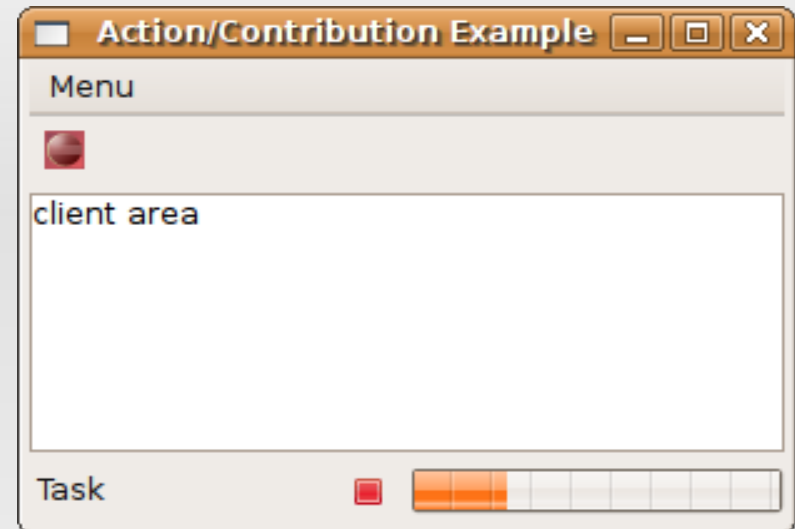
- Model-View-Controller for Table, Tree, List
- ContentProvider
- LabelProvider
- EditingSupport, CellEditor, validation
- ViewerSorter, ViewerFilter
- LabelDecorator

Givenname	Surname
Frank	Benoit
John	Reimer



# JFace: ApplicationWindow

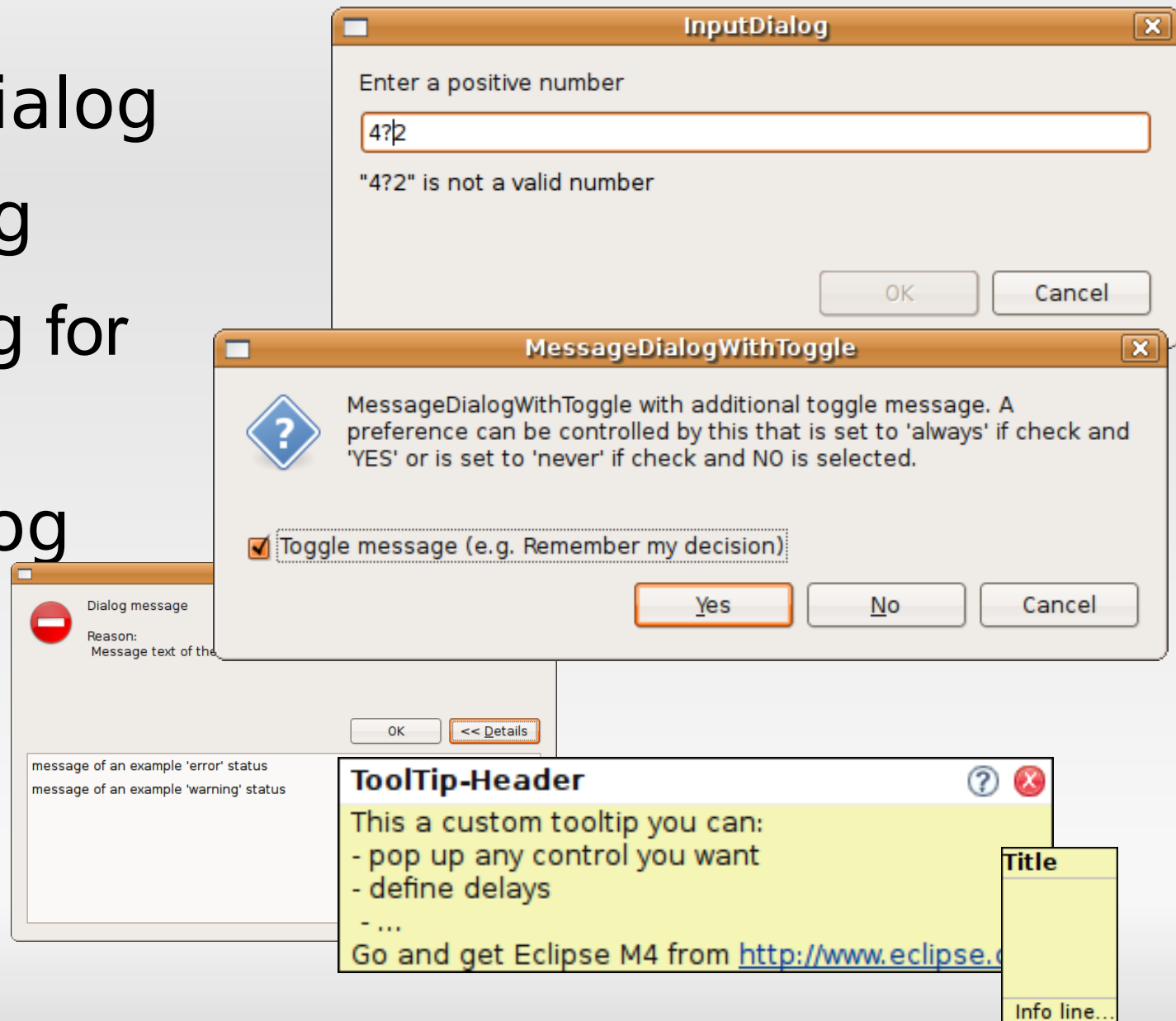
- Ready to use
- Internal Event Loop
- MenuBar, ToolBar, StatusLine.
- Long running operations



Tip: `dwtx.jface.window.WindowManager`

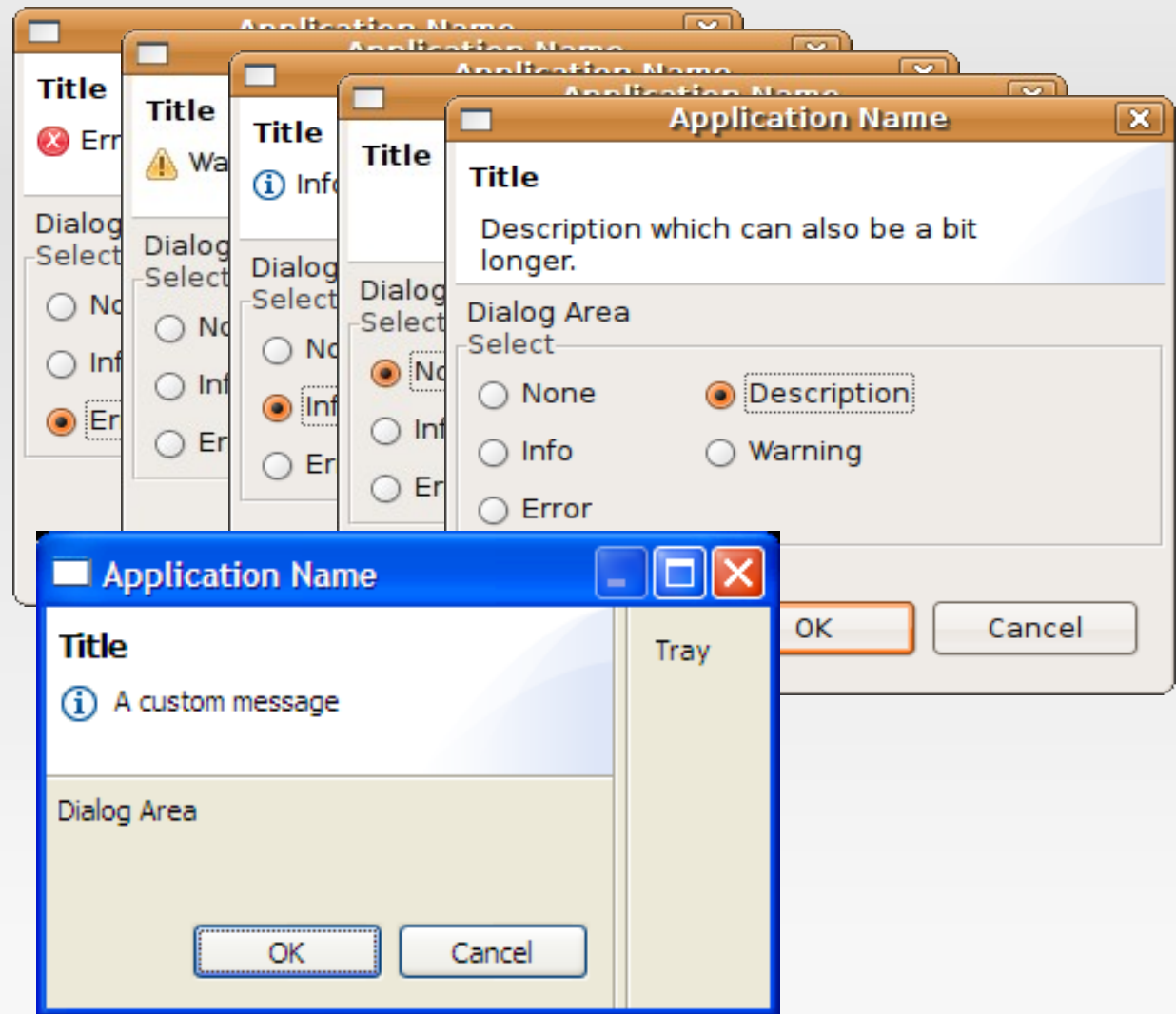
# JFace: Dialogs

- MessageDialog
- InputDialog
- ErrorDialog for IStatus
- PopupDialog
- ToolTip



# JFace: TitleAreaDialog

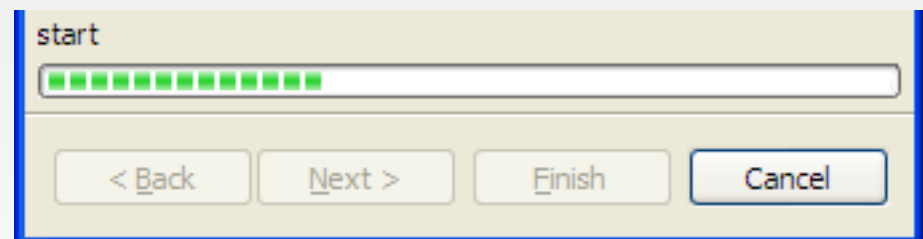
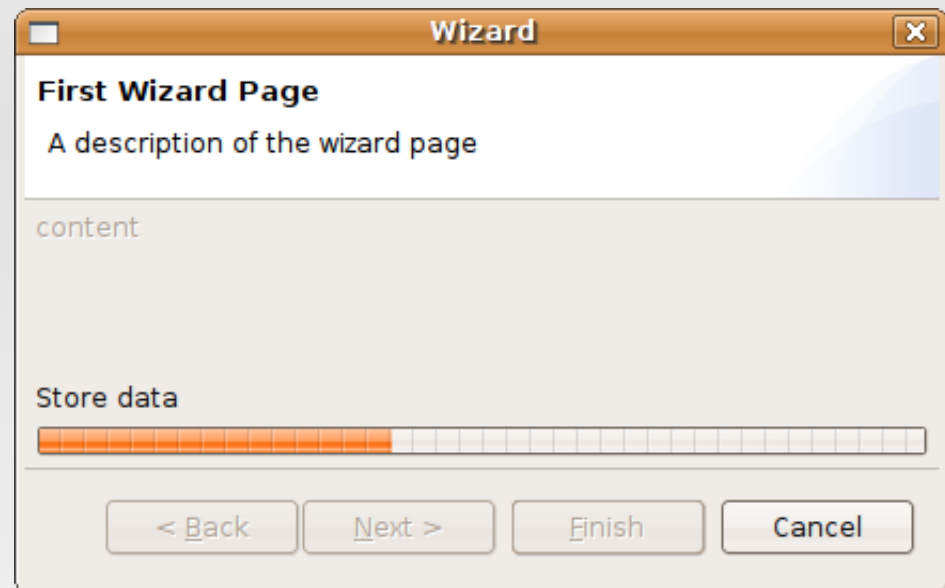
- Title
- Description, Message
- Button Bar
- optional Tray



Tip: `dwtx.jface.dialogs.DialogSettings`

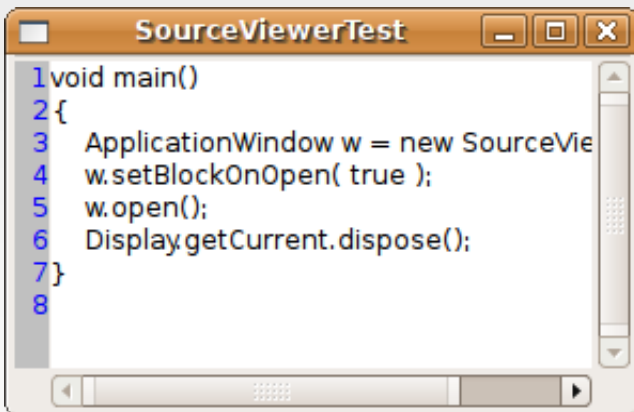
# JFace: Wizards

- Custom Page sequence
- Long running operation support
- Nested Wizards

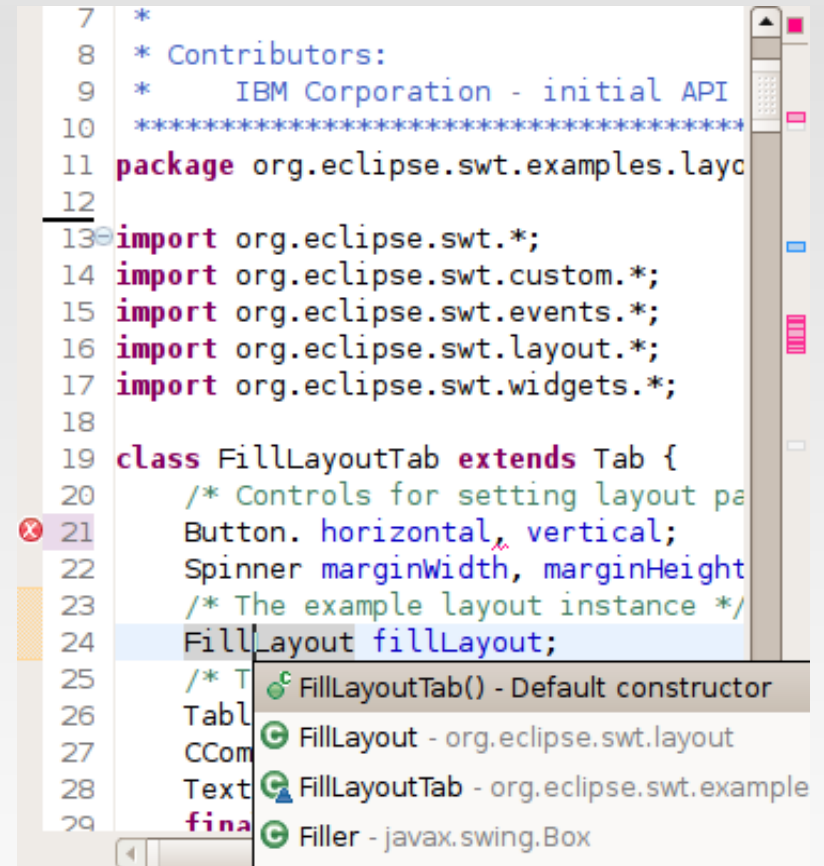


# JFace: Text

- Text Editor component
- rulers, folding, hovers, syntax highlighting, autocompletion, annotations, ...



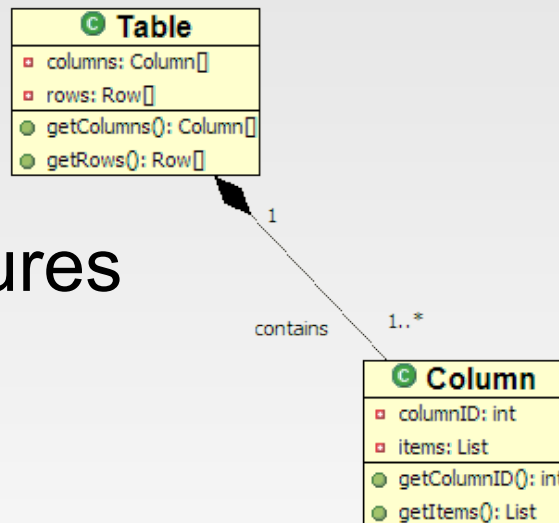
With current DWT port



Eclipse

# UI.Forms & Draw2D

- UI.Forms
  - Flat Look & Feel
  - Declarative GUI with XML
- Draw2D
  - Hierarchical Figures



## Hello, Eclipse Forms

This is an example of a form that is much longer and will need to wrap.

Text field label:

A checkbox in a form

**Expandable Composite title**

**Section title**


This is the description that goes below the title

Radio 1

Radio 2

Here is some plain text for the text to render; this text is at <http://www.eclipse.org> web site.

**This text is in header font and color.**

This line will contain some **bold** and some source text. We can also add  an image.

- A default (bulleted) list item.

- Another bullet list item.

1. A list item with text.

2. Another list item with text

-  List item with an image bullet

3. A list item with text.

4. A list item with text.

leading blanks; more white new lines  
more **bb** white .

# Why SWT & Eclipse?

- License is open source
- Many possible platforms
  - Windows, Linux, Mac, PPC, 32/64 Bit
  - GTK, Motif, Carbon, Cocoa, WPF
- SWT + Eclipse activity
- Java code can be ported easily to D
  - Update merges possible

# Future of DWT

- JFace.DataBinding
- Workbench with Views and Editors
- OSGi – Equinox
- Rich Client Platform