



RAP (The Rich Ajax Platform)

plug-in technology for the web

Jochen Krause
RAP Project lead
jkrause@innoopract.com

why RAP



RAP enables
**component oriented development and assembly
of web applications**

using
SWT, JFace and Workbench technology

what we like about Eclipse



plug-ins, plug-ins, plug-ins – bundles too ...

- dependency management
- extension points
- life cycle management

contribution to a common ui (workbench)

- this is called “mashup” in web 2.0

what we like about Eclipse (cont'd)



org.eclipse.rap.webdemo

Extensions

All Extensions ↓ a z

Define extensions for this plug-in in the following section.

org.eclipse.rap.ui.entrypoint

- org.eclipse.ui.perspectives**
 - standard (perspective)
 - another (perspective)
- org.eclipse.ui.views**
 - Message (view)
 - Mailboxes (view)
 - Navigationpane View (view)
 - Content View (view)
- org.eclipse.ui.commands
- org.eclipse.rap.ui.themes
- org.eclipse.rap.ui.branding
- org.eclipse.equinox.http.registry.resources
- org.eclipse.rap.ui.themeableWidgets

Extension Details

Set the properties of the selected extension. Required fields are denoted by "**".

ID:

Name:

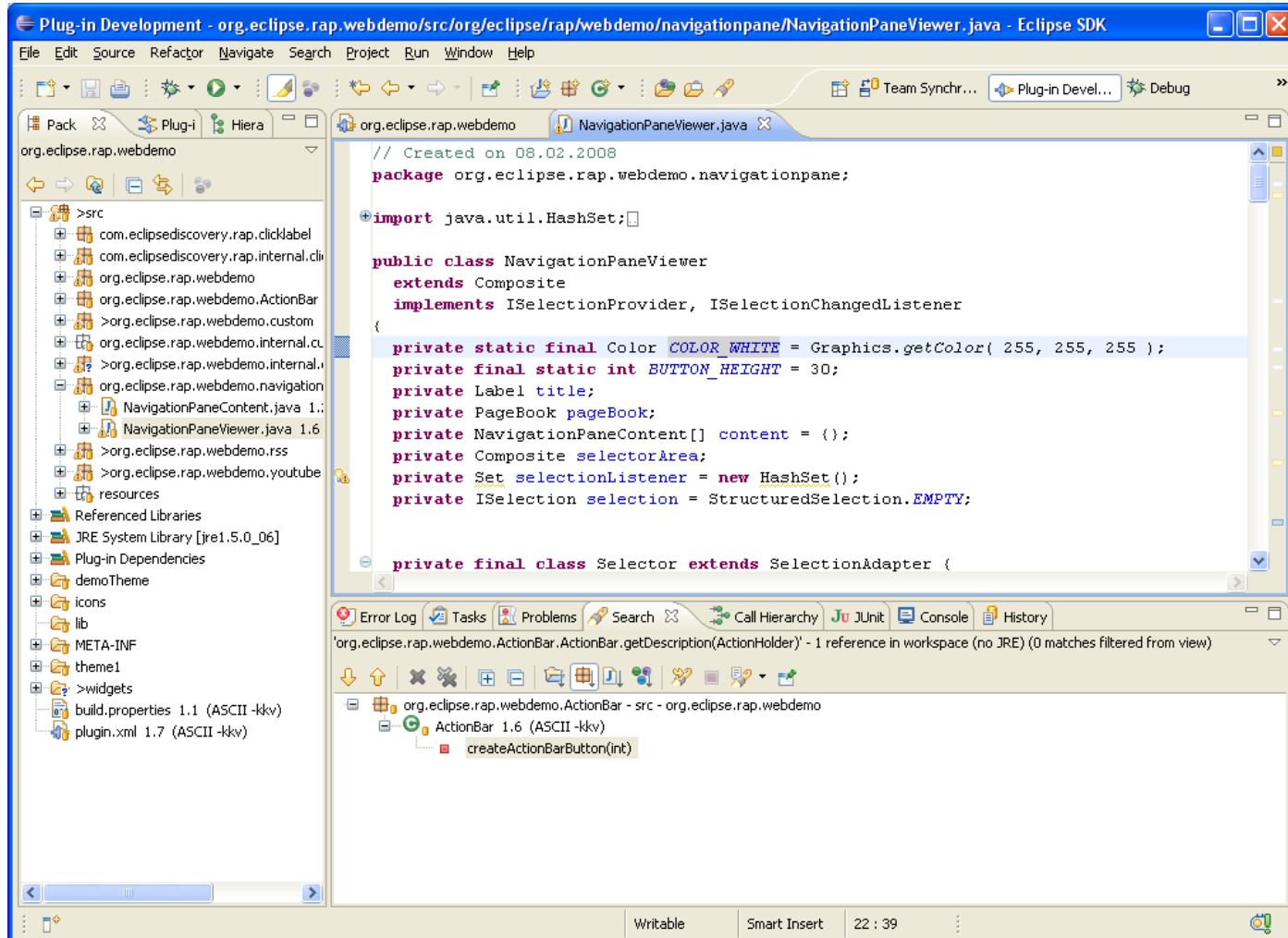
[Show extension point description](#)

[Open extension point schema](#)

[Find declaring extension point](#)

Overview Dependencies Runtime **Extensions** Extension Points Build MANIFEST.MF plugin.xml build.properties

what we like about Eclipse (cont'd)



how does that translate to the web?



code (plug-in) reuse

- 70% - 90% reuse is possible
- RAP provides only a subset of RCP
- needs separation of code that is not compatible
- application needs to become multi-user enabled
- no GC

getting all the things we like about Eclipse

nice idea – but I don' like RCP in a browser



Workbench Demo

File Window Help

View I View II Selection View Browser

Root
Locate in browser view
EclipseCon location
Eclipse Foundation
Innoopract Inc
Parent 2
Child X - filter me!

View III

Root

View IV View V

Column0	Column1	Column2	Column3	Column4	Column5	Column6
Item0-0	Item0-1	Item0-2	Item0-3	Item0-4	Item0-5	Item0-6
Item1-0	Item1-1	Item1-2	Item1-3	Item1-4	Item1-5	Item1-6
Item2-0	Item2-1	Item2-2	Item2-3	Item2-4	Item2-5	Item2-6
Item3-0	Item3-1	Item3-2	Item3-3	Item3-4	Item3-5	Item3-6

Revenue (in Millions)

- Classic Cars: 3.85
- Motorcycles: 1.12
- Planes: 0.95
- Ships: 0.66
- Trains: 0.19
- Trucks and Buses: 1.02
- Vintage Cars: 1.80

o.k. - but it does not have to be this way












Banner | click me 0 | click me 1 | click me 2

In | Out | Over | Under | Through | Perspective 1 | Perspective 2

Content

- This is Text for chapter 0.
- This is Text for chapter 1.
- This is Text for chapter 2.
- This is Text for chapter 3.
- This is Text for chapter 4.

The content goes here

- 0**  **recep ivedik fragman**
XD 7uUQKER0  2y 
youtube.com
Recep ?vedik'in beklenen filminin fragman?.
- 0**  **amazing guitar player**
m3qMqK7h-BA  2y 
youtube.com
....one of those that will make you say...holy %\$#^
- 0**  **Peanut and Jeff # 2**
EpRW8jh8AqY  2y 
youtube.com
Ventriliquist

and still uses workbench technology



The screenshot shows a browser window with the URL `http://127.0.0.1:2869/rap?startup=mail`. The interface is divided into several sections:

- Banner**: A blue header area containing three buttons labeled "click me 0", "click me 1", and "click me 2".
- commands**: A green bar below the banner with buttons labeled "In", "Out", "Over", "Under", and "Through".
- perspectives**: A green bar on the right side with buttons labeled "Perspective 1" and "Perspective 2".
- Content**: A list of four items, each with a radio button and the text "This is Text for chapter 0.", "This is Text for chapter 1.", "This is Text for chapter 2.", and "This is Text for chapter 3.".
- The content goes here**: A central area displaying three video thumbnails. Each thumbnail has a "0" in a yellow box and a "rap it" button below it. The videos are titled "recep ivedik fragman", "amazing guitar player", and "Peanut and Jeff # 2".

commands

perspectives

selection service

and still uses workbench technology (cont'd)



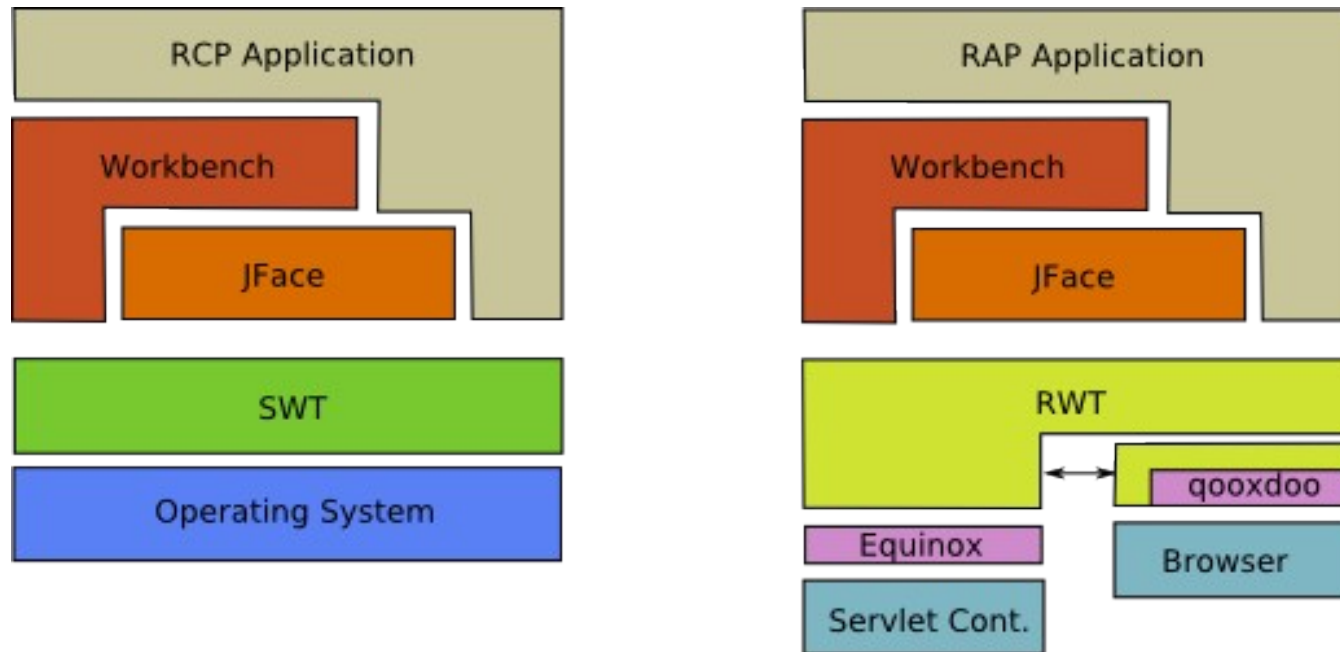
The screenshot shows a web browser window with the URL `http://127.0.0.1:2869/rap?startup=mail`. The application has a blue header bar with the word "Banner" and three buttons labeled "click me 0", "click me 1", and "click me 2". Below the header is a light green bar with five tabs: "In", "Out", "Over", "Under", and "Through". On the left is a sidebar with a tree view showing folders for "me@this.com" (containing "Inbox", "Drafts", "Sent") and "other@aol.com". The main content area is split into two panes. The left pane is labeled "views" in red text. The right pane is labeled "editors" in red text and displays an email message with the following details:

Subject: This is a message about the cool Eclipse RCP!
From: nicole@mail.org
Date: 10:34 am

Below the message details is a list of instructions:

- This RCP Application was generated from the PDE Plug-in Project wizard.
- add a top-level menu and toolbar with actions
- create views that can't be closed and multiple instances of the same view
- perspectives with placeholders for new views
- use the default about dialog

how does it work?



- replacing SWT with an implementation that can render to browsers
- everything else is pretty much the same
- RWT uses qooxdoo Javascript library to render widgets in the browser
- Difference: multi-user environment
 - Application and OSGi bundles shared between sessions

Developer's View on a RAP Application

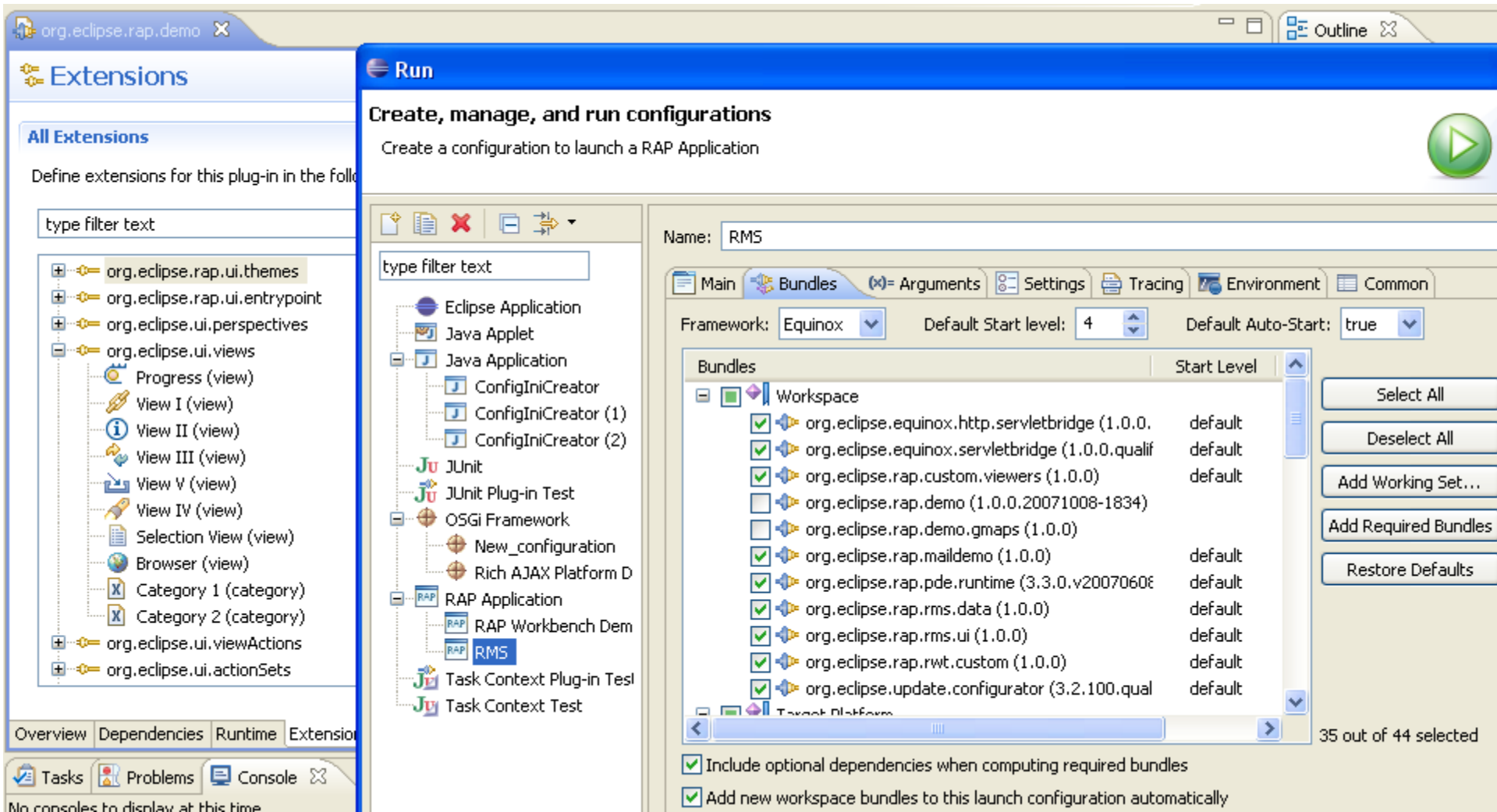


```
org.eclipse.rap.demo *DemoTreeViewPart.java x
+ * Copyright (c) 2002-2006 Innoopract Informationssysteme GmbH.
package org.eclipse.rap.demo;
+import java.util.ArrayList;
public class DemoTreeViewPart extends ViewPart implements IDoubleClickListener {
    private TreeViewer viewer;
- public void createPartControl( final Composite parent ) {
    viewer = new TreeViewer( parent );
    viewer.setLabelProvider(new DecoratingLabelProvider(new LabelProvider(),
                                                    new LeafStarLabelDecorator()));
viewer.setC
viewer.
viewer.
getSite
)
- private f
- public
    if(
    Press 'Ctrl+Space' to show Template Proposals
```

The AbstractTreeViewer implementation of this method checks to ensure that the content provider is an ITreeContentProvider.

- JDT: content assist, refactoring, etc., PDE, Javadoc available
- Developer does not get in touch with Javascript, CSS, HTTP ...

Developer's View on a RAP Application cont'd



The screenshot displays the Eclipse IDE interface with the 'Run' dialog box open. The dialog is titled 'Create, manage, and run configurations' and is used to launch a RAP application. The configuration name is 'RMS'. The 'Bundles' tab is selected, showing a list of bundles to be included in the launch configuration. The bundles are organized into a tree structure under 'Workspace'. The 'Start Level' column indicates the default start level for each bundle. The 'Include optional dependencies when computing required bundles' and 'Add new workspace bundles to this launch configuration automatically' checkboxes are checked.

Run Configuration Details:

- Name: RMS
- Framework: Equinox
- Default Start level: 4
- Default Auto-Start: true

Bundle	Start Level
org.eclipse.equinox.http.servletbridge (1.0.0)	default
org.eclipse.equinox.servletbridge (1.0.0.qualif	default
org.eclipse.rap.custom.viewers (1.0.0)	default
org.eclipse.rap.demo (1.0.0.20071008-1834)	
org.eclipse.rap.demo.gmaps (1.0.0)	
org.eclipse.rap.maildemo (1.0.0)	default
org.eclipse.rap.pde.runtime (3.3.0.v20070606)	default
org.eclipse.rap.rms.data (1.0.0)	default
org.eclipse.rap.rms.ui (1.0.0)	default
org.eclipse.rap.rwt.custom (1.0.0)	default
org.eclipse.update.configurator (3.2.100.qual	default

35 out of 44 selected

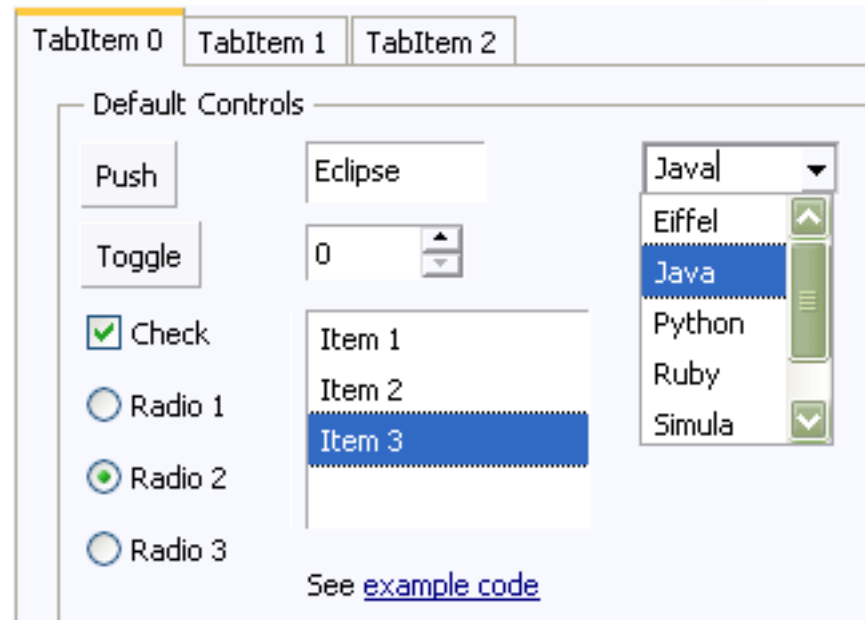
Include optional dependencies when computing required bundles

Add new workspace bundles to this launch configuration automatically

RWT Controls



- Most basic SWT widgets
- Advanced Workbench widgets
CTabFolder, CBanner, etc.
- Browser widget, external browser
- Tree improved
supports columns
- Table: new implementation
VIRTUAL, improved performance
CHECK, MULTI, images, cell colors



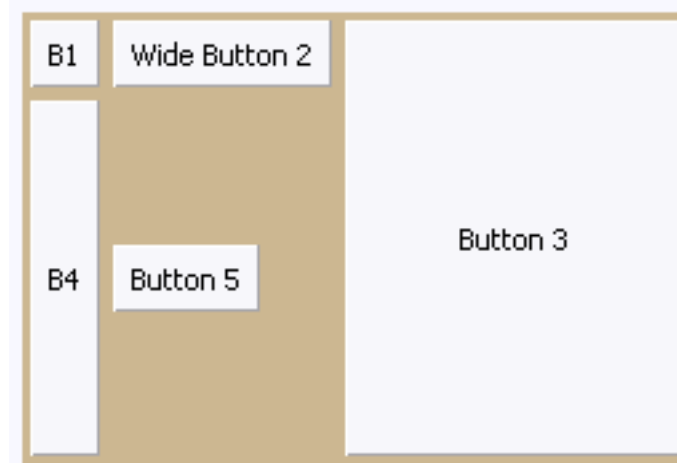
Column 1	Column 2
<input type="checkbox"/> Node_1	abc
<input checked="" type="checkbox"/> Node_2	jklmnop
<input type="checkbox"/> Subnode	jklmnop
<input checked="" type="checkbox"/> Node_3	jklmnop
<input type="checkbox"/> Node_4	jklmnop

Col 0	Col 1	Col 2	Col 3	Col 4
<input checked="" type="checkbox"/> Item0-0	Item0-1	Item0-2	Item0-3	Item0-4
<input type="checkbox"/> Item1-0	Item1-1	Item1-2	Item1-3	Item1-4
<input type="checkbox"/> Item2-0	Item2-1	Item2-2	Item2-3	Item2-4
<input type="checkbox"/> Item3-0	Item3-1	Item3-2	Item3-3	Item3-4
<input type="checkbox"/> Item4-0	Item4-1	Item4-2	Item4-3	Item4-4

RWT Layouts



- All usual layouts:
 - GridLayout,
 - RowLayout
 - FillLayout
 - FormLayout
 - StackLayout
 - and a lot more ...

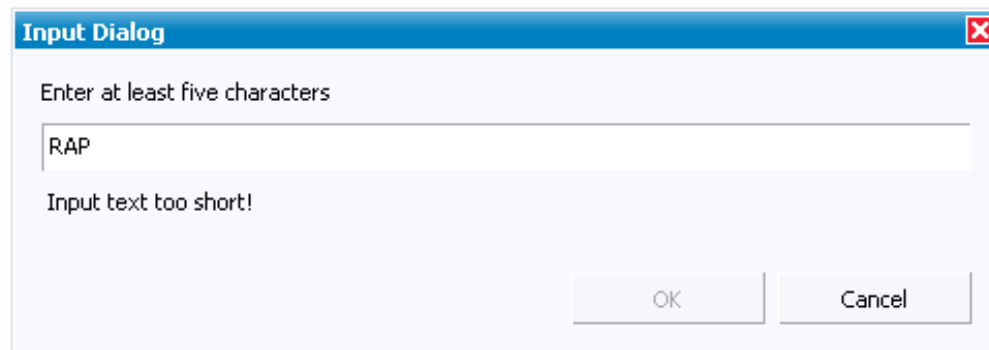


- mostly verbatim copies of SWT (OS independent)
- Layout algorithms work exactly as in SWT
- Layouts are computed on the server, e.g. after a Shell has been resized

RWT Events and Listeners



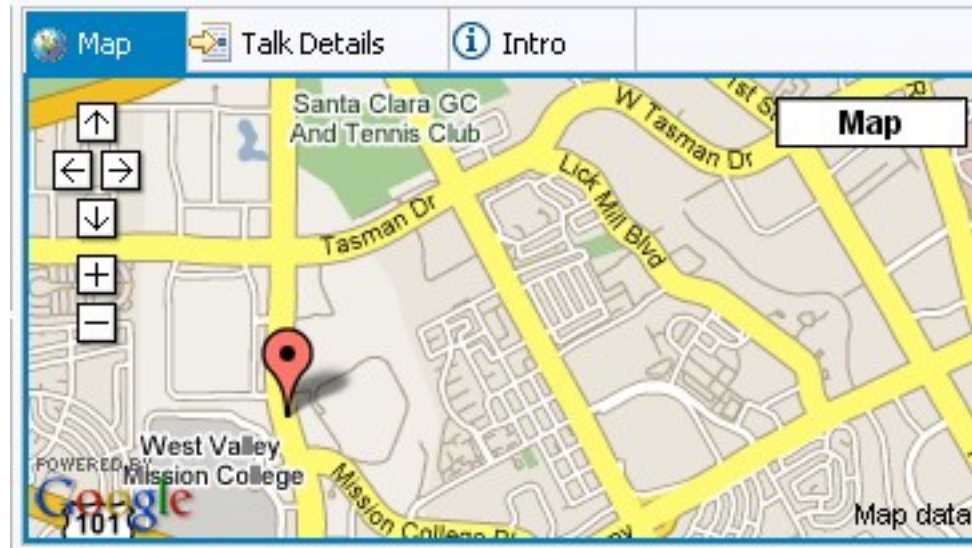
- Typed and untyped Listeners are supported
- Many Listeners implemented:
 - SelectionListener
 - ControllListener
 - ShellListener (supports doit flag now)
 - MenuListener
 - ModifyListener
 - VerifyListener (new)
 - ...



Custom Widgets



- Like in SWT, requires good knowledge of the platform
- Component developer needs Javascript, qooxdoo and RAP knowledge
- Application developer simply uses Java API
- Tutorial in RAP Help



```
GMap map = new GMap( shell, SWT.NONE );  
map.setAddress( "5001 Great America Pkwy, Santa Clara" );
```

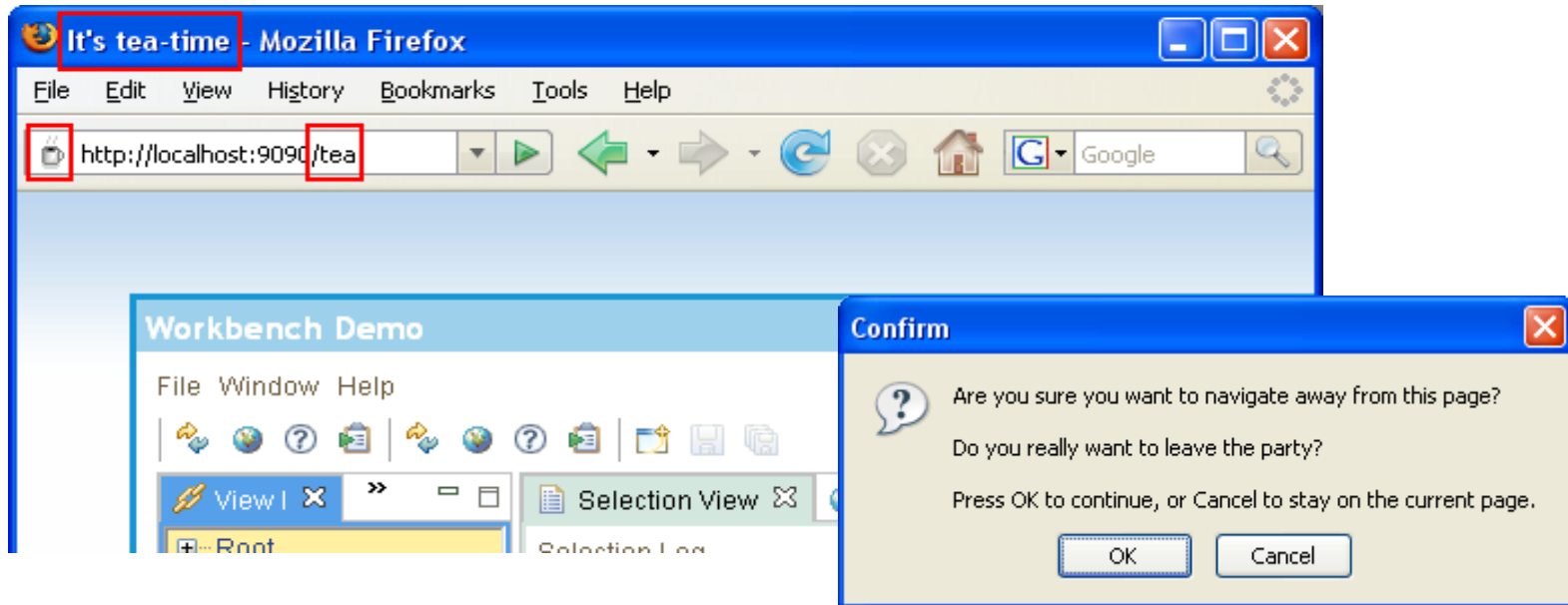
RWT Theming



- Objective: allow for a custom look of web applications
- Predefined properties of widgets can be customized
- Dimensions, Colors, Borders, Fonts, Images
- Simple Java .properties file
- Themeable custom widgets



RAP Branding



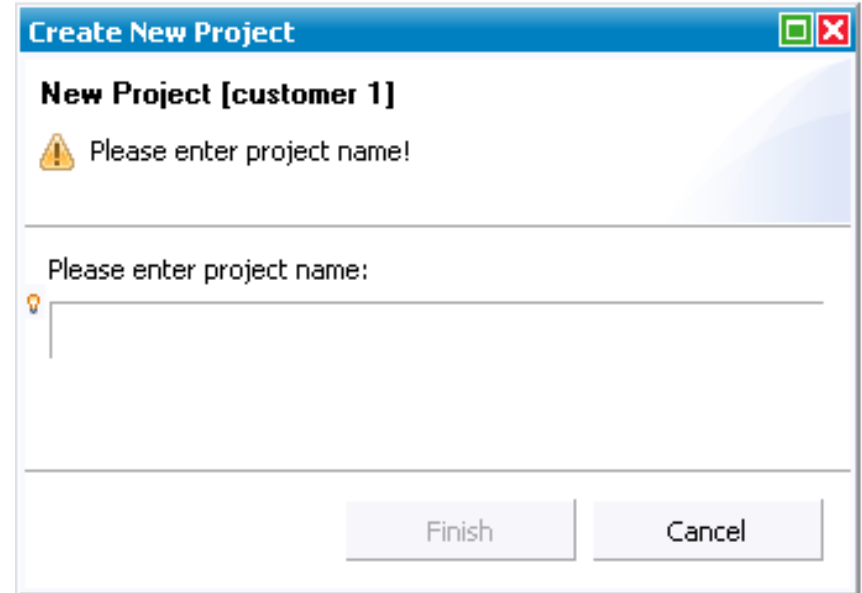
```
<extension
  point="org.eclipse.rap.ui.branding">
  <branding
    id="org.eclipse.rap.demo.branding1"
    servletName="tea"
    defaultEntrypointId="org.eclipse.rap.demo.entrypoint1"
    themeId="org.eclipse.rap.demo.alttheme"
    title="It&apos;s tea-time"
    favicon="icons/favicon2.ico"
    body="body.html"
    exitConfirmation="Do you really want to leave the party?">
  </branding>
</extension>
```

JFace



- Support for all JFace viewers
 - TableView
 - TreeView
 - ...
- Support for most Viewer concepts
 - Provider (Content, Label, Color...)
 - Sorter
 - Filter
 - Decorator (text only yet)
 -
- Support for Field Decorations
- **No Cell-Editors (yet)**

- Dialogs (blocking)
 - All standard JFace dialogs like
 - ErrorDialog
 - TrayDialog
 - Support for own Dialogs
- Wizards



Workbench: Parts, Perspectives & Interaction



- Full support for views
 - Additional views
 - Outline
 - Properties
- Editor support
 - Multi-page editors available
 - ISaveablePart available
- Support for perspectives
 - Perspective Switcher
 - Extensions
- New Eclipse 3.3 Menus Framework
 - Commands & Handler
 - Expression support for visibleWhen and enabledWhen

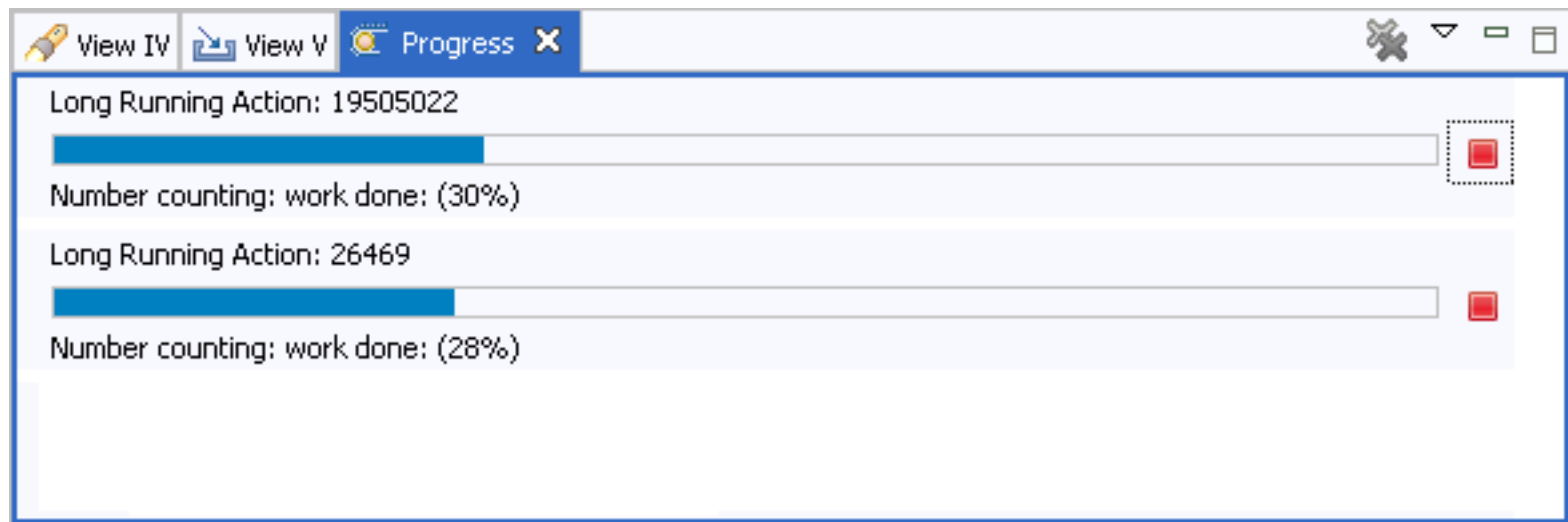
The screenshot shows the Eclipse Properties view for a project named 'project 2'. The view is titled 'Properties' and contains a table with two columns: 'Property' and 'Value'. The 'End Date' property is currently selected and highlighted.

Property	Value
Name	project 2
Description	
End Date	Sun Oct 28 00:00:00 CEST 2007
Start Date	Tue Sep 18 00:00:00 CEST 2007

Workbench: Jobs



- General support for background jobs in RWT
- Support for long-running tasks in the UI
- Progress View



Additional bundles: Data Binding & Forms



- Introduction of 3.3 Data Binding
 - Support for all available RAP widgets
- Eclipse UI Forms
 - Initial support for Forms Toolkit and Forms Editors
 - UI does not yet mimic original implementation
- **First prototypes of RAP / EMF integration look promising**

View IV View V Eclipse Form X

Hello, Eclipse Forms

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

Text field label:

An example of a checkbox in a form

Expandable Composite title

We will now create a somewhat long text so that we can use it as content for the expandable composite. Expandable composite is used to hide or show the text using the toggle control

▼ **Section title**

This is the description that goes below the title

Radio 1

Radio 2

What' next?



- 2008-01-07 M1 Performance (Client side) - delivered
- 2008-02-20 M2 Events & Listeners (Mouse, maybe Key), theming of widgets with classes like in CSS, Activities, readAndDispatch
- 2008-04-05 M3 Drag & Drop, workbench
- 2008-05-20 M4 qooxdoo 0.8, finalize 3.4 workbench

Get the RAP - <http://eclipse.org/rap>



Demos

See some demos here

Downloads

Get the latest RAP release

The RAP project enables developers to build rich, Ajax-enabled Web applications by using the Eclipse development model, plug-ins with the well known Eclipse workbench extension points, JFace, and a widget toolkit with SWT API (using **qooxdoo** for the client-side presentation). The project has graduated from incubation and released its 1.0 release.

[Learn more ...](#)



References

- <http://www.eclipse.org/rap> - RAP project page
- <http://wiki.eclipse.org/RAP> - RAP project wiki
- <http://www.qooxdoo.org> - qooxdoo js library